

Calculation of grain boundary normals directly from 3D microstructure images

This content has been downloaded from IOPscience. Please scroll down to see the full text.

2015 Modelling Simul. Mater. Sci. Eng. 23 035005

(<http://iopscience.iop.org/0965-0393/23/3/035005>)

View [the table of contents for this issue](#), or go to the [journal homepage](#) for more

Download details:

IP Address: 192.12.184.7

This content was downloaded on 11/03/2015 at 15:32

Please note that [terms and conditions apply](#).

Calculation of grain boundary normals directly from 3D microstructure images

E J Lieberman^{1,2}, A D Rollett¹, R A Lebensohn² and E M Kober³

¹ Department of Materials Science and Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA

² Materials Science and Technology Division, Los Alamos National Laboratory, MS G755, Los Alamos, NM 87455, USA

³ Theoretical Division, Los Alamos National Laboratory, MS B214, Los Alamos, NM 87455, USA

E-mail: emk@lanl.gov

Received 10 October 2014, revised 17 January 2015

Accepted for publication 10 February 2015

Published 11 March 2015



CrossMark

Abstract

The determination of grain boundary normals is an integral part of the characterization of grain boundaries in polycrystalline materials. These normal vectors are difficult to quantify due to the discretized nature of available microstructure characterization techniques. The most common method to determine grain boundary normals is by generating a surface mesh from an image of the microstructure, but this process can be slow, and is subject to smoothing issues. A new technique is proposed, utilizing first order Cartesian moments of binary indicator functions, to determine grain boundary normals directly from a voxelized microstructure image. To validate the accuracy of this technique, the surface normals obtained by the proposed method are compared to those generated by a surface meshing algorithm. Specifically, the local divergence between the surface normals obtained by different variants of the proposed technique and those generated from a surface mesh of a synthetic microstructure constructed using a marching cubes algorithm followed by Laplacian smoothing is quantified. Next, surface normals obtained with the proposed method from a measured 3D microstructure image of a Ni polycrystal are used to generate grain boundary character distributions (GBCD) for $\Sigma 3$ and $\Sigma 9$ boundaries, and compared to the GBCD generated using a surface mesh obtained from the same image. The results show that the proposed technique is an efficient and accurate method to determine voxelized fields of grain boundary normals.

Keywords: grain boundaries, microstructure, image analysis, moment analysis

(Some figures may appear in colour only in the online journal)

1. Introduction

Grain boundaries (GBs) are the interfaces between two disoriented crystals and, because they delimit changes in the orientation-dependent local properties of the solid, they have considerable influence over the properties of polycrystalline materials. For example, many forms of material failure, such as voids and cracks, tend to initiate and propagate along GBs [1, 2]. The field of Grain Boundary Engineering has arisen to manipulate these interfaces to improve material properties [3, 4], and in order to do so full characterization of GBs is necessary. GBs can be characterized at the mesoscale by five parameters. Three of these parameters specify the misorientation, which is the transformation that brings the crystal lattices of the grains on either side of the boundary into coincidence. The other two specify the unit vector that is locally normal to the GB plane. These components are combined to produce a five-dimensional (5D) *grain boundary character* (GBC). The GBC is assumed to be sufficient to specify GB properties such as energy and mobility. The misorientation components of the GBC are rather easily calculated in terms of the orientation of neighboring points in space, which can be measured by techniques such as electron backscatter diffraction (EBSD). The GB plane normals, however, are rather more difficult to acquire since they must be inferred from a set of points distributed in three dimensions. The GB plane normals are needed, for example, to calculate the dihedral angles around triple junctions, which in turn are used to calculate the GB energies and create grain boundary energy distributions [5, 6]. One method of visually representing the GBC is to utilize grain boundary character distributions (GBCD), which is a stereographic projection of the interface orientations for a particular misorientation of interest [7, 8].

Some of the reasons as to why GB normals are difficult to obtain are explored here. First, as mentioned above, three-dimensional (3D) information is required and this is expensive and time-consuming to acquire. The most common methods for getting 3D microstructure information are serial-sectioning combined with EBSD measurements, which is a destructive technique [9], and the recently introduced non-destructive high-energy diffraction microscopy (HEDM) [10]. Both methods result in a series of two-dimensional (2D) images that can then be aligned to form a 3D image. However, the GB normals are still not easily obtained from this stack of 2D images, due to the lack of information on the actual boundary inclination. The microstructure image obtained from EBSD or HEDM is similar to any other digital image; it consists of a grid of discrete points that does not explicitly include surfaces that would be needed to provide information on GB topology. At best, boundaries are known to lie between neighboring sets of points that belong to two different grains as shown in figure 1. However, the details in between are not directly known.

Requiring actual 3D information can be avoided by using stereological methods to derive general plane normal distributions [11, 12]. However, information on individual boundaries cannot be obtained using stereology. Surface reconstruction, i.e. the method of generating surface information from a set of discrete points representing a shape, is required in order to define the missing GB information in 3D data. This constitutes a difficult problem and many different techniques have been proposed aiming at solving it. For example, Dey and Goswami [13] developed a technique for surface reconstruction utilizing intersecting Delaunay balls to determine the surfaces. Hoppe *et al.* [14] and Mitra and Nguyen [15] generated surface information by fitting tangent planes to some specified number of local points. A similar method was developed by Ivasishin [16] who combined it with a 3D Monte-Carlo model. A technique by Moore and Warren [17] utilized polynomial fitting to develop polyhedral surfaces. Among surface reconstruction techniques, the most common and widely accepted way of approximating the GB surfaces is to generate a surface mesh from voxelized microstructure

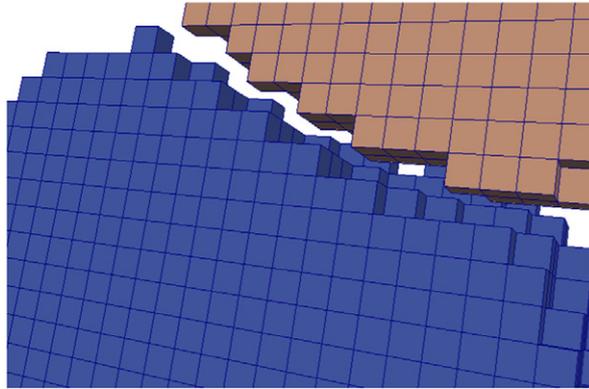


Figure 1. An image for two grains, where each corner of the cubic grid of lines represents a measured orientation point in the material. If all eight vertices of the cube agree in their orientation, the cube can be identified as belonging to that grain and is colored appropriately. If the eight vertices are not all in agreement, the volume must be associated with the GB and is left transparent for this illustration.

images. Different variants of this methodology have been proposed. One of the most used algorithms for developing a surface mesh from image data is the ‘marching cubes’ method [18], which is used in many software programs including the one utilized in this work, known as Dream.3D [19].

As with general surface reconstruction, the most common technique for GB characterization is based upon developing an explicit surface mesh. However, there are also other methods that do not explicitly generate this mesh but still calculate surface information [7, 8, 20]. The methodology proposed by Ivasishin *et al* [16] is one of the few cases of GB normals calculated directly from 3D integer grid-based data. However, this technique was not validated against experimental microstructures. Another example of GB information inferred from voxelized images is that of Chandross *et al* [21], who, however, focused on obtaining dihedral angles at triple lines.

In this work, we develop a new method based on the gradients of binary indicator fields determined from the voxelized image data. This is discussed in section 2. In section 3, we discuss the validation of this technique first with synthetic microstructures and compare the surface normals determined by the new technique with those calculated from a surface mesh with triangular elements, obtained using Dream.3D. Based on the analysis of these synthetic cases, the method is further utilized to obtain the GBCD of a pure nickel 3D microstructure image obtained by HEDM. These results are compared to the GBCD previously determined by Hefferan [22–24] for the same microstructure, but using a different surface reconstruction technique that instead utilizes the Computational Geometry Algorithms Library (CGAL) package. CGAL is software that is used to model surfaces for many applications ranging from Materials Science to Geology to Biology [25–27], to mention a few examples. Its algorithm is based on Delaunay refinement.

