

Edge-Weighted Centroid Voronoi Tessellation with Propagation of Consistency Constraint for 3D Grain Segmentation in Microscopic Superalloy Images

Youjie Zhou*, Lili Ju[†], Yu Cao*, Jarrell Waggoner*, Yuewei Lin*, Jeff Simmons[‡] and Song Wang*

*Department of Computer Science and Engineering, University of South Carolina, SC 29208

[†]Department of Mathematics, University of South Carolina, SC 29208

[‡]Materials and Manufacturing Directorate, Air Force Research Labs, OH 45433

zhou42@email.sc.edu, ju@math.sc.edu, cao@cec.sc.edu, malloc47@gmail.com,

ywlin.cq@gmail.com, jeff.simmons@wpafb.af.mil, songwang@cec.sc.edu

Abstract—3D microstructures are important for material scientists to analyze physical properties of materials. While such microstructures are too small to be directly visible to human vision, modern microscopic and serial-sectioning techniques can provide their high-resolution 3D images in the form of a sequence of 2D image slices. In this paper, we propose an algorithm based on the Edge-Weighted Centroid Voronoi Tessellation which uses propagation of the inter-slice consistency constraint. It can segment a 3D superalloy image, slice by slice, to obtain the underlying grain microstructures. With the propagation of the consistency constraint, the proposed method can automatically match grain segments between slices. On each of the 2D image slices, stable structures identified from the previous slice can be well-preserved, with further refinement by clustering the pixels in terms of both intensity and spatial information. We tested the proposed algorithm on a 3D superalloy image consisting of 170 2D slices. Performance is evaluated against manually annotated ground-truth segmentation. The results show that the proposed method outperforms several state-of-the-art 2D, 3D, and propagation-based segmentation methods in terms of both segmentation accuracy and running time.

Keywords—3D image segmentation, segmentation propagation, grain segmentation, centroidal Voronoi tessellation

I. INTRODUCTION

Superalloy materials have been widely used in both commercial and military applications [14], [16] because of their excellent tensile strength and resistance to creep under high temperatures [21]. Such physical properties are mainly determined by the underlying micro-structures of superalloy samples, which are usually in the form of set of grains [14]. These grains are too small to be visible to human vision. In practice, high-performance electron microscopy is usually used to get the 2D surface image of the material sample [14]. To better identify the microstructures, various chemicals, like acids, may be applied to the material surface to highlight the grain boundaries. In addition, to achieve the underlying 3D grain structures, a serial-sectioning technique is usually applied to unveil the internal structure of the material to the microscopy [21]. Example image slices from a 2D serial section of a 3D superalloy material sample are shown in the first column of Figure 1 where each cell is a grain. In order to

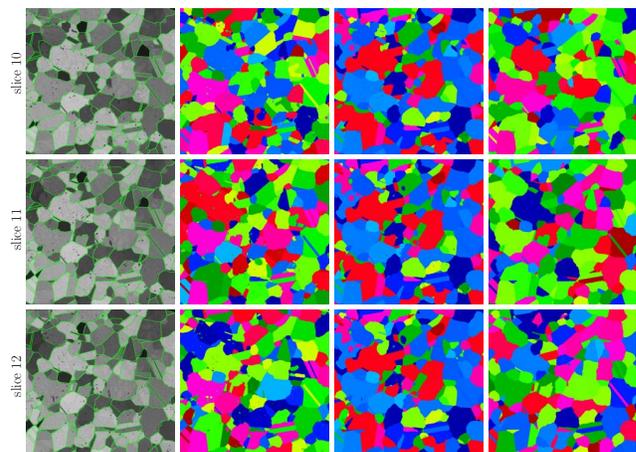


Figure 1. Microscopic grain images and segmentation. From left to right: original image slices with ground-truth grain boundaries, segmentation using a 2D method, segmentation using a 3D method, and segmentation using the proposed algorithm, respectively.

reveal grain structures from such 2D image slices, material scientists must manually annotate the grain boundaries on each of the 2D slices, and then correspond 2D grains across all the slices to reconstruct the 3D grain structure. This manual annotation process is tedious, time-consuming, and often prone to error, given a large number of grains and serial-sectioned slices in a high-resolution 3D superalloy image. This calls for efficient and effective automatic grain segmentation, which not only captures the grain boundaries accurately, but also completes quickly.

In principle, 2D image segmentation methods [1], [4], [13], [15], [18], [6], [5] can automate this process. However, these 2D methods usually perform segmentation on the 2D image slices without considering segmentation consistency between slices, and this may make it very difficult to correspond the 2D segments across slices for 3D reconstruction. The second column of Figure 1 shows 2D segmentation results on three consecutive image slices using the so-called *edge-weighted centroidal Voronoi tessellation*

(EWCVT) algorithm [18]. Clearly many grains are over-segmented/undersegmented in these slices and it is difficult to correspond them between slices. Another approach is to directly apply 3D image segmentation methods to the whole volume [10], [23], [20], [7], [17]. However, most of these 3D methods require a number of initial seeds in the 3D volume provided automatically, or by human interaction. Without any prior knowledge about the size of the grains, it is difficult to choose the proper location and number of seeds.

More recently, two automatic 3D EWCVT-based methods (MCEWCVT and CMEWCVT) [3], [2] were proposed for superalloy image segmentation. These two clustering methods perform 3D segmentation by minimizing an energy function which considers both voxel intensity similarity and the smoothness of the segmentation boundaries. During the energy minimization, these two 3D clustering methods enumerate every voxel and collect smoothness information around the neighborhood of each voxel, which leads to very high algorithmic complexity. Additionally, the L^∞ -norm used in these two methods makes the centroids difficult to calculate. Although these two EWCVT-based methods capture grain boundaries accurately, as claimed in [3], they typically require 10 hours to segment a 3D $671 \times 671 \times 170$ image; the same as we use in our later experiments in Section III. Furthermore, strong noise in the dense 3D image space is usually grouped into separate clusters as shown in the third column of Figure 1.

