Post-print version of the paper by Boyuan. Ma. et al. in Micron 116 (2019): 5-14. https://doi.org/10.1016/j.micron.2018.09.002

Fast-FineCut : Grain boundary detection in microscopic images considering 3D information

Boyuan Ma[a,b,c], Xiaojuan Ban[a,b,c,*], Ya Su[a,b,c], Chuni Liu[a,b,c], Hao Wang[d,*], Weihua Xue[d,e], Yonghong Zhi[f], Di Wu[g]

[a] *Beijing Advanced Innovation Center for Materials Genome Engineering*
[b] *School of Computer and Communication Engineering, University of Science and Technology Beijing, Xueyuan Road 30, Haidian District, Beijing, 100083, China*
[c] *Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing, 100083, China*
[d] *School of Materials Science and Engineering, University of Science and Technology Beijing, Xueyuan Road 30, Haidian District, Beijing, 100083, China*
[e] *School of Materials Science and Technology, Liaoning Technical University, China*
[f] *Mechanical and electrical design and research institute of Shanxi Province, Shengli Street 228, Xinghualing District, Taiyuan city, Shanxi province, 030009, China*
[g] *Department of ICT and Natural Sciences, Norwegian University of Science and Technology, Aalesund, Norway*

Abstract

The inner structure of a material is called its microstructure. It stores the genesis of a material and determines all the physical and chemical properties. However, the microstructure is highly complex and numerous image defects such as vague or missing boundaries formed during sample preparation, which makes it difficult to extract the grain boundaries precisely. In this work, we address the task of grain boundary detection in microscopic image processing and develop a graph-cut based method called Fast-FineCut to solve the problem. Our algorithm makes two key contributions: 1) an improved approach that incorporates 3D information between slices as domain knowledge, which can detect the boundaries precisely, even for the vague and missing boundaries. 2) a local processing method based on overlap-tile strategy, which can not only solve the "chain scission" problem at the edge of images, but also economize on the consumption of computing resources. We conduct experiments on a stack of 296 slices of microscopic images of polycrystalline iron (1600 × 2800) and compare the performance against several state-of-the-art boundary detection methods. We conclude that Fast-FineCut can detect boundaries effectively and efficiently.

Keywords: graph cut approaches, grain boundary detection, overlap-tile

[*] Corresponding author

Email addresses: boyuanma_tony@xs.ustb.edu.cn (Boyuan Ma), banxj@ustb.edu.cn (Xiaojuan Ban), suyacv@163.com (Ya Su), liuchuni1996@xs.ustb.edu.cn (Chuni Liu), hwang@ustb.edu.cn (Hao Wang), mrxuecn@126.com (Weihua Xue), yonghong_zhi@126.com (Yonghong Zhi), di.wu@ntnu.no (Di Wu)

strategy, microscopic images, polycrystalline iron

1. Introduction

Microstructure is of great importance for controlling the properties and performance in materials science (Hu et al., 2017). During the quantitative analysis of microstructure, an important step is microscopic image processing (Sonka et al., 2014), which is used for extracting the key information in microstructure (Lewis & Howe, 2014). As shown in Figure.1 with images (or slices) of polycrystalline iron, this means detecting a grain boundary: the interface between two grains (Cantwell et al., 2014), shown in Figure.1(b) with straight and thick arrows. However, grain boundary detection is a highly challenging problem in microscopic image processing [1] for several reasons. First, the microstructure of material is highly complex (Wu et al., 2010). For example, there are hundreds of substructures with different shapes and sizes such as grains in metallic materials (Hu et al., 2017) or cells in biomaterials (D'Amore et al., 2010). Those grains have to be detected with high accuracy in each image. Second, as shown in Figure.1 (b), numerous image defects such as vague or missing boundaries (straight and thin arrows), noise (curved arrows), and spurious scratches (notched arrows). These flaws are coming from sample preparation, image acquisition, or other processes. They usually lead to a difficult transfer from grain boundary network to digital structure. Third, most microstructure in different sections retain consistency in shape and topology (adjacency relations), as shown in Figure.1(a). It is critical to model and incorporate such domain knowledge in the detection method (Waggoner et al., 2013; Huffman et al., 2011). Fourth, microscopic images are usually large and of high resolution. This asks for an efficient boundary detection method with high speed and less memory consumption.

Many existing image processing methods have been – or have the potential to be – used for detecting boundaries in microscopic images. In general, they can be grouped into two categories: 2D image processing methods to directly detect boundaries and 3D image processing methods based on tracking information. Many existing 2D methods detect boundaries independently, such as otsu (Vala & Baxi, 2013), adaptive threshold (Dewan et al., 2011), canny (Mcilhagga, 2011), bayesian (Comer et al., 2011; Simmons et al., 2008), watershed (Li et al., 2007), EM/MPM algorithm (Comer & Delp, 2000), graph cut (Birkbeck et al., 2009), multi-scale morphological method (Ullah et al., 2014), fuzzy clustering (Ali et al., 2017), deep learning-based image segmentation (Boyuan M, 2018) and so on. However, those methods did not consider consistency between slices. In addition, 2D methods often over- or under-segment individual slices, and thus they resulted in many inevitable errors. Tracking-based methods are developed

[1] This challenge can be described by two similar terms in the field of computer vision, boundary detection and segmentation. Boundary detection means extracting grain boundaries in the image, while segmentation means separating different grains in the image. Our work prefers the former one.