Our overall interest is to develop an algorithm for calculating GB normals directly from discretized data, with a particular focus on regular grids. The subsequent application of the proposed technique will be to combine it with novel and very efficient voxel-based mesoscale simulation tools [28–30], specifically conceived to model micromechanical behavior directly from images obtained from emerging 3D characterization techniques. These spectral

formulations are *not* based on finite element analysis and therefore do not require the generation of surface meshes. However, they would still need surface normal information, e.g. to evaluate quantities such as tractions and displacements at GBs. Consequently, avoiding the generation of surface meshes and instead directly using voxelized data to compute a voxelized field of GB normals is computationally advantageous, in the context of these emerging image-based formulations.

2. Methods

A technique is presented here that can approximate GB plane normals directly from 3D microstructure data through the use of first order Cartesian moments. The equation for finding general Cartesian moments of arbitrary order [31, 32] is

$$M_{opq} = \iiint_S w(\mathbf{r}) x^o y^p z^q f(\mathbf{r}) d\mathbf{r} = \sum_{(i,j,k) \in S} w(\mathbf{r}) x_i^o y_j^p z_k^q f(\mathbf{r}). \quad (1)$$

The order of a moment is determined by the sum of its indexes, represented here by the non-negative integers o , p and q . The function $w(\mathbf{r})$ is a weighting function that depends upon the position $\mathbf{r} = (x_i, y_j, z_k)$ within the volume of interest S . The weight is often taken to be simply 1 for all points within S , but more generally it can be specified to have a range of values depending on the particular application. The function $f(\mathbf{r})$ is a scalar field that will be used as the indicator function. Some examples of scalar fields that can be used as indicator functions are composition, misorientation angle or von Mises equivalent stress. In this work, we use a *binary* indicator function. Such functions, which can adopt values of 0 or 1, are used to establish which points are part of the object of interest and which are not. For our present application, we know to which grain each voxel belongs, so we define an integer function $h(\mathbf{r})$ which returns the grain number for that location. The indicator function $f(\mathbf{r})$ is then defined as the delta function between the desired grain number m and the actual grain number $h(\mathbf{r})$: $f(\mathbf{r}) = \delta(m, h(\mathbf{r}))$. The function is 1 if the grain numbers match, and zero otherwise, and this would select out the volume elements belonging to grain m . Typically, the origin of the coordinate frame is selected as the center of position of the object of interest and the resulting evaluations define the *central* moments. Moment invariants can then be calculated from the central Cartesian moments of such binary indicator function, and these in turn can be used to characterize the shape of the object in a variety of ways [31, 32].

A related moment formulation can be used to define the local derivatives of the indicator function and provides a formal way of defining the edges and surfaces of an object [33–35]. Here, in particular, we are interested in calculating the surface normal of the grains, which are parallel to the gradient of the grain number indicator function [36–38]. Because the data is present on a regularly spaced grid, the gradients can be defined by the least-squares formulations given in (2) [39–41].

$$\begin{aligned} \left. \frac{\partial f}{\partial x} \right|_{(0,0,0)} &= \frac{\sum_{(i,j,k) \in S} w(\mathbf{r}) f(\mathbf{r}) x_i}{\sum_{(i,j,k) \in S} w(\mathbf{r}) x_i^2} = \frac{\sum_{(i,j,k) \in S} w(\mathbf{r}) f(\mathbf{r}) i}{\Delta x \sum_{(i,j,k) \in S} w(\mathbf{r}) i^2} \\ \left. \frac{\partial f}{\partial y} \right|_{(0,0,0)} &= \frac{\sum_{(i,j,k) \in S} w(\mathbf{r}) f(\mathbf{r}) y_j}{\sum_{(i,j,k) \in S} w(\mathbf{r}) y_j^2} = \frac{\sum_{(i,j,k) \in S} w(\mathbf{r}) f(\mathbf{r}) j}{\Delta y \sum_{(i,j,k) \in S} w(\mathbf{r}) j^2} \\ \left. \frac{\partial f}{\partial z} \right|_{(0,0,0)} &= \frac{\sum_{(i,j,k) \in S} w(\mathbf{r}) f(\mathbf{r}) z_k}{\sum_{(i,j,k) \in S} w(\mathbf{r}) z_k^2} = \frac{\sum_{(i,j,k) \in S} w(\mathbf{r}) f(\mathbf{r}) k}{\Delta z \sum_{(i,j,k) \in S} w(\mathbf{r}) k^2}. \end{aligned} \quad (2)$$

Here, in order to simplify the notation, we have shifted the origin to the center of the voxel of interest and define it to have the (i,j,k) indices of $(0,0,0)$. The summations over (i,j,k)

are within a region S which is used to evaluate the derivative. That defines a neighborhood about that central voxel, where the indices define the positions of the other voxels relative to it. The specifics of the weighting function $w(\mathbf{r})$ will be discussed in further detail below. The first set of equalities above are written in terms of the physical distances between points, and the second set of equalities transforms those to the differences in indices of the points. This takes advantage of the regularly spaced grid, where the distance vector from the central point is simply defined as:

$$\mathbf{r} = (x_i, y_j, z_k) = (i\Delta x, j\Delta y, k\Delta z). \quad (3)$$

Here, we emphasize that it is not required that the grid spacings ($\Delta x, \Delta y, \Delta z$) in the three different directions be equivalent. This is especially convenient as both the experimental data (EBSD, HEDM) and results of the Fourier transform-based simulations often have this attribute of different spacings in the three directions. Computationally, we have utilized the second set of equalities in (2).

The numerators of the first set of equalities in (2) are seen to be the first order Cartesian moments within the neighborhood, and that these are proportional to the components of the gradient of $f(\mathbf{r})$, with the proportionality depending upon the definition of $w(\mathbf{r})$ [33, 34]. We note that the use of this formulation requires that $w(\mathbf{r})$ and the definition of the neighborhood S be symmetric about the origin in addition to the data being present on a regularly spaced grid. More complex formulations must be utilized if these conditions are not met⁴. Our formulations can be further simplified by treating the indices in the three directions equivalently. This can be achieved by requiring the weighting function to be simply a function of the sum of the squares of the indices, reducing its dependence from a vector to a scalar quantity. A similar constraint on the definition of the neighborhood S is adopted as shown in (4) and (5).

$$w(\mathbf{r}) \rightarrow w(i^2 + j^2 + k^2) \quad (4)$$

$$(i, j, k) \in S \forall [i^2 + j^2 + k^2 \leq s^2]. \quad (5)$$

Under these conditions, the summations in the three denominators in the second equalities of (2) become equivalent, and they can be replaced by a single quantity W .

This also enforces an equivalent statistical treatment of the gradient evaluation that is independent of the direction and specific axes orientation. The resulting formulations can then be written as in (6), where the scaled first order moments μ_{opq} are now defined.

$$\begin{aligned} \mu_{100} &= \frac{1}{\Delta x} \sum_{(i,j,k) \in S} w(\mathbf{r}) f(\mathbf{r}) i = W \left. \frac{\partial f}{\partial x} \right|_{(0,0,0)} \\ \mu_{010} &= \frac{1}{\Delta y} \sum_{(i,j,k) \in S} w(\mathbf{r}) f(\mathbf{r}) j = W \left. \frac{\partial f}{\partial y} \right|_{(0,0,0)} \\ \mu_{001} &= \frac{1}{\Delta z} \sum_{(i,j,k) \in S} w(\mathbf{r}) f(\mathbf{r}) k = W \left. \frac{\partial f}{\partial z} \right|_{(0,0,0)}. \end{aligned} \quad (6)$$

⁴ The most general formulation for the derivative evaluation is

$$\left. \frac{\partial f}{\partial x} \right|_{\bar{x}} = \frac{[\sum w(\mathbf{r}) f(\mathbf{r}) x_i] - \bar{x} [\sum w(\mathbf{r}) f(\mathbf{r})]}{[\sum w(\mathbf{r}) x_i^2] - \bar{x}^2 [\sum w(\mathbf{r})]}$$

where \bar{x} is the weighted average value of the coordinates used in the evaluation:

$$\bar{x} = \frac{\sum w(\mathbf{r}) x_i}{\sum w(\mathbf{r})}.$$

If one considers that higher order curvatures effects are present, this evaluation of the first derivative is only precise at the point \bar{x} . A Taylor series expansion, including corrections from the second (and possibly higher) derivatives would be required to obtain a precise value at $x = 0$. The symmetry requirements on $w(\mathbf{r})$ and the even spacing of the x_i s are one method of insuring that $\bar{x} = 0$. This eliminates the second terms in the numerator and denominator and generates the simpler formulations used in the text [39].