In order to efficiently segment 3D volumes, propagation-based 3D image segmentation has attracted much attention, especially for video processing. In [8], the authors propagate the result of previous video frame as the initialization for segmenting the current frame. After a modified active-contour based segmentation, the algorithm further merges and splits segments according to the partial shape matching across video frames. However, this algorithm is not applicable to grain image segmentation for two reasons: 1) The interslice resolution of a grain image is much lower than the intraslice resolution which causes the 2D shape of a grain to vary substantially when serial-sectioned by two consecutive slices, and 2) unlike video, which may contain structures with different shapes, many grains in superalloy images bear very similar shapes, and this may increase the ambiguity of the partial shape matching. In [22], the StreamGBH algorithm is proposed for segmenting streaming videos. StreamGBH segments a sequence of video frames by merging an over-segmentation on each frame, guided by the segmentation on the previous frame. Without specific constraints on boundary smoothness, StreamGBH usually generates highly fragmented and scattered segments, and is not suitable for grain segmentation.

By combining the advantages of the EWCVT-based and the propagation-based methods, we propose a modified EWCVT clustering algorithm which propagates inter-slice consistency constraints for accurate and fast segmentation of

3D grains in superalloy images. Specifically, the proposed algorithm performs a 2D-constrained EWCVT segmentation on each image slice using the segmentation of the previous slice as the initialization, and during the clustering process the stable grain structure of the previous slice is also preserved. On the first image slice, we use the segmentation result of the EWCVT algorithm as the initialization. This way, the proposed algorithm obtains a segmentation on the new image slice while simultaneously preserving the segment correspondence with the previous image slice.

The remainder of this paper is organized as follows, we first introduce the proposed method in Section II and then present the experimental results in Section III. Finally, Section IV concludes the paper.

II. THE EWCVT WITH PROPAGATION OF THE CONSISTENCY CONSTRAINT

We first revisit the classical CVT and EWCVT, and next define the constraints need to be satisfied for corresponding segmentations together on two adjacent image slices. Then we propose the modified EWCVT algorithm in which the correspondence constraints are well preserved during the segmentation process.

A. The classical CVT and EWCVT

An image slice I can be regarded as an intensity function u on a 2D domain. Since the pixels in the slice are indexed by integer tuples, we can treat u as a discrete function defined over a set of points with integer coordinates. Thus the domain of u for an image slice is an index set $D = \{(x, y) \mid x = 1, \dots, X, y = 1, \dots, Y\}$. Let $\mathbb{U} = \{u(x, y)\}_{(x, y) \in D}$ denote the set of intensity values of an image slice. Given $\mathcal{W} = \{w_l\}_{l=1}^L$ a set of distinct intensity levels, we can compute the corresponding Voronoi regions V_l in \mathbb{U} by

$$V_l = \{u(x, y) \in \mathbb{U} : |u(x, y) - w_l| \leq |u(x, y) - w_k| \text{ for } k = 1, \dots, L \text{ and } k \neq l\}, \quad (1)$$

where $|\cdot|$ denotes a predefined metric measure. The set of chosen intensities $\mathcal{W} = \{w_l\}_{l=1}^L$ are referred as the Voronoi generators. $\{V_l\}_{l=1}^L$ can be viewed as a special partition of \mathbb{U} because $\cup_{l=1}^L V_l = \mathbb{U}$ and $V_p \cap V_q = \emptyset$ for any pair of Voronoi regions.

Given a partition of \mathbb{U} , denoted as $\mathcal{V} = \{V_l\}_{l=1}^L$, we normally define the centroid (cluster center) of each V_l as $\bar{w}_l = \min_{w \in V_l} \sum_{u(x, y) \in V_l} |u(x, y) - w|^2$. We call the Voronoi tessellation $\{V_l\}_{l=1}^L$ a *Centroidal Voronoi Tessellation* (CVT) of \mathbb{U} if and only if

$$w_l = \bar{w}_l, \text{ for } l = 1, \dots, L.$$

Using the correspondence between pixel indexes and colors through u , we can easily construct a corresponding partition of the physical domain D of the image slice I . Let $\mathcal{D} = \{D_l\}_{l=1}^L$ be a partition of D , the CVT energy is defined as

$$\tilde{E}(\mathcal{W}; \mathcal{D}) = \sum_{l=1}^L \sum_{(x, y) \in D_l} |u(x, y) - w_l|^2 \quad (2)$$

The CVTs can be constructed by minimizing the CVT clustering energy.

In addition to considering the intensity similarity only, the Edge-Weighted Centroidal Voronoi Tessellation (EWCVT) clustering model also includes an edge energy term in order to smooth the segmentation boundary. The EWCVT energy is defined as

$$E(\mathcal{W}; \mathcal{D}) = \sum_{l=1}^L \sum_{(x,y) \in D_l} |u(x,y) - w_l|^2 + \lambda \sum_{(x,y) \in D} \sum_{(x',y') \in \mathbb{N}_\omega(x,y)} \chi_{(x,y)}(x',y'), \quad (3)$$

where $\mathbb{N}_\omega(x,y)$ denotes a circle neighboring area of the pixel (x,y) with radius ω , $\chi_{(x,y)}(x',y')$ is defined as

$$\chi_{(x,y)}(x',y') = \begin{cases} 1, & \text{if } \pi(x',y') \neq \pi(x,y), \\ 0, & \text{otherwise,} \end{cases}$$

where $\pi(x,y)$ is the cluster index function that tells the Voronoi region to which the pixel (x,y) belongs.