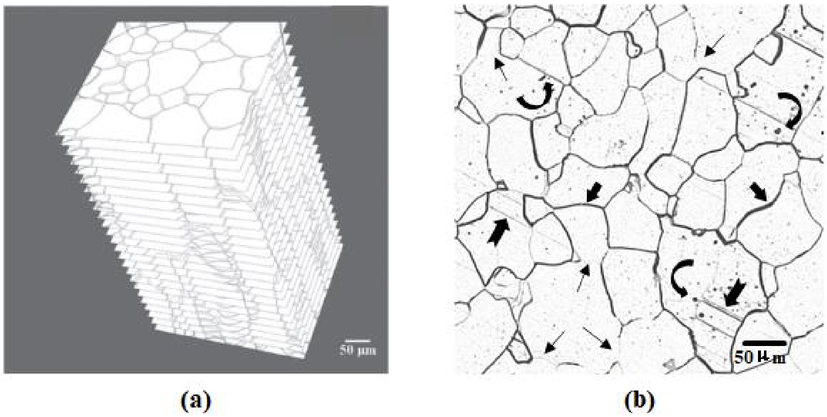


Figure 1: Microscopic images of polycrystalline iron. (a) Stack of serial sections for polycrystalline iron. (b) One slice of metallographic image, it contains grain boundaries (straight and thick arrows), vague or missing boundaries (straight and thin arrows), noise (curved arrows) and spurious scratches (notched arrows). For detailed visiualization, we only add the scale bar in this image. All microscopic images share same scale bar in this paper.

to detect boundaries in a stack of 2D slices. (Feng et al., 2017) proposed an interactive segmentation method based on breakpoint detection. But it will take a long time for manual refinement. (Waggoner et al., 2013, 2014) presented a more elegant architecture, "segmentation propagation", a graph-cut based segmentation method (Boykov & Kolmogorov, 2004; Kolmogorov & Zabin, 2004) which minimizes the energy set based on 3D information between slices, but can not recover the vague and missing boundary. Beside, with the slice's resolution increases, the computation time and memory consumption grow exponentially. We can not solve this by simply applying divide-and-conquer strategies because of 'chain scission' at image border.

In this work, we propose a boundary detection algorithm called Fast-FineCut, using 3D information between slices as domain knowledge. This algorithm is built on the propagative framework (Waggoner et al., 2013) and modifies its architecture, so that it can yield more precise result and make it possible to detect the boundaries for slices with high resolution. To the practical extent, the two advantages of our Fast-FineCut algorithm are :

(1) Accuracy: we modify and extend the binary term of multi-label graph-cut (Boykov et al., 2002) in the framework of propagative method (Waggoner et al., 2013), such that it can make better use of domain knowledge between slices and detect boundaries more precisely, even for the vague and missing boundaries. As shown in our experiments, Fast-FineCut has highest figure of merit (Lopez-Molina et al., 2013) and small deviation compare to other methods.

(2) Speed: by virtue of the idea of divide-and-conquer (Ronneberger et al., 2015), we propose a local propagative method based on overlap-tile strategy

(shown in Figure.6). This strategy can not only save the memory consumption, but make it faster than the global propagation. In addition, it will deal with the problem of ”chain scission” at the edge of the image (shown in Figure.7), which appears in the method of simple local propagation.

We share our code at web site: https://github.com/clovermini/Fast-FineCut The remainder of this paper is organized as follows: In Section 2, we provide a brief introduction to the material background in this paper. Section 3, describes the Fast-FineCut Algorithm and the overlap-tile strategy. We present, in Section 4, our experiments on different segmentation methods to show the effectiveness and efficiencies of the proposed approach. In Section 5, we show the 3D reconsruction result. Finally, We conclude the paper in Section 6.

2. Material

The specimen was intercepted from a hot-rolled iron slab and forged into round bars with a diameter equals to 30 millimeter. The pure iron bars were then fully recrystallized by annealing at 880°C for 3 hours to gain uniform grain microstructures in two dimensions. The samples were polished for a fixed time, and each polished layer was etched with 4vol% nital solution in preparation for optical microscopy. Images of microstructure were collected by an optical microscope, and a total of 296 serial sections with an average section thickness of 1.8 µm were obtained. We used Leica VMHT 3000 Victorinox microhardness tester to produce two sets of points, which meant that new points were struck before the previous ones disappeared to mark for image alignment and control the thickness of each section. This procedure effectively ensured that the images of the same area of interest were collected.

3. Fast-FineCut Algorithm

3.1. Overview of the proposed propagation method

Figure 2 shows an overall flowchart of the Fast-FineCut algorithm. Firstly, we clip the previous and next slice with high resolution into sub-images based on overlap-tile strategy. Secondly, we apply our method to define the unary and binary term for each sub-images in next slice. Thirdly, we apply multi-label graph-cut algorithm to get segmentation and edge detection result. Fourthly, we stitch the detection results by using overlap-tile strategy. Finally, we could repeatedly propagate it to obtain detection result of remaining slices in the stack.

3.2. Proposed propagation method

After collecting the serial sections of the specimen, the key challenge is to detect the grain boundaries among all the intricate microscopic images. Based on the high consistency between two neighboring slices in materials' images, (Waggoner et al., 2013) proposed a segmentation propagation method by propagating

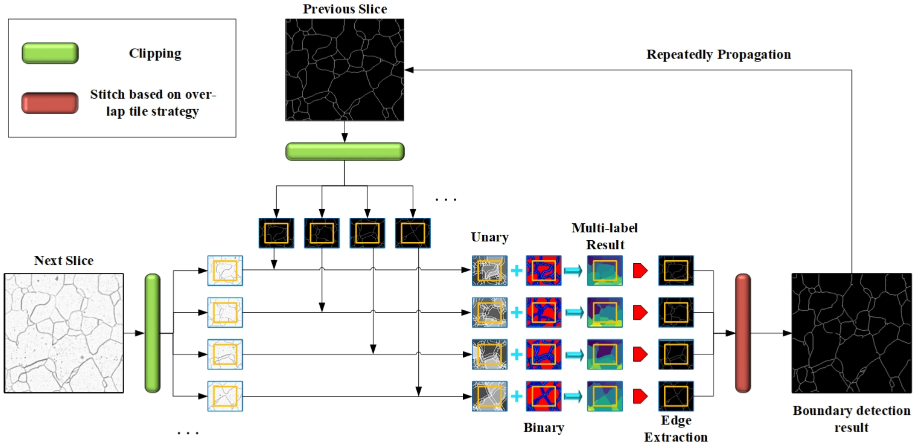


Figure 2: Flowchart of the proposed method.

segmentation S^P of a slice P to its neighboring slice N [2], yielding segmentation S^N. In this way, using an initial segmentation on one slice, the algorithm can repeatedly propagating it to remaining slices in the stack.