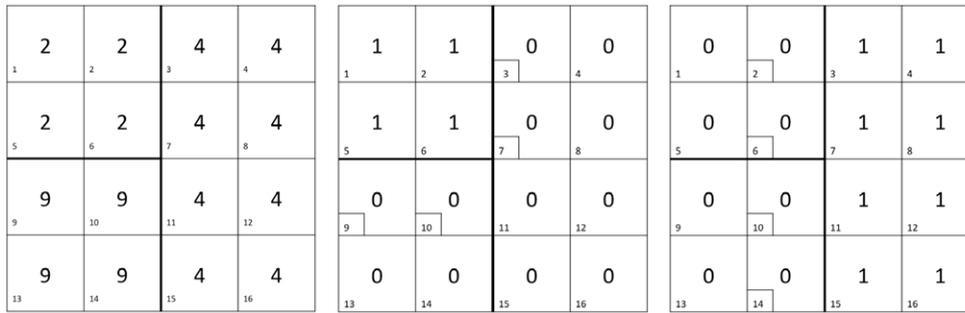


Figure 2. Example of the two indicator functions generated at point 10, which is the point in grain ID 9 that is at a triple point. The left figure shows the grid with the grain ID number in the center, and the point number (small font) in the bottom left corner of each cell. The middle and right figures show the two different indicator functions associated with point number 10, the first being between grains 9 and 2, and the second being between grains 9 and 4. The point numbers highlighted by boxes are those that would use the kind of shape function presented in the image, showing that points that are not directly associated with triple lines are also affected by this, depending on the neighborhood size.

As mentioned above, here we are interested in the direction of surface normal and do not need to know the magnitude of the gradient. Consequently, these vector components can then be normalized by the vector magnitude to generate a unit vector whose direction is the surface normal. Relationships for higher order derivatives could be similarly defined from the higher order moments to evaluate the grain surface curvature and triple junction characteristics, and those will be discussed in subsequent work.

Here, we just utilize first order Cartesian moments and in three dimensions these have indexes of 100, 010 and 001 for the x , y and z directions, respectively. The origin of the calculation volume is located at the center of the voxel found to be part of the surface of the shape, defining $(i, j, k) = (0, 0, 0)$. Then the gradient components represent a vector that is in the same direction as the normal vector to that surface. Here, because we know the identities of the grains to which each voxel belongs, we designate the surface voxels as simply being those for which at least one of the six face-sharing neighbor voxels belongs to a different grain. For evaluations centered on voxels located internally to the grain, small gradients that point towards the surface would arise if the voxel is sufficiently close to the surface. This is of significance only if the grain identities and surface points are not known *a priori*, but does not impact the current study.

In the present case of analyzing grain structure, the binary indicator function requires special consideration since there is the possibility of a point having nearest-neighbor points belonging to more than one different grain. Such points are designated *triple points* in the case of a total of three grains, or *quad points* in the case of four grains. Thus, to properly describe the surface at and near these points, the binary indicator function is modified such that the points that are in the same central grain m are assigned a value of 0, the points in the one special neighboring grain n hold a value of 1, and all other points are also assigned a value of 0. Therefore, we modify the definition of the indicator function as follows: for defining the surface normal of a cell belong to a grain of type m with respect to a second grain of type n , $f(\mathbf{r}) = \delta(n, h(\mathbf{r}))$. Therefore for each point with g unique grain numbers among its nearest neighbors, there will be $g-1$ unique binary indicator functions, as shown in figure 2.

By having multiple binary indicator functions at a single triple point, we ensure that each vector calculated represents the surface between two grains. The points other than point 10 above that are highlighted by boxes show how regular GB voxels near triple points will only be influenced by the single neighboring grain with no contribution from the nearby but not adjacent third grain.

The three variants of the moment calculation that we will test are primarily different in terms of the weighting function $w(\mathbf{r})$ utilized in (2) and (6). The first variant studied here is inspired by techniques used in the general image processing literature, which utilize a Gaussian weighting (GW) function or *kernel*, $w(\mathbf{r}) = \exp[-(i^2 + j^2 + k^2)/\lambda^2]$, to effectively smooth the discrete data to a continuous function [34, 35, 42]. The derivatives of that smoothed function can then be readily calculated and used to identify edges, corners and similar features. Similar calculation has been performed with the Deriche filter [33], another function often used to smooth discrete data. For the Gaussian kernel, the value of λ defines the radius within which the data has significant weighting (i.e. the weight is $<1/e$ outside of that radius). To define the computational neighborhood S as described in (5), we use a value of s determined for when the weighting function decreases below a somewhat arbitrary value of 10^{-5} . Beyond that radius, we set $w(\mathbf{r}) = 0$. Hence, the values of λ and s are interrelated: $s = (\lambda/2)\ln(10^{-5})$. The values of λ that are tested in this work are $\sqrt{3}$, $\sqrt{6}$ and $\sqrt{9}$, which result in s values of 5.88, 8.48 and 10.18 respectively. Of particular significance for the GW function is that there exists a general lower limit of $\lambda = \sqrt{2}$ which generates a smooth function from regularly spaced discrete data [34, 42], and that neither adds nor subtracts extrema (minima, maxima, inflection points) to the raw discrete data. Somewhat larger values then add a level of smoothing to the data, which is desirable for data containing noise or uncertainties. Smaller values of λ could be used to define the local derivatives at the grid points themselves, but these could not be used for general points in space. Because there is a level of noise/uncertainty in this data, the comparison will start at $\sqrt{3}$, which was sufficient to filter that uncertainty, but did not noticeably remove any extrema. For the rest of this work this variant will be referred to as the GW variant.

The second variant treats every point within some specified cut-off radius, l , as having equal weighting. That is, $w(\mathbf{r}) = 1$ for $(i^2 + j^2 + k^2) \leq l^2$ and $w(\mathbf{r}) = 0$ for $(i^2 + j^2 + k^2) > l^2$. This method will be tested with values of l being $\sqrt{3}$, $\sqrt{6}$ and $\sqrt{9}$. Because of this hard-cut on the radius, the neighborhood S can be defined to have this same radius, $s = l$. A comparison of the difference in weighting between these two variants is shown in figure 3. Both of these approaches strive to define an spherically symmetric neighborhood over which to define the moments/gradients. Many earlier approaches are based on cubic neighborhoods, which add some bias for alignment of the gradients along the axis directions [34, 35, 42]. The GW method deprecates the contribution of more distant neighbor cells with respect to the primary nearest neighbors as a function of distance, enhancing the locality of the measure despite the larger neighborhood. The second variant treats all included points equally, smoothing the data more broadly within that footprint. For the rest of this work this second variant will be referred to as the non-scaling, equal weighing (NEW) variant. By increasing the values of (λ, l) and increasing the volume included in the evaluation, one basically biases the results towards smoother and flatter surfaces, while decreasing those values enhances errors due to noise and the discrete nature of the data. The unknown factor is how much smoothing is required to suppress those errors without smoothing the true structure too far towards a planar surface.