Based on Eq. (3), the corresponding *EWCVT distance* from a pixel (x,y) to a generator w_l can be defined as

$$dist((x,y), w_l) = \sqrt{|u(x,y) - w_l|^2 + 2\lambda \tilde{n}_l(x,y)}, \quad (4)$$

where $\tilde{n}_l(x,y) = |\mathbb{N}_\omega(x,y)| - n_l(x,y) - 1$ denotes the number of pixels within $\mathbb{N}_\omega(x,y) \setminus (D_l \cup (x,y))$. Thus we can construct the EWCVT by minimizing the EWCVT energy using the traditional CVT/*k*-means algorithms with the EWCVT distance.

The EWCVT segmentation result of an image slice can be represented as a set of simply connected segments $S = \{s_1, \dots, s_m\}$ where $\cup_{i=1}^m s_i = D$ and $s_i \cap s_j = \emptyset$ for any pair of segments. Remember that an edge-weighted Voronoi region D_l could be disconnected (i.e., in this case the Voronoi region contains more than one segments) and S often are constructed by extracting and relabeling all of the connected components in each of the edge-weighted Voronoi region $D_l \in \mathcal{D}$. In our 3D superalloy image segmentation problem, many grains may be assigned with the same Voronoi generator after the clustering of a slice. In order to guarantee the simple connectivity of each grain segment (i.e., no grain is completely contained in the interior of any of others), we follow the approach of the VCells method proposed in [19] which is based on the EWCVT model. The simple connectivity of individual segments in VCells is mainly achieved by only allowing cluster index transfer between neighbor segments, of pixels located on the boundaries of the physical segments with some topological constraints.

However, without further correspondence constraints, it is still hard to obtain consistent segmentation results and then correspond them between two adjacent image slices (see discussions in Section I). The resulting 3D superalloy image segmentation will contain plenty of truncated grains. In the proposed method, we use the segmentation of the previous

image as the initialization of the EWCVT algorithm, and we add consistency constraints during the clustering energy minimization process in order to propagate and preserve the stable segment structure contained in the initialization.

B. Stable segment structure and its propagation

Given two consequent image slices I^i and I^{i+1} , their segmentation results can be defined as $S^i = \{s_1^i, \dots, s_{m_i}^i\}$ and $S^{i+1} = \{s_1^{i+1}, \dots, s_{m_{i+1}}^{i+1}\}$ where m_i and m_{i+1} are the number of segments (grains) in I^i and I^{i+1} respectively. The segment structure of the segmentation S^i on the image slice I^i can be represented by a graph of segments in S^i , denoted as $G^i(\mathcal{V}^i, \mathcal{E}^i)$, where each vertex in \mathcal{V}^i is a segment and the edge weights in \mathcal{E}^i measure the strength of the adjacency of two neighbor segments (directly connected). Typically, given two segments, we use the number of pixels located on the boundary shared by them as their edge weight.

The stable segment structure of S^i on I^i can be defined as a connected subgraph $G_*^i(\mathcal{V}_*, \mathcal{E}_*^i)$ of G^i . Specifically, it holds that

$$\mathcal{V}_*^i = \left\{ s_p^i \in \mathcal{V}^i \mid |s_p^i| \geq \alpha \right\} \quad (5)$$

and

$$\mathcal{E}_*^i = \left\{ \mathcal{E}_{(p,q)}^i \in \mathcal{E}^i \mid \mathcal{E}_{(p,q)}^i \geq \beta, \quad s_p^i, s_q^i \in \mathcal{V}_*^i \right\}, \quad (6)$$

where the parameter $\alpha > 0$ is the minimal size of segments that are defined as stable ones, and the parameter $\beta > 0$ is the minimal length of boundaries that are stable. In our 3D superalloy image segmentation problem, the stable grain structure of S^i on I^i should be preserved in S^{i+1} on I^{i+1} . Unstable grains and their adjacency, caused by the difference of two consequent image slices, are determined by the image information on I^{i+1} .

We first use the segmentation S^i on the previous image slice I^i as the initialization, i.e., let $S^{i+1} = S^i$. Then by combining G_*^i with the image information I^{i+1} , we can construct the segment structure \tilde{G}_*^{i+1} on S^{i+1} that is invariant to the change of image information from I^i to I^{i+1} . The vertexes in \tilde{G}_*^{i+1} are the segments in S^{i+1} that are corresponding with stable segments in S^i , and the edges indicate corresponding stable segments' neighbor relationships. Thus \tilde{G}_*^{i+1} can be viewed as a propagation of the stable segment structure G_*^i from image slice I^i to I^{i+1} .

Specifically, we define the corresponding distance between a segment s_p^i on I^i and a segment s_q^{i+1} on I^{i+1} as

$$d(s_p^i, s_q^{i+1}) = \frac{|u^{i+1}(s_p^i) - u^{i+1}(s_q^{i+1})|}{|s_p^i \cap s_q^{i+1}|} \quad (7)$$

where $u^{i+1}(s_p^i)$ denotes the average intensity of pixels inside segment s_p^i on I^{i+1} :

$$u^{i+1}(s_p^i) = \frac{1}{|s_p^i|} \sum_{(x,y) \in s_p^i} u^{i+1}(x,y).$$

For each segment $s_p^i \in \mathcal{V}_*^i$, we find its nearest segment $s_q^{i+1} \in S^{i+1}$ and add s_q^{i+1} into $\tilde{\mathcal{V}}_*^{i+1}$, with respect to the distance defined in Eq. (7), i.e.,

$$s_q^{i+1} = \arg \min_{s_k^{i+1} \in S^{i+1}} d(s_p^i, s_k^{i+1}). \quad (8)$$

The intuition here is that two corresponding segments should be similar not only in the intensity space but also in the spatial domain. Otherwise the segment adjacency in G_*^i and \tilde{G}_*^{i+1} are not consistent. The edges $\tilde{\mathcal{E}}_*^{i+1}$ can be simply derived from G_*^i .