S^P is a partition of the pixels in P into n segments $S^p = \{S_1^p, S_2^p, ..., S_n^p\}$, where

$$P = S_1^p \cup S_2^p \cup ... \cup S_n^p \ and \ S_i^p \cap S_j^p = \varnothing \quad \forall i \neq j \tag{1}$$

To propagate segmentation by concerning consistency between slices as domain knowledge, we minimize the energy according to graph-cut theory (Boykov & Kolmogorov, 2004; Kolmogorov & Zabin, 2004).

$$E(S^N) = \sum_{p \in N} \Theta_p(S_i^N) + \sum_{\{p,q\} \in P_n^N} \Phi_{p,q}(S_i^N, S_j^N) \tag{2}$$

Where P_n^N is the set of all four-connected pixels in slice N. The four-connected pixels are pixels whose x or y coordinate (but not both) differ by no more than 1. The unary term $\Theta_p(S_i^N)$ symbols the cost when a particular pixel p being assigned to a segment S_i^N, while the binary term represents the cost for two four-connected pixel p and q ,which belong to S_i^P and S_j^P, being assigned to S_i^N and S_j^N respectively, subject to the structure continuity between P and N. They all constrained by the homomorphism of the RAGs(Region Adjacency Graphs [3]) in segmentation.

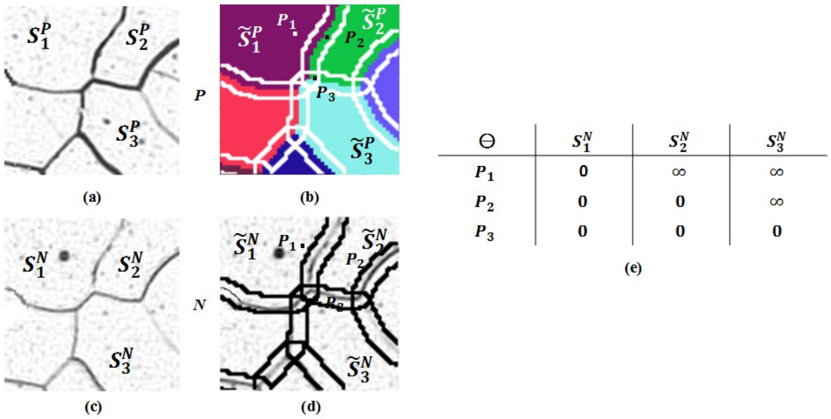
[2] P represents the previous image in the stack, while N represents the next image beneath the P.

[3] Region Adjacency Graphs is a graph in which the nodes correspond to the area, and the adjacent areas are connected by arcs.

We set the unary term like (Waggoner et al., 2013):

$$\Theta_p(S_i^N) = \begin{cases} 0 & p \in \widetilde{S_i^N} \\ \infty & p \notin \widetilde{S_i^N} \end{cases} \tag{3}$$

In the unary term, $\widetilde{S_i^N}$ is the bounding region, containing all the pixels p in N that with a distance less than d from any pixels in S_i^N. As shown in Figure.3, this bounding region stipulates the largest possible region that S_i^N may occupy. In addition, the overlap bounding region shows the maximum possible varying area of grain boundary, which has shown in Figure.3 (d).



Θ	S_1^N	S_2^N	S_3^N
P_1	0	∞	∞
P_2	0	0	∞
P_3	0	0	0

Figure 3: An illustration of defining the unary term in the proposed approach. (a) Three adjacent segments in P, represented by S_1^P, S_2^P, S_2^P respectively. (b) Label assignment of (a). White line denotes bounding region $\widetilde{S_i^P}$. (c) Three adjacent S_i^N in N beneath the P. (d) The overlap bounding region shows the maximum possible varying area of grain boundary. For clarity, associated bounding region $\widetilde{S_i^N}$ is represented by the black line. (e) Unary term value Θ of pixels $p1$, $p2$ and $p3$ in (d), which are located at bounding region $\widetilde{S_1^N}, \widetilde{S_2^N}, \widetilde{S_1^N}$ respectively.

During the processes of sample preparation and image acquisition, numerous image defects such as vague or missing boundaries, noise, and spurious scratches occur, as shown in Figure.1 and Figure.4. In Figure.4(a), the boundary (i) and (ii) are too vague to detect by traditional boundary detection method (Figure.4(e)) or even by naked eye. In this paper, we modify the binary term of Waggoner's energy (Waggoner et al., 2013) according to this phenomenon.

$$\Phi_{p,q}(S_i^N, S_j^N) = \phi_{p,q} \times \delta(S_i^N, S_j^N) \tag{4}$$

$$\delta(S_i^N, S_j^N) = \begin{cases} 1 & i \neq j \\ 0 & otherwise \end{cases} \tag{5}$$

$$\phi_{p,q} = \begin{cases} \infty & p,q \notin \tilde{S_i^N} \cap \tilde{S_j^N} \quad and \quad i = j \\ M \times g(p,q) & p,q \in \tilde{S_i^N} \cap \tilde{S_j^N} \quad and \quad i = j \\ K & p,q \in S_i^N \cap S_j^N \quad and \quad i \neq j \end{cases} \tag{6}$$

where $p \in S_i^P$ *and* $q \in S_j^P$.

$$g(p,q) = exp(-\frac{max(V(p), V(q))^2}{2\beta}) \tag{7}$$

$$\beta = \left\langle (V(p) - V(q))^2 \right\rangle \tag{8}$$

For our method, we deal with edge image of N, which show a higher intensity at the pixel along the boundaries of a substructure than the pixels within each substructure (as shown in Figure.4 (e)). In the binary term, the constants K and M satisfy the condition: $0 < K < M < \infty$. $V(p)$ is the intensity of p in the edge image of N, and function g in formula (7) describes the possibility of pixels p and q are along an image edge, β represents the variance of edge energy over all images.

More specifically, as shown in Figure.4, the energy for pixels p and q outside overlap bounding region are set to infinity, which is marked as red in Figure.4(c). The energy of pixels p and q inside overlap bounding region are set to $M \times g(p,q)$, as shown in Figure.4(f), in which, the blue represents the non-edge part and the black represents the edge part. Besides, we set the energy of pixels p and q, which belong to two different segments in P, to constant K which is represented by green line, as shown in Figure.4 (g). In this paper, we set M equals 10 and K equals 3.

