Optical inspection of coated particle nuclear fuel*

Jeffery R. Price† and John D. Hunn
Oak Ridge National Laboratory, Oak Ridge, TN, USA

ABSTRACT

In this paper, we describe the inspection of coated particle nuclear fuel using optical microscopy. Each ideally spherical particle possesses four coating layers surrounding a fuel kernel. Kernels are designed with diameters of either 350 or 500 microns and the other four layers, from the kernel outward, are 100, 45, 35, and 45 microns, respectively. The inspection of the particles is undertaken in two phases. In the first phase, multiple particles are imaged via back-lighting in a single 3900 x 3090 image at a resolution of about 1.12 pixels/micron. The distance transform, watershed segmentation, edge detection, and the Kasa circle fitting algorithm are employed to compute total outer diameters only. In the second inspection phase, the particles are embedded in an epoxy and cleaved (via polishing) to reveal the cross-section structure of all layers simultaneously. These cleaved particles are imaged individually at a resolution of about 2.27 pixels/micron. We first find points on the kernel boundary and then employ the Kasa algorithm to estimate the overall particle center. We then find boundary points between the remaining layers along rays emanating from the particle center. Kernel and layer boundaries are detected using a novel segmentation approach. From these boundary points, we compute and store layer thickness data.

Keywords: image-based inspection, image segmentation, circle fitting, coated-particle fuel, TRISO fuel

1. INTRODUCTION

The U.S. Department of Energy (DOE) Advanced Gas Reactor Fuel Development and Qualification program was conceived to support near-term deployment of high-temperature, gas-cooled reactor technology and to establish a basis for the development of fuels suitable for very high-temperature, gas-cooled reactors. The first phase of this program concentrates on the re-establishment and improvement of the capability to produce high quality, silicon carbide-based, coated fuel particles (also known as ceramic-coated or TRISO fuel). A cross-section representation of these spherical particles is shown in Fig. 1. A new coated particle fuel development and characterization facility is being established at Oak Ridge National Laboratory (ORNL). In this facility, we are assembling the capability to characterize coated particles containing depleted, natural, and enriched uranium. This capability is required to support coating development work and predict the ultimate performance of the fuel particles. Technological advances, particularly in the areas of digital imaging and computing, offer the opportunity to improve upon the characterization methods that were developed two to four decades ago when coated particle fuel was first developed. With this in mind, ORNL is seeking to apply this newer technology to enhance the efficiency and accuracy of the coating characterization.

In this paper, we discuss fuel particle characterization efforts involving light microscopy and image processing. The particles to be inspected are imaged in two different phases. In the first phase, particles or kernels are collected, in a single layer, in an optically transparent dish and imaged using back-lighting. The resulting images, an example of which can be seen in Fig. 2, are analyzed to measure the outer surface characteristics only (of either kernels or coated particles). In the second phase of optical inspection, complete particles are embedded in an acrylic epoxy and ground down to just short of the spherical midpoint. These images (an example is shown later in Fig. 7) are used to measure width characteristics of all the material layers in the particle. The imaging system employed for these inspection tasks comprises a Leica DMRX analytical, upright microscope with fluotar objectives fitted with a Leica DC500 camera. This

* Prepared by Oak Ridge National Laboratory, managed by UT-Battelle, LLC, for the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.
† Correspondence: J. Price, pricejr@ornl.gov, 865-574-5743.

camera employs a 1-megapixel Bayer mosaic chip and piezoelectric stepping to acquire 12-megapixel images from 32 frames. The 8-bit grayscale images were computed from the Bayer array using the Leica-provided software.