C. Algorithms for computing the EWCVT satisfying consistency constraint

1) Cluster initialization satisfying consistency constraint:

Given the segmentation S^i and the propagated stable segment structure G_*^i , we now compute a good initial clustering $\mathcal{D}^{i+1} = \{D_l\}_{l=1}^L$ of the image slice I^{i+1} for the iterative construction of the EWCVT, and this initial configuration also must satisfy the segment structure \tilde{G}_*^{i+1} inherited from G_*^i . This is done through a CVT/ k -means-type iterative process on S^{i+1} .

We treat each segment $s_q^{i+1} \in S^{i+1}$ as a point and define its value as $u^{i+1}(s_q^{i+1})$ i.e., the average intensity of pixels inside segment s_q^{i+1} . Then we can define a new intensity domain on the segments as $\mathbb{U}_{S^{i+1}} = \{u^{i+1}(s_q^{i+1}) \mid s_q^{i+1} \in S^{i+1}\}$. Let $\hat{\mathcal{D}} = \{\hat{D}_l\}_{l=1}^L$ be a partition of S^{i+1} into L clusters, then the CVT tessellation of S^{i+1} can be constructed based on the new average intensity according to the weighted CVT energy

$$\hat{E}(\mathcal{W}, \hat{\mathcal{D}}) = \sum_{l=1}^L \sum_{s_q^{i+1} \in \hat{D}_l} \left| u^{i+1}(s_q^{i+1}) - w_l \right|^2. \quad (9)$$

and its corresponding CVT distance is

$$\widehat{\text{dist}}(s_q^{i+1}, w_l) = \left| u^{i+1}(s_q^{i+1}) - w_l \right|. \quad (10)$$

Notice that two initially neighbor segments may be merged into a single segment if they are assigned to the same cluster. In this way the propagated stable segment structure \tilde{G}_*^{i+1} can not be preserved. In order to guarantee the resulting partition preserves the segment neighboring relationship defined in S^{i+1} , the clustering result should satisfy

$$\forall s_k^{i+1} \in \mathcal{N}_{s_q^{i+1}}, \quad \pi^{i+1}(s_q^{i+1}) \neq \pi^{i+1}(s_k^{i+1}), \quad (11)$$

where $\mathcal{N}_{s_q^{i+1}}$ denotes the neighbor segments of s_q^{i+1} and $\pi^{i+1}(s_q^{i+1})$ tells the index of the Voronoi region the segment s_q^{i+1} belongs to. During the classic CVT construction based on $\{u^{i+1}(s_q^{i+1}) \mid s_q^{i+1} \in S^{i+1}\}$, any new cluster assignment that violates Eq. (11) should be prevented. The whole cluster initialization process is described in Algorithm 1, which will be used as part of input for further computing the EWCVT with consistency constraint.

Algorithm 1 (Cluster Initialization Satisfying Consistency Constraint)

Inputs: The image slice I^{i+1} and u^{i+1}
 S^i : Segments of the image slice I^i
 π^i : The cluster index function of the image slice I^i
 L : Number of clusters
 $niter$: Number of iterations

- 0 **Initialization:** Create the stable segment structure G_*^i of S^i . Set $S^{i+1} = S^i$ and create \tilde{G}_*^{i+1} for S^{i+1} . Set $\pi^{i+1} = \pi^i$.
- 1 **FOR** $iter = 1, \dots, niter$
- 2 **FOR** $l = 1, \dots, L$
- 3 Compute the centroid

$$w_l = \frac{\sum_{s_q^{i+1} \in \hat{D}_l} |s_q^{i+1}| u^{i+1}(s_q^{i+1})}{\sum_{s_q^{i+1} \in \hat{D}_l} |s_q^{i+1}|}$$
- 4 **FOR** each $s_q^{i+1} \in S^{i+1}$
- 5 Find the nearest $w_k \in \{w_l\}_{l=1}^L$ to s_q^{i+1}
w.r.t. the distance function $\widehat{\text{dist}}(s_q^{i+1}, w_k)$
- 6 **IF** the relation (11) and \tilde{G}_*^{i+1} are satisfied
- 7 Set $\pi^{i+1}(s_q^{i+1}) = k$
- 8 **IF** there is no cluster change among S^{i+1}
- 9 Break

Output: The cluster index function π^{i+1} and $\{w_l\}_{l=1}^L$

2) Construction of EWCVT with consistency constraint:

The initialization through Algorithm 1 guarantees the initialized partition $\mathcal{D}^{i+1} = \{D_l\}_{l=1}^L$ (obtained through π^{i+1}) preserves the propagated segment structure \tilde{G}_*^{i+1} . Now we present a modified EWCVT clustering algorithm such that the propagated segment structure is preserved during the clustering process. Meanwhile new segments/grains can be identified using the image information. The basic idea is, during the EWCVT clustering process, 1) preventing any cluster assignment that breaks the propagated segment structure \tilde{G}_*^{i+1} ; and 2) adjusting the number of segments in S^{i+1} .