In (Kolmogorov & Zabin, 2004), it shows that the graph-cut algorithm can find the global minimum of formula (2) in polynomial time if there are only two segments, while it's a NP-hard problem to find the global minimum in the case of multi-label problem. However, the local minimum can be found quickly by $\alpha\beta$ swap or α expansion. In this paper, we use a constrained $\alpha\beta$ swap to minimize the energy and get the local minimum quickly, which means the swap only happened between the neighboring segments of P. After the propagation of the segmentation, the boundaries is detected as shown in Figure.4 (h).

3.3. Local propagation method based on Overlap-tile strategy

Unfortunately, due to the nature of Graph Cut theory, as the image's resolution increases, the computational time and memory space consumption grows exponentially.

Divide-and-conquer is a common way used to deal with this problem, it works by recursively breaking down a problem into two or more sub-problems of the same type, until these become simple enough to be solved directly. The solutions to the sub-problems are then combined to give a solution to the original problem. Based on that, we splits one image with large resolution into N small size sub-images to reduce the time and memory consumption. For example, if we

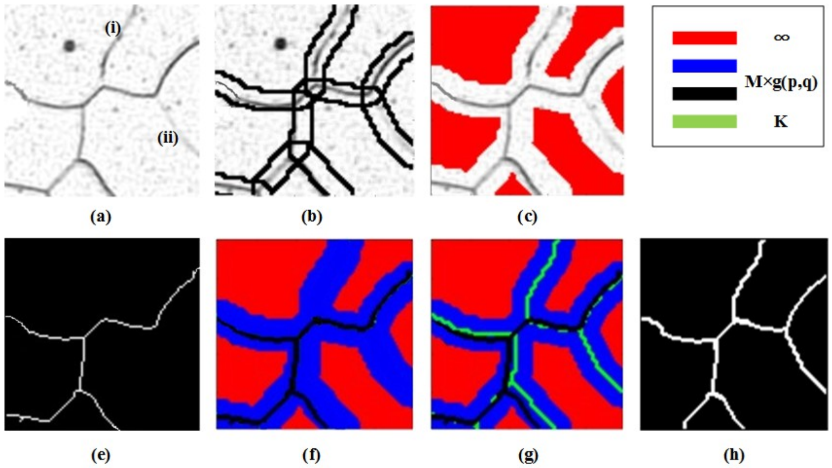


Figure 4: An illustration of defining the binary term in the proposed approach. (a) Original image N. (b) Bounding region same as Figure.3 (d). (c) Outside of overlap bounding region is set to infinity. (e) Edge detected by traditional algorithm. (f) Inside of overlap bounding region is set to function g. (g) The edge of previous layer's image is set to constant K. (h) Propagation result with modified binary term.

split a 400×400 size image into four 200×200 size image, the time consumption may be reduced to one fourth in theory.

However, as shown in Figure.5 with red boxes, we find that there are some "chain scission" problems at the edge of sub-images by applying simple local propagation. After comparing the binary term of global propagation with simple local propagation, we find the fact that the disappeared phenomenon only occurs along the border between sub-images.

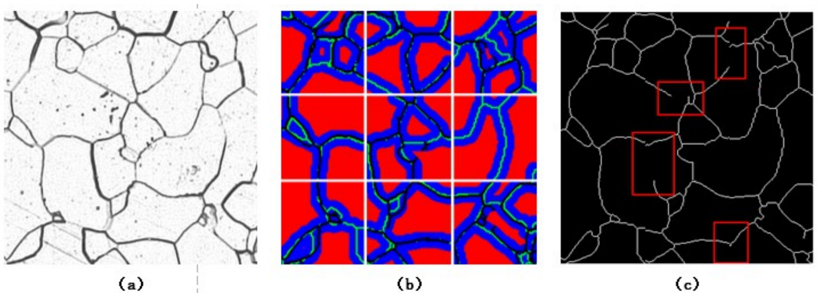


Figure 5: Propagation result of simple local propagation method. (a) Original image. (b) Binary term of simple local propagation. Gridlines represents the split strategy. (c) Propagation result with "chain scission" problems represented by red boxes.

In this paper, we introduce the overlap-tile strategy into local propagation method to solve the problem of "chain scission". As shown in Figure.6, we use image data within the blue area as input, but only pick up boundary detection result in yellow area. Note that the length of overlap region should be set as same as the length of bounding region to make sure all the disappeared boundaries are involved in it.

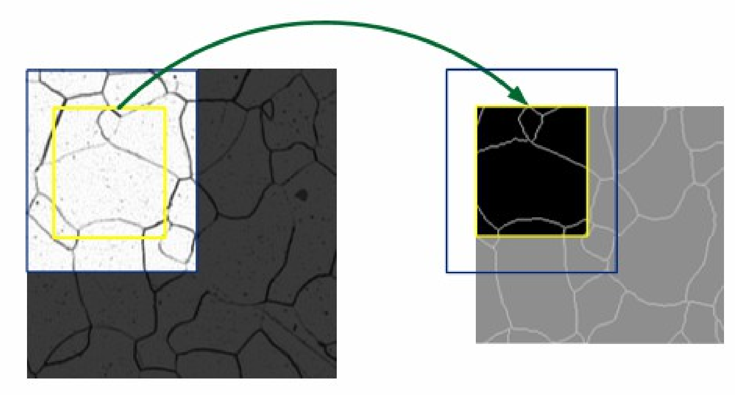


Figure 6: Overlap-tile strategy for seamless boundary detection of arbitrary large images. Result of boundary detection in the yellow area, requires image data within the blue area as input. Missing input data is extrapolated by mirroring.

The comparison of simple local propagation and local propagation based on overlap-tile strategy is shown in Figure.7. It is clearly to see that the local propagation based on overlap-tile strategy can remove the chain scission phenomenon effectively. At the same time, the efficient boundary detection with high speed and less memory space consumption makes it possible to analyze microscopic images with large resolution.