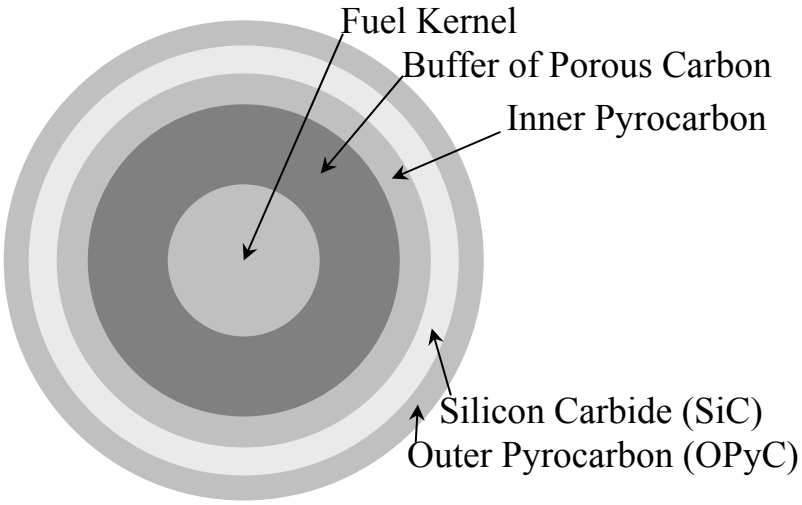

Figure 1. Cross-sectional view of the nominally spherical coated particle fuel. Each particle is composed of five layers which are, from inside to outside, respectively: the fuel kernel, a porous carbon buffer, inner pyrocarbon, silicon carbide, and outer pyrocarbon.

The remainder of this paper is organized as follows. In Section 2, we describe the approach taken to characterize the outer surfaces from the back-lit images. We then discuss the methods employed to characterize the cross-sectional images in Section 3. We conclude in Section 4 with some closing comments and some discussion of ongoing work.

2. OUTER SURFACE MEASUREMENTS

In this section, we describe the methods we have adopted to collect outer surface data on back-lit fuel particles such as those shown in Fig. 2. We seek to sample the radii of the particles in one-degree increments so that a perfectly imaged particle (i.e., no outlier points) that does not touch another particle (causing the portions of the edge to disappear) would have 360 radii samples. The 3900×3090-pixel images used for this phase of the inspection have a resolution of about 1.12 pixels/micron. For the back-lit examples in this paper, the particles we have imaged are approximately 500 microns in diameter. In Section 2.1, we describe how each individual particle is located in images such as that of Fig. 2. Then, in Section 2.2, we discuss the approach we employ to estimate particle radii and discard outliers (due to noise or dirt).

2.1. Individual particle detection

As evident in Fig. 2, the particles in the back-lit images appear as solid black disks on an approximately white background. The large black region along the top of Fig. 2 is an opaque portion of the dish containing the particles. We first seek to locate the approximate center of each particle (disk). For this purpose, the full image resolution is not necessary. Hence, to improve computational performance in this phase, we decrease image resolution by a factor of four via subsampling. Working from this reduced resolution image, we next perform morphological dilation[1] using a disk structuring element of radius equal to one-tenth of the expected particle diameter. Dilation expands the bright areas of the image and in the result the disks are well separated from one another and from the opaque background.

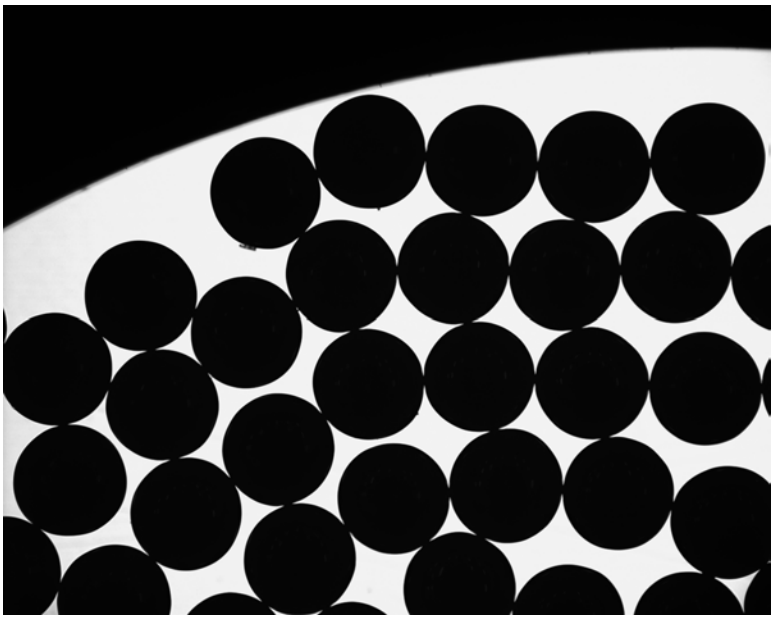

Figure 2. Multiple particles imaged with back-lighting.