Specifically, we only consider pixels located at the boundaries of stable segments in $\tilde{\mathcal{V}}_*^{i+1}$ and inside unstable segments $S^{i+1} - \tilde{\mathcal{V}}_*^{i+1}$, denoted as $(x, y) \in \Omega$. Those pixels can only be assigned to a cluster which is physically connected to them, i.e., $\pi^{i+1}(x, y) \in \{\pi^{i+1}(\mathcal{N}_{(x,y)})\}$ where $\mathcal{N}_{(x,y)} = \{(x-1, y), (x+1, y), (x, y-1), (x, y+1)\}$. By checking the propagated stable segments adjacent relations defined in \tilde{G}_*^{i+1} , in each new cluster assignment, we only allow assignments that preserve the stable segments adjacency. After each assignment, if the clusters of the surrounding segments are different from the cluster of the center pixel, we identify this center pixel as a new segment/grain and add it into S^{i+1} . The whole process can be described in Algorithm 2.

Notice that, we propagate the previous segmentation information by using S^i as the initialization directly. Therefore

Algorithm 2 (EWCVT Construction Maintaining Consistency Constraint)

Inputs: The image slice I^{i+1} and u^{i+1}
 S^i : Segments of the image slice I^i
 π^i : The cluster index function of the image slice I^i
 L : Number of clusters
 $niter$: Number of iterations

- 0 **Initialization:** Run Algorithm 1.
- 1 **FOR** $iter = 1, \dots, niter$
- 2 Construct the set of candidate pixels for transferring, Ω from S^{i+1}
- 3 **FOR** each $(x, y) \in \Omega$
- 4 Find the nearest $w_k \in \{w_l \mid l \in \{\pi^{i+1}(\mathcal{N}_{(x,y)})\}\}$ to (x, y) w.r.t. the distance function $dist((x, y), w_k)$
- 5 **IF** no segments adjacency in \tilde{G}_*^{i+1} is violated
- 6 Set $\tilde{k} = \pi^{i+1}(x, y)$, $\pi^{i+1}(x, y) = \tilde{k}$
- 7 Update w_k and $w_{\tilde{k}}$
- 8 **IF** $\pi^{i+1}(x, y) \notin \pi^{i+1}(\mathcal{N}_{(x,y)})$
- 9 Add a new segment that contains (x, y) in S^{i+1}
- 10 **IF** there is no cluster change among Ω
- 11 Break

Outputs: The final cluster index function π^{i+1} and segmentation S^{i+1} of the image slice I^{i+1}

the segment indexes are consistent across image slices. Furthermore, new identified segments will be also included in the updated S^{i+1} as described in Algorithm 2. Finally, we can easily correspond segments across 2D image slices and then construct 3D segments/grains.

III. EXPERIMENTS

In this section, we first describe the test dataset, parameter settings, and evaluation criterion used in the experiments. Then we compare the proposed algorithm with several 2D/3D segmentation methods in terms of accuracy and running time. Finally, we discuss how the accumulated propagation error affects the segmentation accuracy of the proposed algorithm when segmenting a long sequence of 2D image slices.

The experiments are conducted on the IN100 dataset¹ which contains 170 sequential 2D image slices of a superalloy material sample. These image slices are obtained by photographing (using microscope) the top surface of a superalloy sample block during a top-to-bottom abrading process. Each slice in the IN100 dataset contains 4 grayscale images taken under different microscope configurations. For

¹Provided by our material scientist collaborator and can be downloaded at <http://www.bluequartz.net/Data/>.

Table II
 COMPARISON ON RUNNING TIMES OF 2D/3D/STREAMING SEGMENTATION METHODS.

Methods	MCEWCVT [3]	StreamGBH [22]	Proposed
Time (s)	37243.23	8808.7	7651.72

the proposed algorithm, we combine them into a single 4-channel image, analogous to typical RGB 3-channel images and use L^2 -norm as the distance metric. We do not suppress any channel, which is different from the L^∞ -norm used in MCEWCVT [3]. For comparison methods that cannot handle multi-channel images directly, we first apply such methods to each of the four grayscale images independently. We then combine these independent segmentations into an additional fifth segmentation, either using the logic OR operation (for solid boundary segmentation algorithms, e.g. the NormalizedCuts [15] method), or assigning the maximum probability boundary (pb) value to each pixel (e.g. for the gPb [1] method). For each such comparison method, we report the result (out of the above five results) which yields the best performance.

The proposed algorithm is implemented in C/C++. For comparison methods, we used implementations published by the respective authors. All the experiments are conducted on a Linux workstation with Quad-Core 3.20GHz Intel Xeon CPU and 6G memory.

For the proposed algorithm, there are two key parameters that can be tuned: the radius ω of the local smoothness region and the edge weight λ . We performed a grid search over this parameter space using the whole dataset and selected $\omega = 4$ and $\lambda = 30$ which achieved the best performance. Additionally, we set the number of clusters in the color space as $k = 40$. For the remaining parameters, the minimum size of stable segments α and the minimum length of stable boundaries β , are set according to the average grain size: $\alpha = 80$ and $\beta = 5$.

We use the boundary-overlap criterion suggested in the Berkeley segmentation benchmark [11] to evaluate the results quantitatively. Specifically, detected segmentation boundaries are compared with the ground-truth boundaries to calculate precision, recall and the F-score

$$\text{F-score} = 2 \cdot \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

As mentioned before, in the proposed algorithm we use the segmentation obtained by the EWCVT algorithm on the first slice as an initialization, propagating it to segment the remaining 169 slices sequentially. We evaluate the segmentation accuracy on all 170 slices.