The third variant also treats every point within some cut-off radius l as having equal weighting, but the value of l is allowed to vary. Specifically, the value of l is determined as being the distance to the nearest triple junction or quad point, but its value cannot go below a fixed minimum value, l_{\min} . The value of this minimum will be varied for the different test

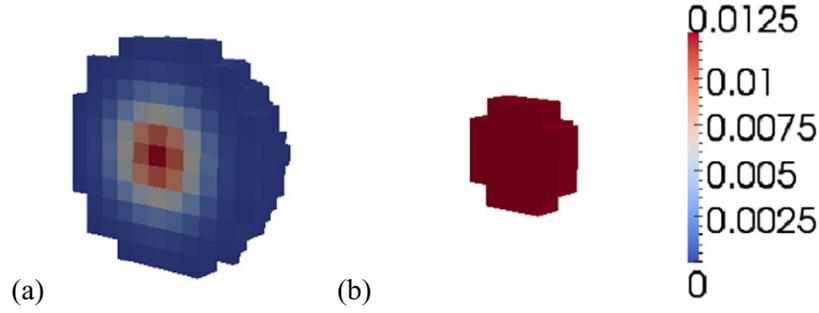


Figure 3. Normalized weighting factor relative to a central point using (a) GW with $\lambda = \sqrt{6}$ and (b) NEW with $l = \sqrt{6}$. A 3D perspective is shown, with a slice drawn through the central cell.

cases and is set to values of $\sqrt{3}$, $\sqrt{6}$ and $\sqrt{9}$ so that these results can be compared to the second variant. The purpose of the adaptable, scaling neighborhood is to smooth out the larger boundary areas while preserving accuracy near the surface discontinuities (i.e. triple junctions). For the rest of this work, this variant will be referred to as the scaling, equal-weighting (SEW) variant. All distance values for these three variants are evaluated and the results compared against an independent calculation of surface normals.

To highlight the effect of the size of the neighborhood, figure 4 shows the divergence angle between calculated normals and true normals for a sphere and a cube using different variants. The divergence angle is defined as the angle that exists between the surface normals generated by this technique and the surface normals they are being compared against. The equation to calculate the divergence angle is

$$\theta_{\text{div}} = \cos^{-1} \left(\frac{\vec{v}_T \cdot \vec{v}_C}{\|\vec{v}_T\| \|\vec{v}_C\|} \right). \quad (7)$$

Here, \vec{v}_T and \vec{v}_C are the two surface normals being compared. For the first sphere (figure 4(a)), the NEW variant is used as the method of smoothing with $l = \sqrt{3}$. These results highlight the issue of discretization (i.e. stair-stepping); by representing a smooth sphere as a collection of cubes, the approximate surface is inherently rough, especially where it does not align well with the mesh, and errors are made in the surface normal calculations. For the value of $l = \sqrt{3}$, the result is an average divergence angle of $\sim 21^\circ$. The result can be improved on by utilizing a larger value of l to achieve greater smoothing. For the second sphere (figure 4(b)), l is set to half the radius of the sphere, which results in a dramatically smaller average divergence angle of $\sim 1^\circ$. However, the penalty for using such a large footprint is illustrated for the first cube (figure 4(c)). Here, the NEW variant is used with l set to half the cube edge length, and the average divergence angle is found to be $\sim 14^\circ$, where there is an excessive rounding of the cube corners. For the second cube (figure 4(d)), the third variant is used with the edges of the cube treated as triple points, which results in greater accuracy close to the edges of the cube. Setting the value of l_{min} to $\sqrt{3}$, the average divergence angle decreases to $\sim 1^\circ$.

These test cases emphasize that one should keep the neighborhood as small as possible in order to avoid these corner-rounding effects, albeit that smaller neighborhoods can increase the level of error arising from stair-steps. The SEW variant was constructed as a compromise between these factors. However, the location of the edges/triple junction must be known to use the third variant of the technique because they are utilized to determine l_{min} and such information may not always be available.

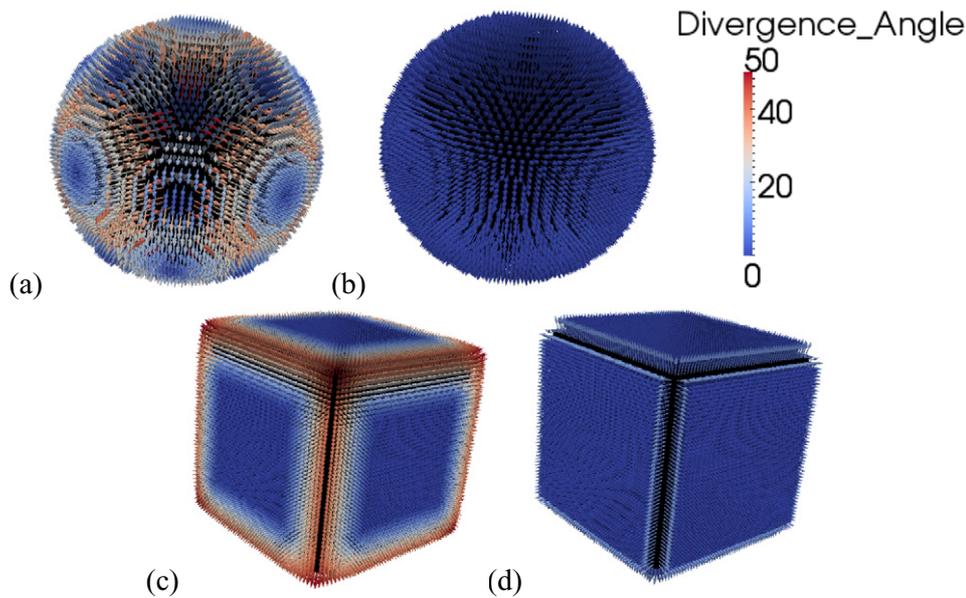


Figure 4. Surface normals for (a) a sphere using the NEW variant and l equal to $\sqrt{3}$, (b) a sphere using the second variant and l equal to half the radius, (c) a cube using the NEW variant and l equal to half the edge length and (d) a cube using the SEW variant and a minimum l_{\min} of $\sqrt{3}$. All vector arrows colored by divergence angle using the same color scale as shown. The cubes show how scaling limits the smoothing of sharp edges.

3. Results and discussion

3.1. Comparison with normals to a surface mesh

One method to evaluate these approaches is to generate a surface mesh of a microstructure image and compare the surface normals from the mesh to those found by the proposed technique for the same image. To do this, the synthetic microstructure shown in figure 5 is used. It was generated using MBuilder [43, 44] and it has 860 grains discretized onto a $128 \times 128 \times 128$ voxel grid.

A triangular surface mesh for this microstructure is generated using the Dream.3D software package with a marching cubes algorithm and smoothed with a Laplacian smoothing algorithm for 40 iterations. Some conversion steps are needed in order to obtain a one-to-one comparison between the surface normals of the mesh and those of the proposed technique. First, all the nearby nodes to a given GB voxel are identified, then the surface normals of the triangle elements which include those nodes are averaged with a weighting based on the triangle area. In particular the triangle elements used in the averaging are only those that are associated with the same GB number pair as those of the GB voxel. Thus if the GB voxel is on a triple line, then two vectors from the triangle mesh will be generated for both possible grain number pairs.