We next take the dilated image, invert the gray values, and then threshold the image using Otsu's method[2] which minimizes the interclass variance, as determined by the image histogram, between thresholded white and black pixels. The result is an image where all of the particle and opaque background regions are white, as illustrated in Fig. 3. We compute the centroid, area, and eccentricity of each distinct white region in Fig. 3 to identify which regions are to be processed as particles and which are to be discarded. Regions too close to the image boundaries, with too little or too much area (relative to the expected particle size), or that are not nearly circular are all discarded.

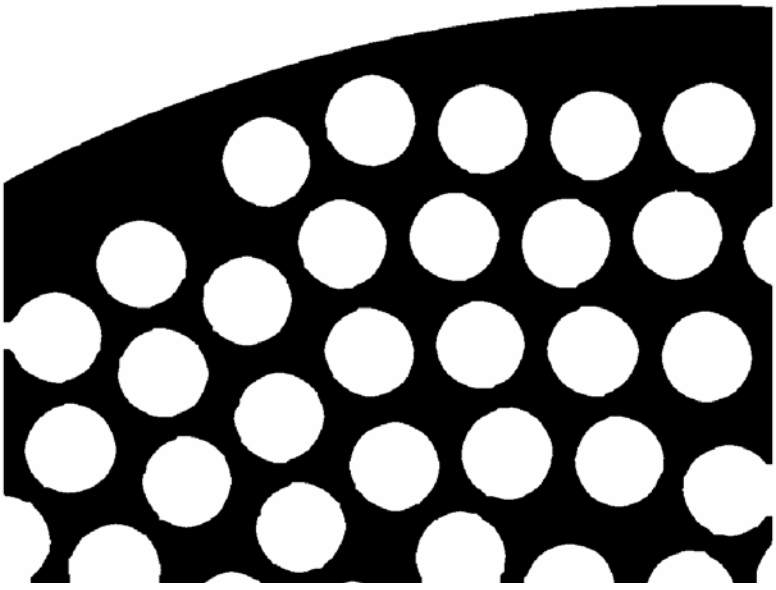

Figure 3. Thresholded image after dilation and gray-scale inversion of image from Fig. 2.

Now that we have identified regions for processing, we need boundaries that separate the regions from one another so that each particle can be analyzed independently. Towards this goal, we apply the distance transform[3] to the image of

Fig. 3 and then perform watershed segmentation[3] on the distance transform result. The watershed segmentation provides us with the boundaries we seek, as illustrated in the example of Fig. 4. We now turn our attention to processing each particle region.