Comparison with 2D segmentation methods – We compare the proposed algorithm with several automatic 2D segmentation methods, including MeanShift [4], the graph-based (GraphBased) method of [9], SRM [13], gPb [1],

Table I
 QUANTITATIVE COMPARISON OF 2D/3D/STREAMING SEGMENTATION METHODS.

Methods		Precision	Recall	F-score
2D Methods	EWCVT [18]	0.838385	0.962131	0.896005
	MeanShift [4]	0.911927	0.844106	0.876707
	GraphBased [9]	0.704163	0.928424	0.800891
	SRM [13]	0.81018	0.800006	0.805061
	gPb [1]	0.828988	0.866076	0.847126
3D/Streaming Methods	NormalizedCuts [15]	0.736609	0.691646	0.71342
	MCEWCVT [3]	0.845894	0.927918	0.885009
	3D Levelset [20]	0.739025	0.581001	0.650554
	3D Watershed [12]	0.864594	0.589135	0.700767
	StreamGBH [22]	0.454185	0.792653	0.577479
Proposed		0.957377	0.896125	0.925739

and NormalizedCuts [15]. Additionally, we also compare the proposed algorithm with the original EWCVT [18] method on 2D image slices. The NormalizedCuts method requires the number of desired segments, which we set to the number of ground truth segments in this dataset. Parameters of other methods are set either to their default values, or the setting that provides the best performance from a coarse grid search.

The quantitative results are shown in the middle six rows of Table I, from which we can see that the original 2D EWCVT already achieves a very good performance with an F-score of 89.6%, while the proposed EWCVT-based propagation algorithm further significantly improves the segmentation accuracy by another 3% to 92.5%. This indicates that, aside from the excellent performance of the EWCVT clustering algorithm, the consistency constraints in the proposed algorithm indeed boost the performance further.

Comparison with 3D/Streaming segmentation methods

– We compare the proposed algorithm with the MCEWCVT method in [3], the 3D levelset method [20], the 3D watershed method [12] and the StreamGBH method in [22]. We select StreamGBH because it is a propagation-based method and it achieves state-of-the-art performance on video segmentation tasks [22], which is similar to the 3D grain image segmentation application. Another related work is the method in [8], however the authors have not released the implementation of this method.

For the MCEWCVT method, we use the parameter configuration provided in the original paper [3]. For the 3D levelset method, the number of seeds are the same as the number of grains contained in the ground truth segmentation, and the seeds are evenly distributed in the 3D space. For StreamGBH, we set the number of consecutive image slices involved in the propagation (i.e., the parameter “range”) to be 2, which is equivalent to only using the previous image slice to do propagation, as in the proposed algorithm. After performing a coarse grid search, other parameters of StreamGBH are set to be: $n_{hie} = 10$, $c = 60$, $c_{reg} = 200$, $min = 100$ and $\sigma = 0.8$.

The quantitative results are shown in the bottom two rows

of Table I, the proposed algorithm clearly outperforms the comparison methods. MCEWCVT underperforms because it groups strong noise in the dense 3D image space into separate clusters. StreamGBH shows lower performance because it lacks structure consistency constraints and boundary smoothness in the propagation, leading to isolated and jagged boundaries.

Moreover, in order to demonstrate the computational efficiency of the proposed algorithm, we also compare its running time with that of the MCEWCVT and the StreamGBH methods. All these methods are implemented in C/C++ and their running times are shown in the last column of Table II. The proposed algorithm exhibits the fastest running time, with a speedup of $5\times$ compared with the MCEWCVT method.

Qualitative segmentation results on three consecutive image slices of both 2D image segmentation methods and 3D/Streaming segmentation methods are shown in Figure 2 and Figure 3 respectively. We can clearly see that the segmentation from the proposed algorithm aligns with grain boundaries much better than the segmentation from the comparison methods. For 2D comparison methods, without considering inter-slice correspondence, they often produce isolated fragments inside a grain. In contrast, the proposed algorithm maintains the consistency and correspondence among grains across image slices. For the 3D methods, compared with the proposed algorithm, they are sensitive to strong noises in the dense 3D space. For StreamGBH, without considering the structure consistency and the boundary smoothness, its results contain many fragments along grain boundaries.

Propagation error – Intuitively, the segmentation error accumulates when the segmentation propagates through a large number of slices. Therefore, we may expect a monotonic decrease of the segmentation accuracy with more steps of propagation. However, as shown in the top panel of Figure 4, the segmentation accuracy only oscillates occasionally during the propagation. The main reason is that, aside from the propagation of structural consistency constraints, the proposed algorithm also includes a EWCVT clustering process

to refine the segmentation using the image information when processing a new slice. Also note that the F-score on the first image is relatively low. This is due to the EWCVT result being used as the initialization. Even from this imperfect initialization, the proposed method is able to improve the performance by considering both structure and image information in propagation. Similarly, one slice of corrupted or highly noisy image, as shown in the bottom panel of Figure 4, has little effect on the segmentation performance on the other slices because of the use of structure constraints in the proposed algorithm.

IV. CONCLUSIONS

In this paper, we proposed an Edge-Weighted Centroid Voronoi Tessellation based algorithm that can propagate structural consistency constraints from slice to slice, which is used to automatically segment 3D grain images. As a volumetric segmentation algorithm, the proposed algorithm can automatically extract grain structures on all the slices, starting from the segmentation on the first slice, which can be constructed by any automatic 2D segmentation method. The proposed propagation-based algorithm is able to: 1) segment a large number of superalloy image slices efficiently, 2) preserve structural consistency across slices, and 3) easily correspond the segments across slices. We conducted experiments on a 3D superalloy image dataset with 170 image slices. Both qualitative and quantitative results indicate that the proposed algorithm outperforms the comparison methods and is robust even when propagated through a large number of slices, and in the presence of strong noise and corruption.