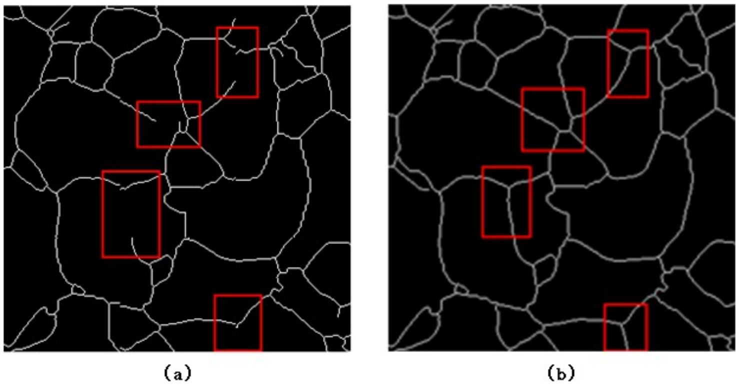


Figure 7: The comparison of simple local propagation and local propagation based on overlap-tile strategy. (a) Boundary detection result of simple local propagation. (b) Boundary detection result of local propagation based on Overlap-tile strategy.

4. Experimental Results

In this section, we use a stack of 296 microscopic pure iron images with large resolution (1600 × 2800 pixels) (shown in Fig.1) as the dataset to evaluate the proposed propagative boundary detection method. However, the dataset has large resolution, which is too large to apply global propagation with limit device resources. Therefore, we sample a sub-dataset to evaluate the performance of the proposed method and other methods. The sub-dataset consist of five group images chosen from 296 microscopic iron images randomly, each group includes a stack of 10 images with resolution 400 × 400.

We conduct five experiments to show the performance of our Fast-FineCut algorithm. The first four experiments use sub-datasets, and the fifth experiment uses the whole dataset. Firstly, according to the hyper-parameter, we adjust our Fast-FineCut algorithm to find the best performance. Secondly, we demonstrate that our algorithm could recover the vague and missing boundaries precisely using 3D information, while waggoner's method (Waggoner et al., 2013) couldn't handle it. Thirdly, we compare its performance to other boundary detection methods, such as Waggoner's method (Waggoner et al., 2013), Otsu (Vala & Baxi, 2013) Adaptive Threshold (Dewan et al., 2011) and Canny (Mcilhagga, 2011). Fourthly, we compare the computational time and memory consumption of global propagation and local propagation based on overlap-tile strategy. Fifthly, attribute to the success of the overlap-tile strategy, we could analyze the microscopy images with high resolution based on graph-cut theory. Therefore, we evaluate the performance of algorithms on the whole dataset.

For performance evaluation, we use the manual labeled boundaries detection of all slices as ground truth, provided by materials scientists in the work of (Feng et al., 2017). The ground truth detection was obtained by morphology segmentation using ImageJ software (Schindelin J, 2013), commonly used in microscopic image processing. However, it required 4 hours to carefully refine the result for each slice. To evaluate the accuracy of all boundary detection method objectively and precisely, we take the figure of merit (Lopez-Molina et al., 2013) as the evaluation criterion. It is defined as:

$$F = \frac{1}{max(N_{GT}, N_{Seg})} \sum_{i=1}^{N_{Seg}} \frac{1}{1 + \zeta d_i^2} \tag{9}$$

In formula (9), N_{GT} represents the number of pixels on boundary which marked by human, N_{Seg} represents the number of pixels on boundary detected by algorithm, and d_i describes the euclidean distance between the i^{th} boundary pixel detected by algorithm with its nearest real boundary pixel. The constant ζ is introduced to punish the boundary pixel that detected wrong, which usually equals 0.1. The bigger value of F, means the detected boundary is closer to the real boundary, in another word, the higher accuracy of boundary detection algorithm.

All the experiments are conducted on an E5-2687W Intel core of a 2.60GHz Windows workstation with 16GB of memory.

The energy minimization components are implemented in Python using the Numpy/SciPy (Van et al., 2014) and OpenCV (Laganiere, 2017), along with the publicly available graph cut optimization (GCO) library (Mueller, 2011).

4.1. The selection of hyper-parameter

The length of bounding region is the only hyper-parameter of the Fast-FineCut algorithm. We design an experiment to illustrate how the hyper-parameter affects the performance of our Fast-FineCut algorithm.

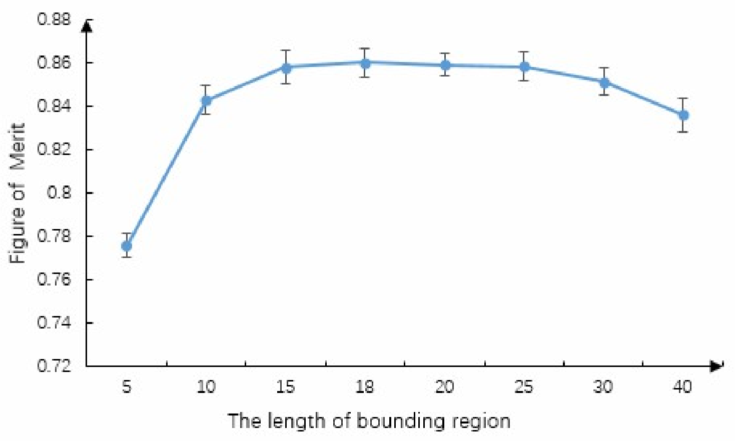


Figure 8: The selection of hyper-parameter

The length of bounding region affects the total length of overlap bounding region, it shows the maximum possible varying area of grain boundary, which is determined by the properties of the materials. We select the length of bounding region as 5, 10, 15, 18, 20, 23, 30 and 40 for comparsion respectively. The results of our Fast-FineCut algorithm are shown in Figure.8. It is clearly to see that the small length has poor performance, because it may cause some boundaries beyond the overlap region and then can not be detected, while the big length decreases the performance, because it may include some noise that affects the performance of our algorithm. According to the results, we assign the length of the bounding region equals 18.

4.2. Recovering the vague and missing boundary

For some microscopic images, there may exist some vague or missing boundaries in structure, as shown in Figure.9 ((a) is the previous slice, and (d) is the next slice which exists many vague and missing boundary).