The divergence angle (equation (7)) between this approximated surface normal from the mesh and the surface normal calculated by our techniques represents a measure of how different the two sets of normal vectors are, with a value of 0° indicating complete agreement between them. Here, we use this angle as a measure of the accuracy of the technique with the aim being to find conditions that minimize the angles. For each of the nine cases described in section 2,

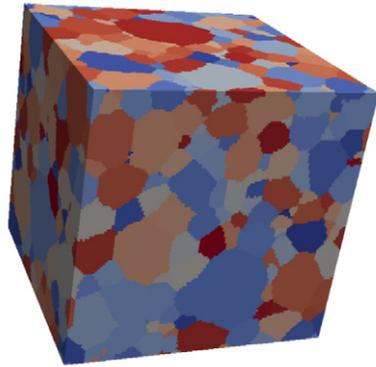


Figure 5. The synthetic microstructure used to validate the GB normal technique.

we determined the divergence angle for each GB surface normal calculated from the image. The probability distribution of the divergence angles for each test case is shown in figure 6, with the results split between GB voxels with a distance to the nearest triple point of one or less, and those with a distance greater than one. We make this distinction because there are greater uncertainties about the surface normals near triple junctions, due to both resolution related issues as well as difficulties that both our current techniques and surface meshing techniques might have in that vicinity.

These probability distributions show that, for each model variant, one can identify a preferred setting that results in the lowest average divergence angle for voxels that have a triple point distance greater than one. We also note that these lowest average divergence angles are fairly close to each other. These conditions are $\lambda = \sqrt{3}$ for the GW variant, $l = \sqrt{6}$ for the NEW, and $l_{\min} = \sqrt{6}$ for the SEW variants. Also, we observe that the distributions for voxels that have a triple line distance less than or equal to one are reasonably insensitive to the choice of technique used, although the divergence angle is generally larger for points on triple lines than the other points. For completeness, we also characterized $\lambda = \sqrt{2}$ for the GW variant, but those results were of lower quality than for $\lambda = \sqrt{3}$.

The GB normal vectors generated from the mesh and from the SEW variant for a particular grain are shown in figure 7, along with the underlying triangular mesh. The images show that both cases reflect relatively smooth surfaces away from the triple junctions. However, for the GB normals approximated from the mesh, there is great variation in the normals at the triple junctions. This is likely due to irregularities in the mesh smoothing in these areas, which can be seen in the center image of figure 7. These irregularities are reflected in the higher divergence angles measured on voxels in these areas. From these results we can conclude that, at least for this microstructure, all three variants of the technique, for at least one value of (λ, l, l_{\min}) , are reasonably accurate at determining GB normal vectors when compared to those from the standard method based on triangular surface meshes.

3.2. GBCD comparison

Next, we compare our normal vector approximations to those obtained from a state-of-the-art meshing algorithm applied to an experimentally measured 3D microstructure. The comparison is statistical in that we compare GB normal distributions rather than point-by-point values as above. This comparison is carried out for two specific misorientations, thus

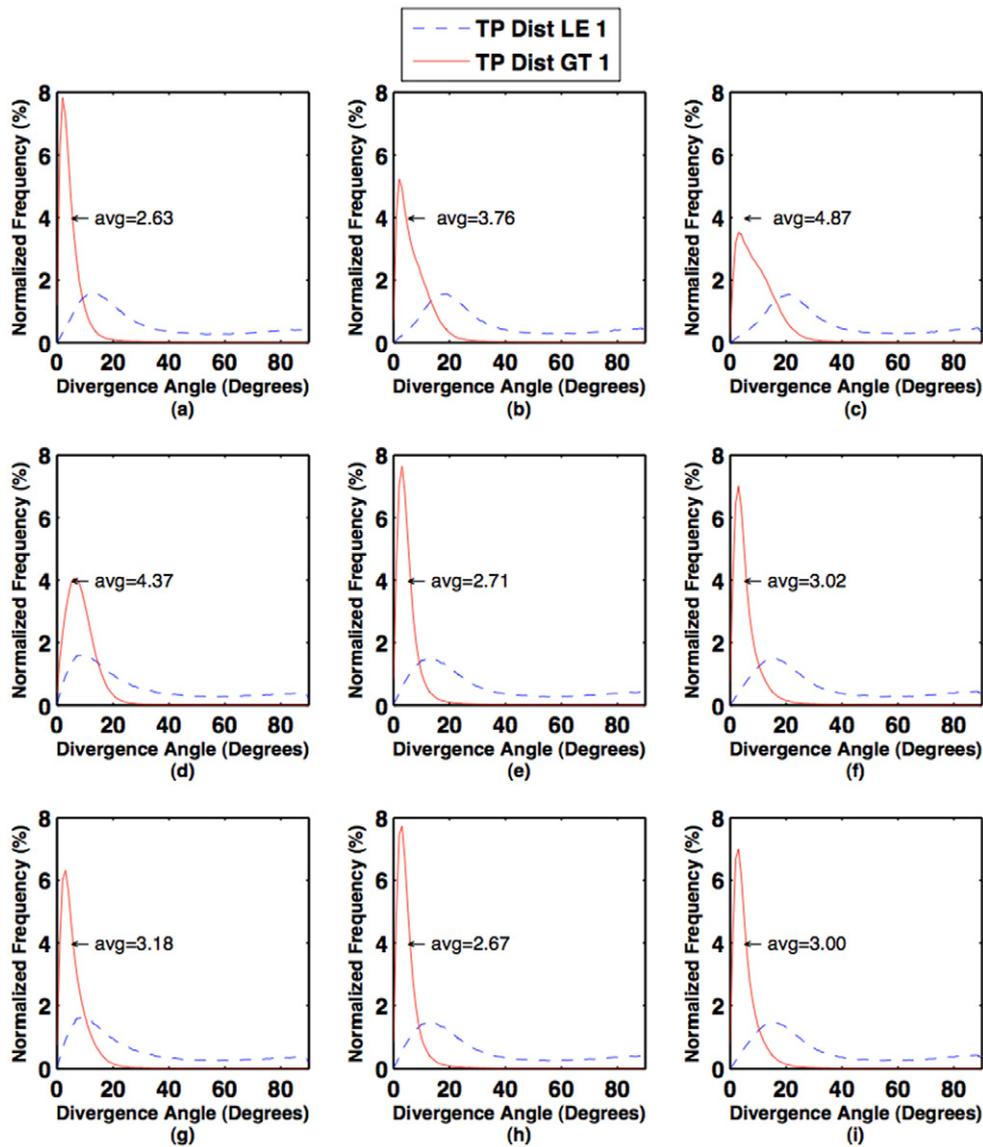


Figure 6. Probability distributions of the nine test cases, with the data separated into voxels with a triple point distance of one or less (blue) and voxels with a triple point distance greater than one (red). The average divergence angle of the red curve is displayed on each graph. The GW variant is represented by (a), (b) and (c) for $\lambda = \sqrt{3}$, $\sqrt{6}$ and $\sqrt{9}$ respectively. The NEW variant is represented by (d), (e) and (f) for $l = \sqrt{3}$, $\sqrt{6}$ and $\sqrt{9}$ respectively. The SEW variant is represented by (g), (h) and (i) for $l_{\min} = \sqrt{3}$, $\sqrt{6}$ and $\sqrt{9}$ respectively.

displaying two-dimensional subspaces of the complete, five-parameter GBCD. The normals are shown in crystallographic coordinates, with each normal appearing twice, once for each of the neighboring crystals.

The sample used is pure nickel and the measurement was performed by Hefferan *et al* [22–24] using HEDM [10, 45]. The HEDM reconstruction yields a set of triangular voxels

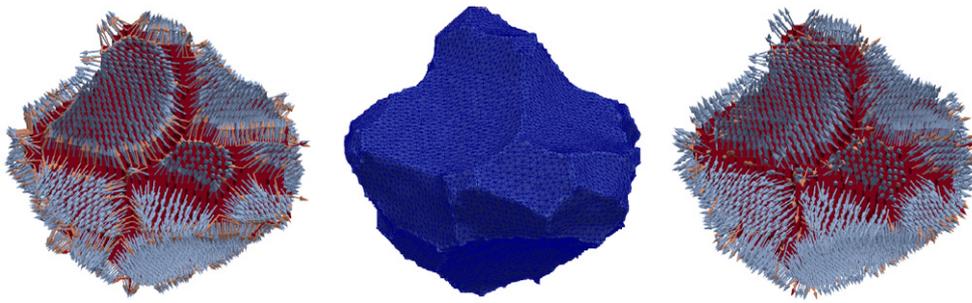


Figure 7. GB normals for a particular grain are shown for the SEW model variant with $l_{\min} = \sqrt{6}$ (left) and for the normals from the mesh (right), with the mesh itself also shown (center). The vectors are colored by the number of nearest-neighbor grains: blue = 1, orange = 2, red = 3.