REFERENCES

- [1] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *TPAMI*, 33:898–916, 2011.
- [2] Y. Cao, L. Ju, and S. Wang. Grain segmentation of 3d superalloy images using multichannel ewcvt under human annotation constraints. In *ECCV*, 2012.
- [3] Y. Cao, L. Ju, Q. Zou, C. Qu, and S. Wang. A multichannel edge-weighted centroidal voronoi tessellation algorithm for 3d super-alloy image segmentation. In *CVPR*, 2011.
- [4] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *TPAMI*, 24:603–619, 2002.
- [5] C. Couprie, L. Grady, L. Najman, and H. Talbot. Power watersheds: A new image segmentation framework extending graph cuts, random walker and optimal spanning forest. In *ICCV*, 2009.
- [6] T. Cour, F. Benezit, and J. Shi. Spectral segmentation with multiscale graph decomposition. In *CVPR*, 2005.
- [7] A. B. Derek Magee and E. Berry. Level set methods for the 3d segmentation of ct images of abdominal. In *Proceedings of the Medical Image Understanding and Analysis*, 2001.
- [8] M. Donoser, M. Urschler, H. Riemenschneider, and H. Bischof. Highly consistent sequential segmentation. In *Proceedings of the 17th Scandinavian conference on Image analysis*, 2011.

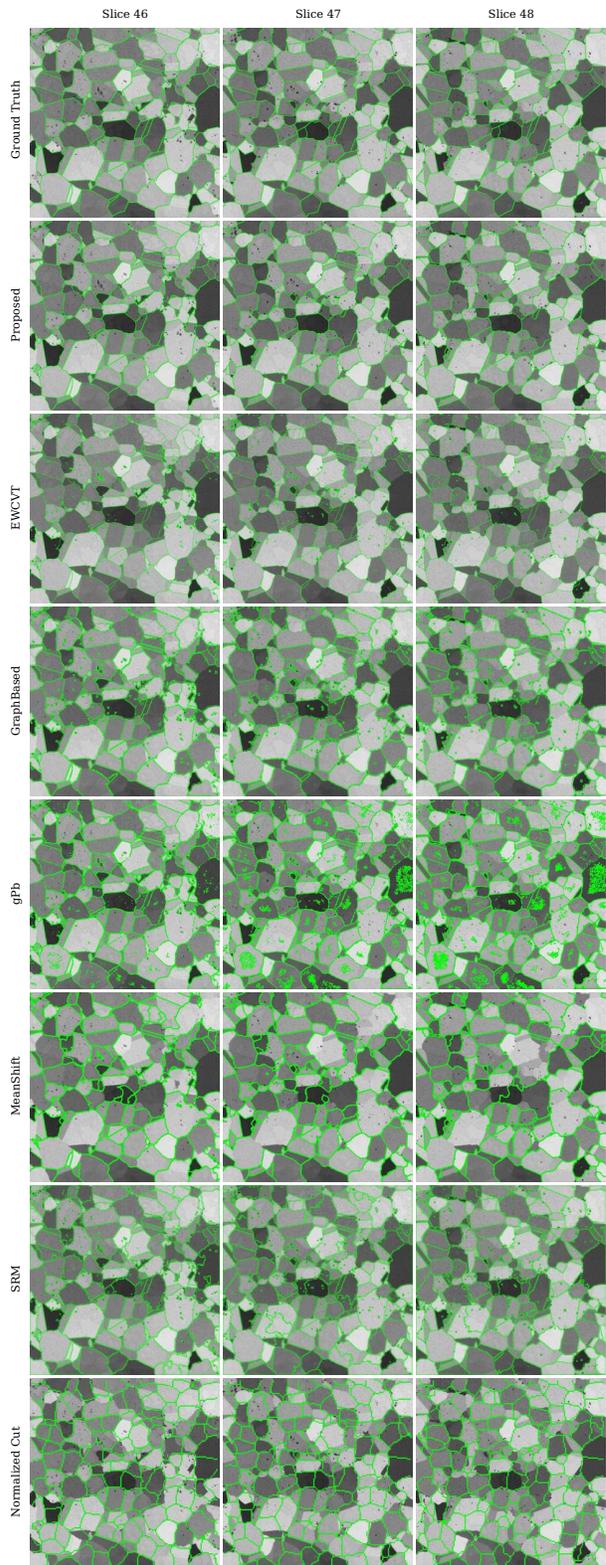


Figure 2. Qualitative comparisons of the seven 2D image segmentation methods (Proposed, EWCVT, GraphBased, gPb, MeanShift, SRM and NormalizedCuts) on three consecutive image slices from the IN100 dataset. From the top to the bottom are the original image with ground truth boundaries and results of different methods.