In general, the section thickness of microscopic images is thin enough to produce two high-similarity neighboring slices (may be exactly same). Therefore, Fast-FineCut sets the binary term according to the boundary extracted from image of this layer in combination with the grain boundaries of the previous image (shown in Figure.9 (c)).

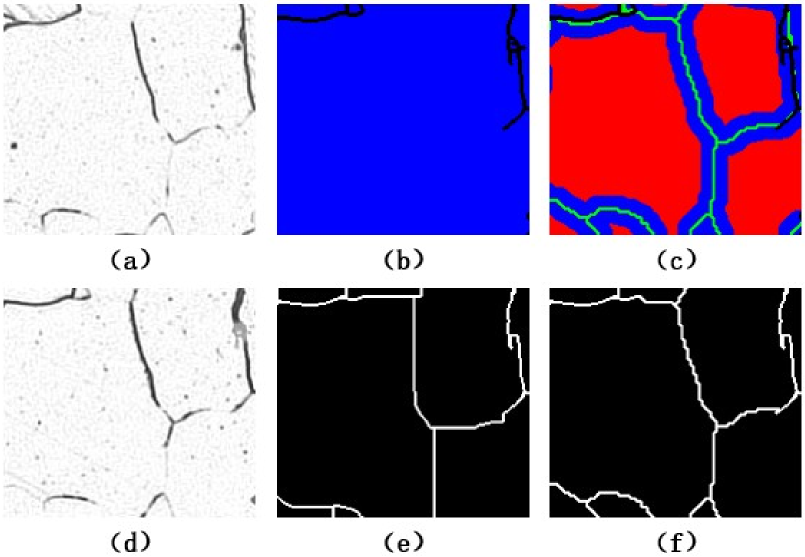


Figure 9: Recovering the vague and missing boundaries. (a) Original image P. (b) Binary term of Waggoner's method. (c) Binary term of method we improved. (d) Original image N with many vague and missing boundaries. (e) Detection Result with Waggoner's method. (f) Detection Result with method we improved.

As the propagtaion results shown in Figure.9 (e) and (f), it is obvious that the result of Fast-FineCut looks more natural and closer to the real image than that of Waggoner's method. (Waggoner et al., 2013).

4.3. Comparison Method

In order to justify the proposed method, we compare its performance to other boundary detection methods, such as Waggoner's method (Waggoner et al., 2013), Otsu (Lopez-Molina et al., 2013), Adaptive threshold (with mean kernel and Gaussian kernel) (Ronneberger et al., 2015), and Canny (Vala & Baxi, 2013). The contrast results are shown in Figure.10, from which we can see that our algorithm has the highest figure of merit (0.87) and smallest deviation compare to other methods. Comparing with 2D image process methods which only consider the information in one slice, such as otsu, adaptive threshold and canny, our algorithm combine 3D information between slices as domain knowledge. While comparing to Waggoner's method, our method modify and extended its binary term to yield more precise result.

4.4. The performance of local propagation based on overlap-tile strategy

The local propagation based on overlap-tile strategy can not only solve the problem of "chain scission" produced by simple local propagation, as shown in Figure.7, but also save the computational time and memory space. To demonstrate it, we compare the computational time and memory space consumption

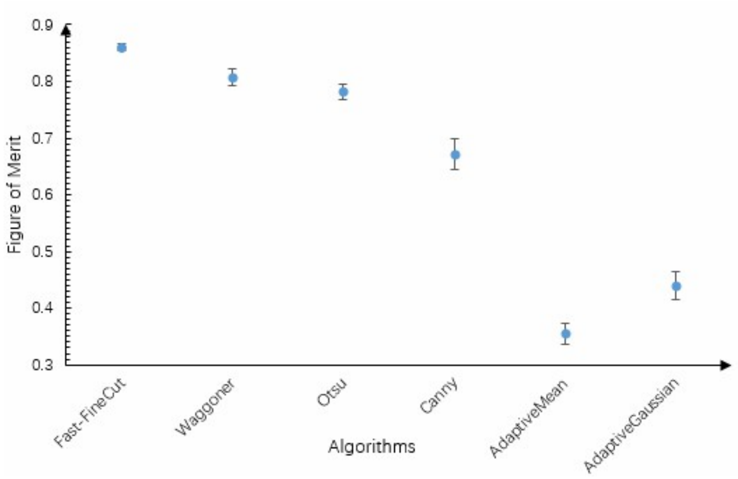


Figure 10: Comparison method

of global propagation with local propagation based on overlap-tile strategy in 200×200, 400×400, 600×600, 800×800, 1000×1000 sized images.

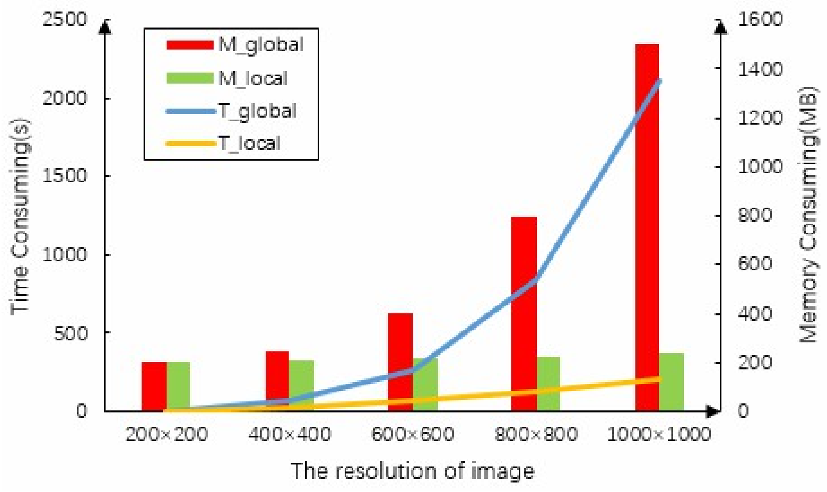


Figure 11: Comparison of global propagation and local propagation based on overlap-tile strategy(M denotes memory, T denotes time)

As shown in Figure.11, with the resolution of image increases, the time and memory space consumed by the global propagation method grows exponentially, while those of overlap-tile strategy based local propagation grows linearly. That is because the time complexity of graph-cut is $O(mn^2|C|)$, where n is the number of nodes and m is the number of edges in the graph, $|C|$ is the cost of the minimum cut (Boykov & Kolmogorov, 2004). In addition, the memory consumption do not changes with the increase of the resolution. This strategy make it possible to analyze the high resolution of microscopic images, which is common in material science.