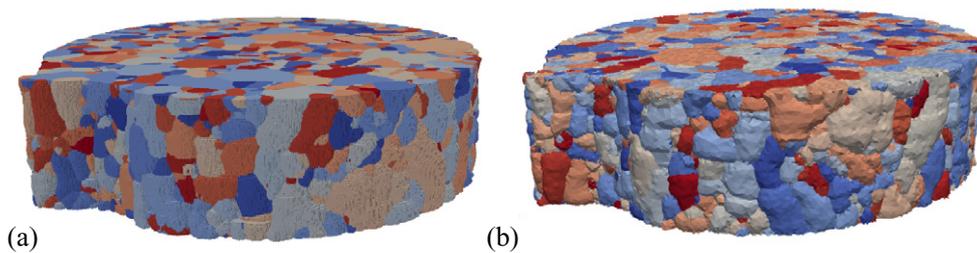


Figure 8. (a) Pure nickel microstructure from [22–24] has a resolution of $1000 \times 1000 \times 71$ points with spacing of $1.2 \times 1.2 \times 4 \mu\text{m}$. (b) Surface mesh generated for this image. In both cases, the false color is based on randomly assigned grain number ID's.

(in this case $1.2 \mu\text{m}$ side length equilateral triangles) in each of a set of 71 measured 2D sections that are spaced in $4 \mu\text{m}$ intervals perpendicular to the section planes. This structure was interpolated onto a cubic lattice, as shown in figure 8(a).

The GBs were meshed using a weighted Delaunay triangulation method developed by Li [46]. It utilizes algorithms available in the Computational Geometry Algorithms Library (CGAL) package^[47] with adaptations influenced by Boltcheva *et al* [48] to preserve important features, such as triple lines and quad points. The resulting mesh of this Ni microstructure is shown in figure 8(b) Using the meshed boundaries, Hefferan [23, 24] displayed the normal distributions for two specific crystallographic misorientations, denoted as the $\Sigma 3$ (60° rotation around the $\langle 111 \rangle$ axis) and $\Sigma 9$ (38.94° rotation around the $\langle 110 \rangle$ axis) boundaries. To generate the GBCD, the GB surface normals are transformed into the crystallographic reference frame of their respective grains and then represented as a stereographic projection of the rotated vectors, with only the GB points that have the misorientations of interest being included. Before rendering the vectors as a stereographic projection, the vectors are first transformed into spherical coordinates and then binned by 10° intervals. The bins are then normalized such that they are expressed as multiples of random density. The values of (λ, l, l_{\min}) found in section 3 that resulted in the greatest accuracy for each variation of the technique are used to generate GBCD from the HEDM images. The $\Sigma 3$ GBCDs generated from the surface meshing and the three variants of the technique are shown in figure 9.

What we see from these GBCD is that each model variant shows a very similar distribution shape but the primary difference is in the spread and peak of the distributions. The GBCD that

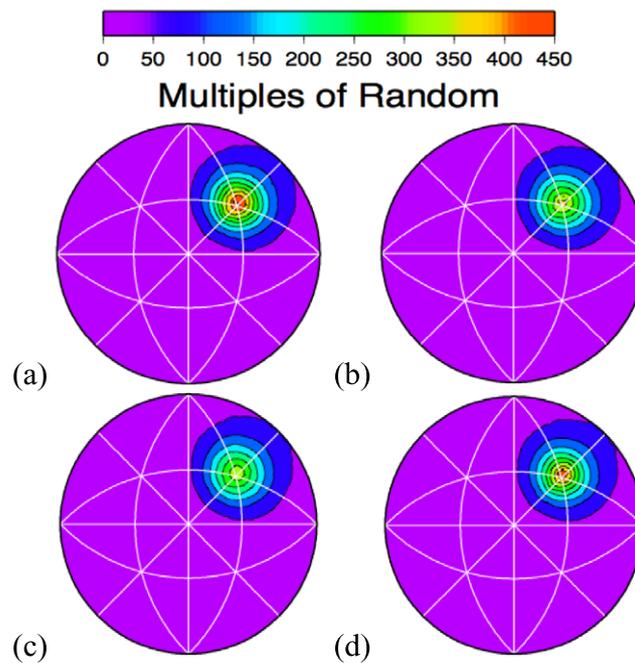


Figure 9. $\Sigma 3$ GBCD (a) from [23, 24], (b) the GW variant of the proposed model with $\lambda = \sqrt{3}$, (c) the second variant (NEW) with $l = \sqrt{6}$ and (d) the SEW variant with $l_{\min} = \sqrt{6}$. The strong peak in the upper right quadrant is associated with coherent twin boundaries, which are very common in most annealed fcc metals such as the Ni studied here.

is closest to Hefferan's $\Sigma 3$ baseline, in terms of peak value, is obtained with the SEW variant, followed by the GW variant with the NEW variant being the least accurate. To quantify this comparison, figure 10 shows these same projections but colored by the value of the difference between with respect to Hefferan's GBCD.

The largest absolute difference calculated in each case is 104.5 MRD for the GW variant, 143.6 MRD for the NEW variant and 50.9 MRD for the SEW variant; these results are also shown in table 1. In particular, the largest difference for the SEW variant is not located at the standard $\Sigma 3$ GBCD peak, as it is for the other two.

The $\Sigma 9$ GBCDs generated from the surface meshing and the three versions of the technique are shown in figure 11. Again the different variants of the technique show general agreement on the shape of the distribution but here there are more differences when compared to the $\Sigma 9$ baseline in that the peaks do not exactly line up, though they are all on the same $\langle 110 \rangle$ tilt profile, and the secondary shapes in the distributions are also slightly different but occur in the same positions in the plot. Again, the differences between the GBCD from Hefferan and the GBCD from this technique are shown in figure 12.

The largest absolute difference calculated in each case is 2.03 MRD for the GW variant, 2.80 MRD for the NEW variant and 2.16 MRD for the SEW variant. These results are also shown in table 1. The NEW variant is the most different from Hefferan's GBCD and is also farther from the other two variants, which are themselves quite similar in their differences compared to Hefferan's. Overall, the $\Sigma 9$ GBCD from the technique have greater relative differences to Hefferan's $\Sigma 9$ GBCD, than for the $\Sigma 3$ GBCD, though the magnitudes are smaller.

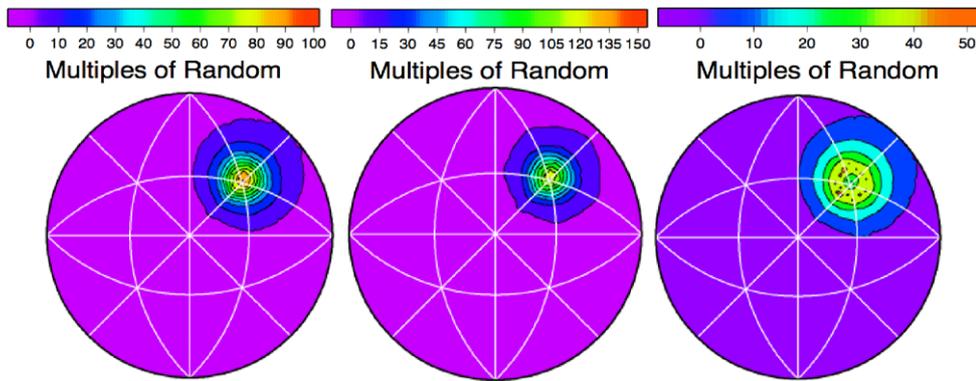


Figure 10. Difference in MRD values for $\Sigma 3$ boundaries between Hefferan's GBCD and the GBCD from (left) the GW variant, (center) the NEW variant and (right) the SEW variant.