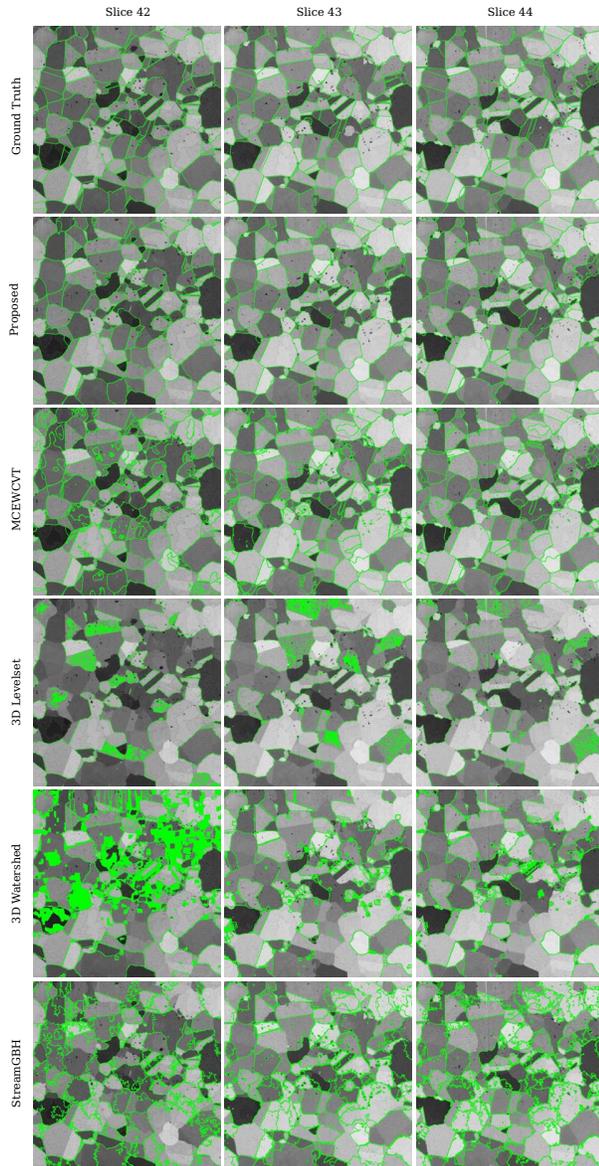


Figure 3. Qualitative comparisons of the five 3D/Streaming image segmentation methods (Proposed, MCEWCVT, 3D Levelset, 3D Watershed and StreamGBH) on three consecutive image slices from the IN100 dataset. From the top to the bottom are the original image with ground truth boundaries and results of different methods.

- [9] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59:167–181, 2004.
- [10] L. Grady. Random walks for image segmentation. *TPAMI*, 28:1768–1783, 2006.
- [11] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, 2001.
- [12] F. Meyer. Topographic distance and watershed lines. *Signal Processing*, 38:113–125, 1994.
- [13] R. Nock and F. Nielsen. Statistical region merging. *TPAMI*, 26:1452–1458, 2004.

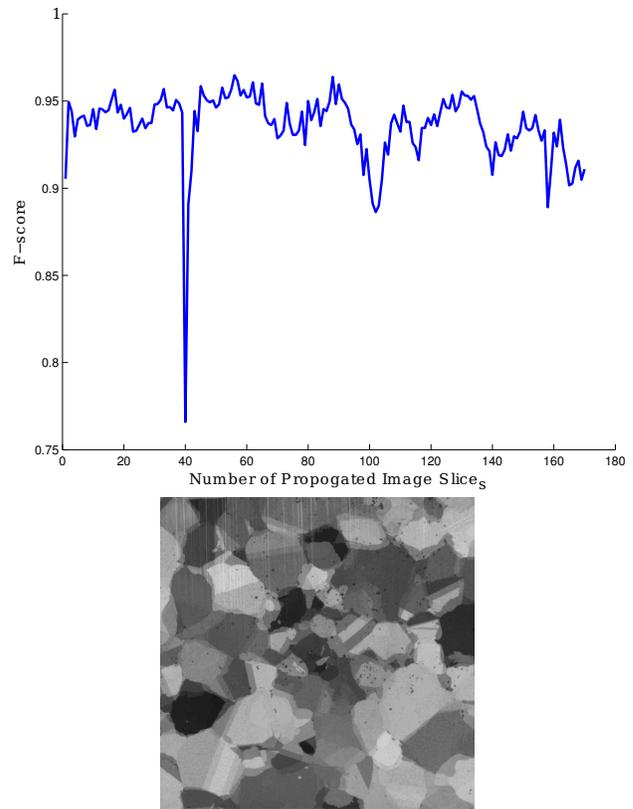


Figure 4. The F-score on each slice (top) during propagation. The sharp drop is caused by a corrupted slice (bottom) in the original dataset.

- [14] R. C. Reed. *The Superalloys Fundamentals and Applications*. Cambridge Press, 2001.
- [15] J. Shi and J. Malik. Normalized cuts and image segmentation. *TPAMI*, 22:888–905, 2000.
- [16] P. Vander Voort, Warmuth and Szirmae. *Metallography: Past, Present, and Future, 75th Anniversary Volume*. ASTM, 1993.
- [17] F. Vivodtzev, G.-P. Bonneau, L. Linsen, B. Hamann, K. I. Joy, and B. A. Olshausen. Hierarchical isosurface segmentation based on discrete curvature. In *Proceedings of the Symposium on Data visualisation*, 2003.
- [18] J. Wang, L. Ju, and X. Wang. An edge-weighted centroidal voronoi tessellation model for image segmentation. *TIP*, 18:1844–1858, 2009.
- [19] J. Wang and X. Wang. Vcells: Simple and efficient super-pixels using edge-weighted centroidal voronoi tessellations. *TPAMI*, 34(6):1241–1247, 2012.
- [20] R. T. Whitaker. A level-set approach to 3d reconstruction from range data. *IJCV*, 29:203–231, 1998.
- [21] D. G. R. William D. Callister. *Materials Science and Engineering: An Introduction*. Wiley, 2010.
- [22] C. Xu, C. Xiong, and J. J. Corso. Streaming hierarchical video segmentation. In *ECCV*, 2012.
- [23] R. R. M. Yu-Kun Lai, Shi-Min Hu and P. L. Rosin. Rapid and effective segmentation of 3d models using randomwalks. *Computer Aided Geometric Design*, 26:665–679, 2009.