4.5. The Evaluation of Fast-FineCut in whole dataset

Attribute to the success of overlap-tile strategy, we can analyze the microscopic images with high resolution. In this section, we illustrate the performance of seven methods apply to the high dataset (stack of 296 microscopic pure iron images with resolution of 1600×2800). For fair comparison, we apply overlap-tile strategy both to Fast-FineCut and Waggoner's work.

As shown in Table 1, the ground truth provided by the materials scientists in the work of (Feng et al., 2017).It was obtained by morphology segmentation using ImageJ software (Schindelin J, 2013), commonly used in microscopic image processing. However, it required 4 hours to manually refine the result for each slice.

Since the combination of 3D information, the Fast-FineCut has the highest average figure of merit (0.87) compare to other methods. However, due to the nature of graph-cut theory, it cost much time both in Fast-FineCut and Waggoner's with overlap strategy. Besides, due to the complexity of binary-term computation, Fast-FineCut will cost more time than the Waggoner's with Overlap strategy. However, it could be acceptable beacause its highest accuracy. On the aspect of memory consumption, the Fast-FineCut and Waggoner's work equipped with overlap-tile strategy are local processing methods which could cost lower memory compare to other global methods, which make our method could be applied to practice. We visualize the results of segmentation in Figure.13 in Appendix.

Table 1: The average performance of one slice

Methods	**Figure of Merit**	**Time(s)**	**Memory Consumption(MB)**
Ground Truth	1.000	14,400.000	500
Fast-Fine Cut	**0.870**	**1478.033**	**200**
Waggoner with Overlap strategy	0.827	512.760	200
Otsu	0.782	0.006	270
Canny	0.738	0.019	280
Adaptive Mean	0.433	0.009	275
Adaptive Gaussian	0.586	0.014	275

There are still some problems exist in the propagative boundary detection algorithm. First, it is hard to detect the boundary which beyond the bounding region, as shown in the red cycle of last row in Figure.13. Second, because the nature of multi-label graph-cut theory, the number of label could not be changed in the process, so that it can not automatically detect the new grain boundary occurred in the volume, as shown in the cycle of 3rd row in Figure.13. Thirdly, it will cost much time on the computation in propagative framework. Our team will work on those problems in later work.

5. 3D reconstruction

After extracting all the grain boundaries of 296 slices with proposed algorithm, we used 297 serial sections (including the first manually labeled slice) to perform 3D reconstruction on the microstructure. The result is shown in Figure.12. The 3D reconstructed pure iron's volume is 1090 µm×1730 µm ×540 µm, it contains a sum of 16254 intact iron grains from more than 40,000 grains, including surface ones. For illustration, the picture and figures of the 3D reconstructed pure iron are provided by (Feng et al., 2017).

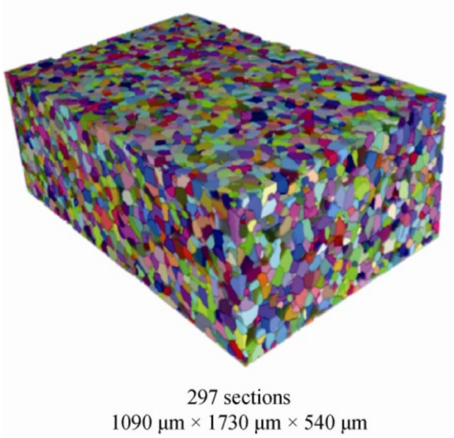


Figure 12: 3D reconstruction of 297 serial iron grain slices

6. Conclusion

In this paper, we proposed a propagative grain boundary detection algorithm called Fast-FineCut considering 3D information as domain knowledge. This algorithm is built on the Waggoner's work and modify their architecture so that it can detect the vague and missing boundary and yield more precise result. In addition, we developed a local propagation method based on overlap-tile strategy, it not only solve the "chain scission" problem, but save the consumption of resources (such as computational time and memory space). We tested the proposed method on a stack of 296 slices microscopic images of polycrystalline iron with high resolution and achieved promising performance that is superior to the previous methods and other state-of-the-art methods. Besides, Our method can support the other applications in 3D microscopic images, such as Magnetic resonance imaging (MRI) and computed tomography (CT) images.

7. Funding

Funding: This work was supported by the National Key Research and Development Program of China [grant number 2016YFB0700500]; the National Science Foundation of China [grant numbers 61572075, 61702036]; Fundamental Research Funds for the Central Universities [grant number FRF-TP-17-012A1]; and China Postdoctoral Science Foundation [grant number 2017M620619].