Table 1. Peak Difference in MRD between the GBCD from Hefferan and the GBCD generated by the three variants for both $\Sigma 3$ and $\Sigma 9$ boundaries.

	Difference in $\Sigma 3$ GBCD (MRD)	Difference in $\Sigma 9$ GBCD (MRD)
GEW	104.5	2.03
NEW	143.6	2.80
SEW	50.9	2.16

In general, the GBCD obtained from the NEW variant exhibits the best agreement with Hefferan's. The explanation for why the NEW variant is favored here while in section 3.1 all three had similar accuracy is that the resolution of the HEDM sample (71×10^6 voxels) is greater than that of the synthetic microstructure ($\sim 2 \times 10^6$ voxels) and this particular variant has a consistent smoothing effect with increasing resolution while the other two are indifferent to resolution changes.

4. Conclusion

The proposed technique based on the use of first order Cartesian moments to calculate a voxelized field of grain boundary normals has been shown to compare favorably to normals found using surface meshing. When locally comparing the grain boundary normals obtained by the proposed technique to those from a triangular mesh, the greatest differences are found near the triple junctions and quad points, regions which are known to have inconsistencies with the smoothing due to most algorithms allowing triple junction nodes less freedom of positioning than other grain boundary nodes. Elsewhere the differences between the mesh normals and the normals generated by the technique are on average less than 3° . Differences between the different variants of the technique appear when tested on the higher resolution HEDM sample. The scaling, equal-weighting (SEW) variant is the one having the most success at replicating the GBCDs produced by Hefferan [23], which is reasonable considering that this variant is the only one to scale with resolution changes. This allows the smoothing to increase with increasing resolution and match the results of surface mesh smoothing that is indifferent to resolution. Thus the scaling, equal-weighting (SEW) variant is the best choice if one assumes that the

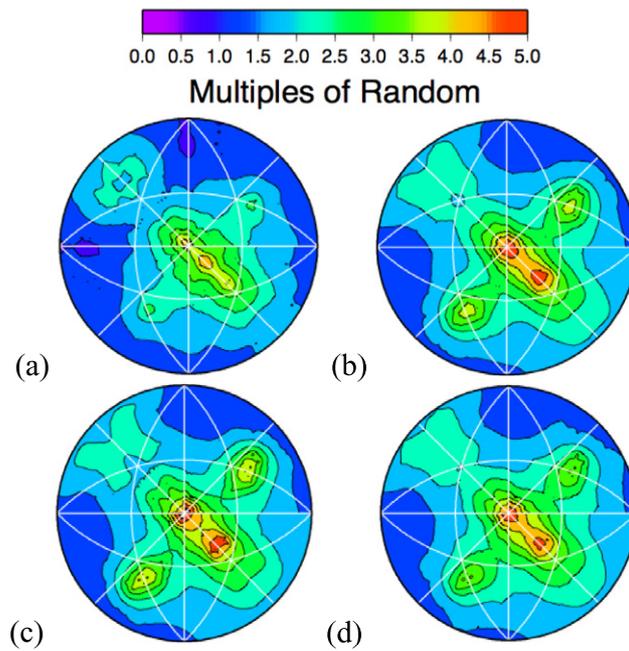


Figure 11. Σ_9 GBCD (a) from [23, 24], (b) the first variant (GW) of the proposed model with $\lambda = \sqrt{3}$, (c) the NEW variant with $l = \sqrt{6}$ and (d) the SEW variant with $l_{\min} = \sqrt{6}$.

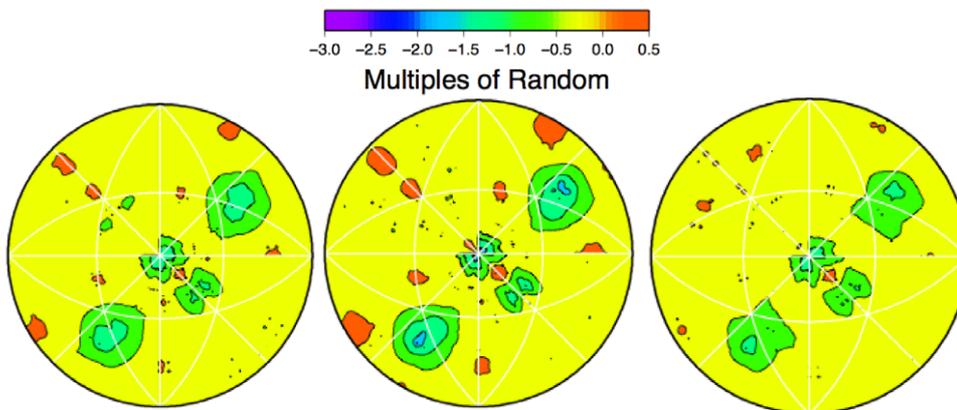


Figure 12. Difference in MRD values for Σ_9 boundaries between Hefferan’s GBCD and the GBCD from (left) the first variant (GW), (center) the second variant (NEW) and (right) the third variant (SEW).

boundaries being considered are smooth. However, if the above does not hold or the triple junction or other surface edge information is not known, then the Gaussian weighting (GW) variant, which had the next best success at replicating the GBCD, would be a viable option as well. Potential uses of the technique include estimating triple junction dihedral angles and incorporation in crystal plasticity models of grain boundary effects on mechanical responses,

such as slip transmission across the boundary [49] or by calculating surface tractions. We note that all of the methods exhibited higher divergence near the triple lines in the first surface mesh comparison in part because of a combination of an effective high curvature at the corners and the inability to recognize the transition between two different neighboring grains, which should give rise to a discontinuity in this simple definition of a surface normal. The addition of higher order moments/derivatives should improve the descriptions near the triple junctions. This aspect is currently being explored.

Acknowledgments

This work was supported by Los Alamos National Laboratory's Directed Research and Development (LDRD-DR Project 20140114DR) and the Institute for Materials Science. We thank C M Hefferan, S F Li, J Lind and R M Suter for sharing the Ni data set and for helpful discussions and assistance with interpretation. The data were collected at the Advanced Photon Source, which is supported by the US Department of Energy, Office of Science, Office of Basic Energy Sciences under contract number DE-AC02-06CH11357.

References

- [1] Fensin S J *et al* 2014 Effect of loading direction on grain boundary failure under shock loading *Acta Mater.* **64** 113–22
- [2] Lin P, Palumbo G, Erb U and Aust K T 1995 Influence of grain boundary character distribution on sensitization and intergranular corrosion of alloy 600 *Scr. Metall. Mater.* **33** 1387–92
- [3] Randle V 2004 Twinning-related grain boundary engineering *Acta Mater.* **52** 4067–81
- [4] Randle V, Rohrer G S, Miller H M, Coleman M and Owen G T 2008 Five-parameter grain boundary distribution of commercially grain boundary engineered nickel and copper *Acta Mater.* **56** 2363–73
- [5] Smith C S 1948 Introduction to grains, phases, and interfaces—an interpretation of microstructure *Trans. AIME* **175** 15–51
- [6] Saylor D M, Morawiec A and Rohrer G S 2003 The relative free energies of grain boundaries in magnesia as a function of five macroscopic parameters *Acta Mater.* **51** 3675–86
- [7] Li J, Dillon S J and Rohrer G S 2009 Relative grain boundary area and energy distributions in nickel *Acta Mater.* **57** 4304–11
- [8] Rohrer G S, Li J, Lee S, Rollett A D, Groeber M and Uchic M D 2010 Deriving grain boundary character distributions and relative grain boundary energies from three-dimensional EBSD data *Mater. Sci. Technol.* **26** 661–9
- [9] Schwartz A J, Kumar M and Adams B L 2010 *Electron Backscatter Diffraction in Materials Science* 2 edn (Dordrecht: Springer)
- [10] Suter R M, Hennessy D, Xiao C and Lienert U 2006 Forward modeling method for microstructure reconstruction using x-ray diffraction microscopy: single-crystal verification *Rev. Sci. Instrum.* **77** 123905
- [11] Saylor D M, El Dasher B S, Rollett A D and Rohrer G S 2004 Distribution of grain boundaries in aluminum as a function of five macroscopic parameters *Acta Mater.* **52** 3649–55
- [12] Saylor D M, El-Dasher B S, Adams B L and Rohrer G S 2004 Measuring the five-parameter grain-boundary distribution from observations of planar sections *Metall. Mater. Trans. A* **35** 1981–9
- [13] Dey T K and Goswami S 2004 Provable surface reconstruction from noisy samples *Proc. 20th Annual Symp. on Computational Geometry (Brooklyn, NY)* vol 20 pp 330–9
- [14] Hoppe H, DeRose T, Duchamp T, McDonald J and Stuetzle W 1992 Surface reconstruction from unorganized points *Comput. Graph. (ACM)* **26** 71–8
- [15] Mitra N J, Nguyen A N and Guibas L 2004 Estimating surface normals in noisy point cloud data *Int. J. Comput. Geom. Appl.* **14** 261–76

- [16] Ivasishin O M, Shevchenko S V and Semiatin S L 2009 Implementation of exact grain-boundary geometry into a 3-D Monte-Carlo (Potts) model for microstructure evolution *Acta Mater.* **57** 2834–44
- [17] Moore D and Warren J 1991 Approximation of dense scattered data using algebraic surfaces *Proc. 24th Annual Hawaii Int. Conf. on System Sciences (Kauai, HI)* vol i, pp 681–90
- [18] Lorensen W E and Cline H E 1987 Marching cubes: a high resolution 3D surface construction algorithm *SIGGRAPH Comput. Graph.* **21** 163–9
- [19] Groeber M and Jackson M 2014 DREAM.3D: a digital representation environment for the analysis of microstructure in 3D *Integr. Mater. Manuf. Innov.* **3** 5
- [20] Saylor D M, Morawiec A and Rohrer G S 2003 Distribution of grain boundaries in magnesia as a function of five macroscopic parameters *Acta Mater.* **51** 3663–74
- [21] Chandross M and Holm E A 2010 Measuring grain junction angles in discretized microstructures *Metall. Mater. Trans. A* **41** 3018–25
- [22] Hefferan C M *et al* 2009 Statistics of high purity nickel microstructure from high energy x-ray diffraction microscopy *Comput. Mater. Continua* **14** 207–17
- [23] Hefferan C 2012 Measurement of annealing phenomena in high purity metals with near-field high energy x-ray diffraction microscopy *PhD Thesis* Carnegie Mellon University
- [24] Suter R M, Li S F, Hefferan C and Lind J, Thermally induced coarsening in a nickel polycrystal observed in three dimensions, in preparation
- [25] Kalogerakis E, Simari P, Nowrouzezahrai D and Singh K 2007 Robust statistical estimation of curvature on discretized surfaces *Symp. on Geometry Processing (Barcelona, Spain)* pp 13–22
- [26] Garapić G, Faul U H and Brisson E 2013 High-resolution imaging of the melt distribution in partially molten upper mantle rocks: evidence for wetted two-grain boundaries *Geochem. Geophys. Geosyst.* **14** 556–66
- [27] Feng X, Xia K, Tong Y and Wei G-W 2012 Geometric modeling of subcellular structures, organelles, and multiprotein complexes *Int. J. Numer. Methods Biomed. Eng.* **28** 1198–223
- [28] Lebensohn R A, Brenner R, Castelnaud O and Rollett A D 2008 Orientation image-based micromechanical modelling of subgrain texture evolution in polycrystalline copper *Acta Mater.* **56** 3914–26
- [29] Lebensohn R A, Kanjarla A K and Eisenlohr P 2012 An elasto-viscoplastic formulation based on fast Fourier transforms for the prediction of micromechanical fields in polycrystalline materials *Int. J. Plast.* **32–33** 59–69
- [30] Lebensohn R A, Escobedo J P, Cerreta E K, Dennis-Koller D, Bronkhorst C A and Bingert J F 2013 Modeling void growth in polycrystalline materials *Acta Mater.* **61** 6918–32
- [31] MacSleynne J P, Simmons J P and De Graef M 2008 On the use of moment invariants for the automated analysis of 3D particle shapes *Modelling Simul. Mater. Sci. Eng.* **16** 045008
- [32] Lo C-H and Don H-S 1989 3-D moment forms: their construction and application to object identification and positioning *IEEE Trans. Pattern Anal. Mach. Intell.* **11** 1053–64
- [33] Monga O and Benayoun S 1995 Using partial derivatives of 3D images to extract typical surface features *Comput. Vis. Image Understand.* **61** 171–89
- [34] Lindeberg T 1993 Discrete derivative approximations with scale-space properties: a basis for low-level feature extraction *J. Math. Imag. Vis.* **3** 349–76
- [35] Lindeberg T 1998 Edge detection and ridge detection with automatic scale selection *Int. J. Comput. Vis.* **30** 117–56
- [36] Tasdizen T, Whitaker R, Burchard P and Osher S 2003 Geometric surface processing via normal maps *ACM Trans. Graph.* **22** 1012–33
- [37] Wang Y, Sheng Y, Lu G, Tian P and Zhang K 2008 Feature-constrained surface reconstruction approach for point cloud data acquired with 3D laser scanner *Proc. SPIE* **7000** 700021
- [38] Apostol T M 1969 *Calculus* vol 2, 2nd edn (Waltham, MA: Xerox Publishing) sections 8.16 and 12.1-3
- [39] Guest P G 1961 *Numerical Methods of Curve Fitting* (New York: Cambridge University Press) especially chapters 6 and 7
- Ryan T P 1997 *Modern Regression Methods* (New York: Wiley) especially chapters 1 and 8

- [40] Leclaire S, El-Hachem M, Trépanier J-Y and Reggio M 2014 High order spatial generalization of 2D and 3D isotropic discrete gradient operators with fast evaluation on GPUs *J. Sci. Comput.* **59** 545–73
- [41] Shima E, Kitamura K and Haga T 2013 Green–gauss/weighted-least-squares hybrid gradient reconstruction for arbitrary polyhedra unstructured grids *AIAA J.* **51** 2740–7
- [42] Lindeberg T 2011 Generalized Gaussian scale-space axiomatics comprising linear scale-space, affine scale-space and spatio-temporal scale-space *J. Math. Imag. Vis.* **40** 36–81
- [43] Saylor D, Fridy J, El-Dasher B, Jung K-Y and Rollett A 2004 Statistically representative three-dimensional microstructures based on orthogonal observation sections *Metall. Mater. Trans. A* **35** 1969–79
- [44] Brahme A, Alvi M H, Saylor D, Fridy J and Rollett A D 2006 3D reconstruction of microstructure in a commercial purity aluminum *Scr. Mater.* **55** 75–80
- [45] Lienert U *et al* 2011 High-energy diffraction microscopy at the advanced photon source *JOM* **63** 70–7
- [46] Li S F 2011 Imaging of orientation and geometry in microstructures: development and applications of high energy x-ray diffraction microscopy *PhD Thesis* Carnegie Mellon University
- [47] Alliez P, Rineau L, Tayeb S, Tournois J and Yvinec M 2011 3D mesh generation *CGAL User and Reference Manual* ed CGAL Editorial Board
- [48] Boltcheva D, Yvinec M and Boissonnat J D 2009 Mesh generation from 3D multi-material images *12th Int. Conf. on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2009) (Berlin, Germany, 20–24 September)* pp 283–90
- [49] Bieler T R *et al* 2009 The role of heterogeneous deformation on damage nucleation at grain boundaries in single phase metals *Int. J. Plast.* **25** 1655–83