8. Appendix

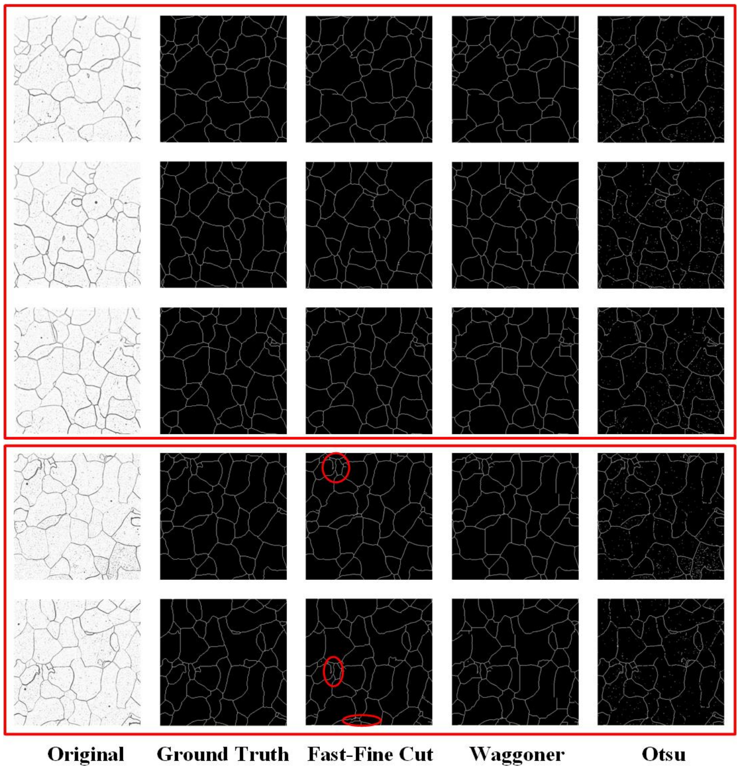


Figure 13: Detection result with various methods

References

Ali, M., Le, H. S., Khan, M., & Tung, N. T. (2017). Segmentation of dental x-ray images in medical imaging using neutrosophic orthogonal matrices. *Expert Systems with Applications*, .

Birkbeck, N., Cobzas, D., Jagersand, M., & Murtha, A. (2009). An interactive graph cut method for brain tumor segmentation. In *Applications of Computer Vision* (pp. 1–7).

Boykov, Y., & Kolmogorov, V. (2004). An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*, 1124–1137.

Boykov, Y., Veksler, O., & Zabih, R. (2002). Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *23*, 1222–1239.

Boyuan M, H. H. Y. C. W. L. Y. Z., Xiaojuan B (2018). Deep learning-based image segmentation for al-la alloy microscopic images. *Symmetry*, *10*, 107.

Cantwell, P. R., Tang, M., Shen, J. D., Luo, J., Rohrer, G. S., & Harmer, M. P. (2014). Grain boundary complexions. *Acta Materialia*, *62*, 1–48.

Comer, M., Bouman, C. A., Graef, M. D., & Simmons, J. P. (2011). Bayesian methods for image segmentation. *JOM*, *63*, 55–57.

Comer, M. L., & Delp, E. J. (2000). The em/mpm algorithm for segmentation of textured images: analysis and further experimental results. *IEEE Trans Image Process*, *9*, 1731–44.

D'Amore, A., Stella, J. A., Wagner, W. R., & Sacks, M. S. (2010). Characterization of the complete fiber network topology of planar fibrous tissues and scaffolds. *Biomaterials*, *31*, 5345.

Dewan, M. A. A., M Omair, A., & Swamy, M. N. S. (2011). Tracking biological cells in time-lapse microscopy: an adaptive technique combining motion and topological features. *IEEE transactions on bio-medical engineering*, *58*, 1637–47.

Feng, M. N., Wang, Y. C., Wang, H., Liu, G. Q., & Xue, W. H. (2017). Reconstruction of three-dimensional grain structure in polycrystalline iron via an interactive segmentation method. *International Journal of Minerals, Metallurgy and Materials*, *24*, 257–263.

Hu, J., Shi, Y. N., Sauvage, X., Sha, G., & Lu, K. (2017). Grain boundary stability governs hardening and softening in extremely-fine nanograined metals. *Science Foundation in China*, *355*, 1292.

Huffman, L., Simmons, J., Graef, M. D., & Pollak, I. (2011). Shape priors for map segmentation of alloy micrographs using graph cuts. In *Statistical Signal Processing Workshop* (pp. 661–664).

Kolmogorov, V., & Zabin, R. (2004). What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*, 147–159.

Laganiere, R. (2017). *OpenCV 3 Computer Vision Application Programming Cookbook*. Packt Publ. Limited,.

Lewis, A. C., & Howe, D. (2014). Future directions in 3d materials science: Outlook from the first international conference on 3d materials science. *JOM*, *66*, 670–673.

Li, Q., Ni, X., & Liu, G. (2007). Ceramic image processing using the second curvelet transform and watershed algorithm. In *IEEE International Conference on Robotics and Biomimetics* (pp. 2037 – 2042).

Lopez-Molina, C., Baets, B. D., & Bustince, H. (2013). Quantitative error measures for edge detection. *Pattern Recognition*, *46*, 1125–1139.

Mcilhagga, W. (2011). The canny edge detector revisited. *International Journal of Computer Vision*, *91*, 251–261.

Mueller, A. (2011). Graphcuts for python: pygco. `http://peekaboo-vision.blogspot.ca/2012/05/graphcuts-for-python-pygco.html`.

Ronneberger, O., Fischer, P., & Brox, T. (2015). *U-Net: Convolutional Networks for Biomedical Image Segmentation*. Springer International Publishing.

Schindelin J, F. E., Arganda-Carreras I (2013). Fiji: an open-source platform for biological-image analysis. *Nature methods*, *9*, 676.

Simmons, J. P., Bartha, B., Graef, M. D., & Comer, M. (2008). Development of bayesian segmentation techniques for automated segmentation of titanium alloy images. *Microscopy and Microanalysis*, *14*, 602–603.

Sonka, M., Hlavac, V., & Boyle, R. (2014). Image processing analysis and machine vision. *Journal of Electronic Imaging*, *xix*, 685686.

Ullah, A., Liu, G., Luan, J., Li, W., Rahman, M. U., & Ali, M. (2014). Three-dimensional visualization and quantitative characterization of grains in polycrystalline iron. *Materials Characterization*, *91*, 65–75.

Vala, M. H. J., & Baxi, A. (2013). A review on otsu image segmentation algorithm. *International Journal of Advanced Research in Computer Engineering and Technology*, *2*.

Van, d. W. S., Schnberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., Yu, T., & Contributors, S. I. (2014). scikit-image: image processing in python. *Peerj*, *2*, e453.

Waggoner, J., Zhou, Y., Simmons, J., De, G. M., & Wang, S. (2013). 3d materials image segmentation by 2d propagation: a graph-cut approach considering homomorphism. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, *22*, 5282.

Waggoner, J., Zhou, Y., Simmons, J., Graef, M., & Wang, S. (2014). Graph-cut based interactive segmentation of 3d materials-science images. *Machine Vision and Applications*, *25*, 1615–1629.

Wu, Q., Merchant, F. A., & Castleman, K. R. (2010). *Microscope Image Processing*. Academic press.