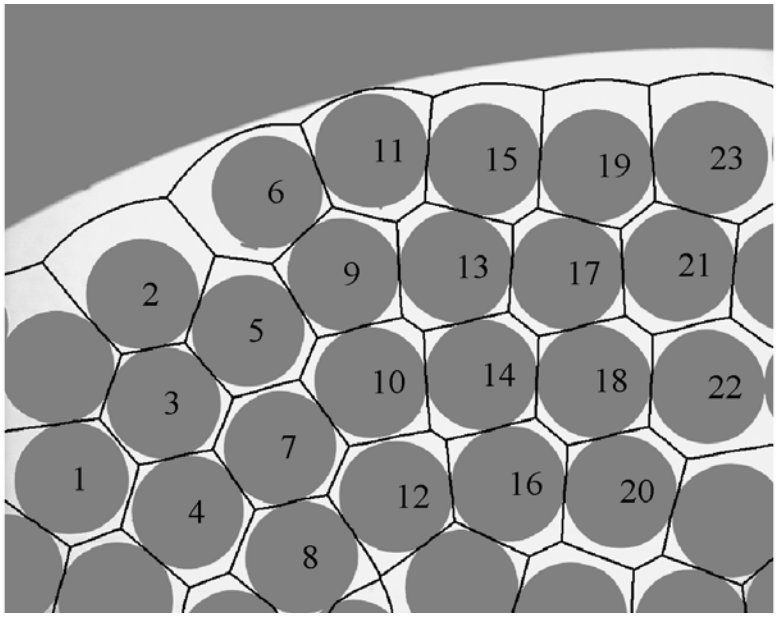

Figure 4. Results of watershed segmentation and region selection. Particles too near the image edge are not processed. (We have manually adjusted the gray-level values for black-and-white visualization purposes.)

2.2. Radii measurements and outlier detection

We begin by mapping the reduced-resolution watershed boundaries back to the full resolution image coordinates. We next extract a rectangular sub-image, corresponding to each region's bounding box, from the full resolution image. We perform Sobel edge-detection on this sub-image (discarding any edges outside the watershed region boundary) and skeletonize the result to acquire edges of only one pixel in width. As an initial step in outlier rejection, we discard any edge regions of less than 20 contiguous pixels. We then fit the remaining edge pixels to a circle, using the Kasa method,[4,5] to provide an initial estimate for the particle center.

Since we desire only one radius measurement per degree, we next divide the edge image into 360 one-degree wedges emanating from the estimated particle center. We find the number of edge pixels in each such wedge; if this number is greater than zero and less than or equal to a pre-determined maximum (which we will describe shortly), we keep the minimum radius of the pixels in that wedge as the radius for the given angular sample. The maximum number of pixels allowed in a given wedge is set to the expected arc length in pixels, where the expected arc length is calculated from the particle design radius. The presence of more pixels in the wedge than expected indicates noise (due to dirt) or perhaps another particle boundary that was not properly segmented. We now have up to 360 radii measurements (i.e., boundary points) relative to our initial estimate of particle center. We now repeat the Kasa circle fit using these boundary points to update our estimate of particle center.

Using the updated estimate of the particle center, we find the radii of the boundary points. We then apply a commonly used statistical rule-of-thumb[6] and label as a potential outlier any point whose radius is greater than 1.5 times the interquartile range above the third quartile (the so-called 1.5 × IQR rule). As this approach can sometimes label true boundary points as outliers – if, for example, most of a particle is very nearly circular – we perform an iterative process to verify the outliers.

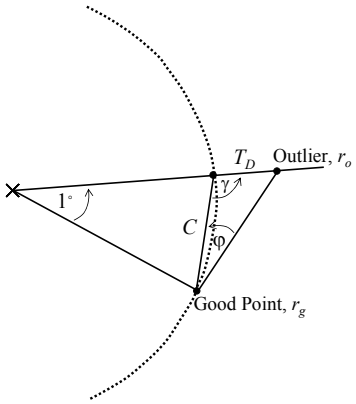

Figure 5. Quantity, T_D, used to determine if a potential outlier should be relabeled as a good point.

For each potential outlier, we check to see if it has any good neighbors (i.e., points not labeled as outliers). If not, the point in question remains an outlier at this step. Otherwise, we consider a quantity, T_D, as illustrated in Fig. 5. This quantity is based upon the maximum angular deviation, φ, that we will allow from a circular curve. Examining Fig. 5, we note that the angle γ is essentially 90°. If we accept this approximation, then $T_D = C\tan\varphi$, where we define C to be the chord length determined by the radius of the good point, r_g. Hence, $T_D = 2r_g \sin(0.5^\circ)\tan(\varphi)$, and the potential outlier is relabeled as a good point if $|r_o - r_g| \le T_D$. This process is repeated iteratively until no potential outliers are relabeled as good points. We then perform the Kasa circle fit one last time on the good points to compute the final estimate of the particle center. An example result is illustrated in Fig. 6. The radii of the good points are then measured relative to this center.

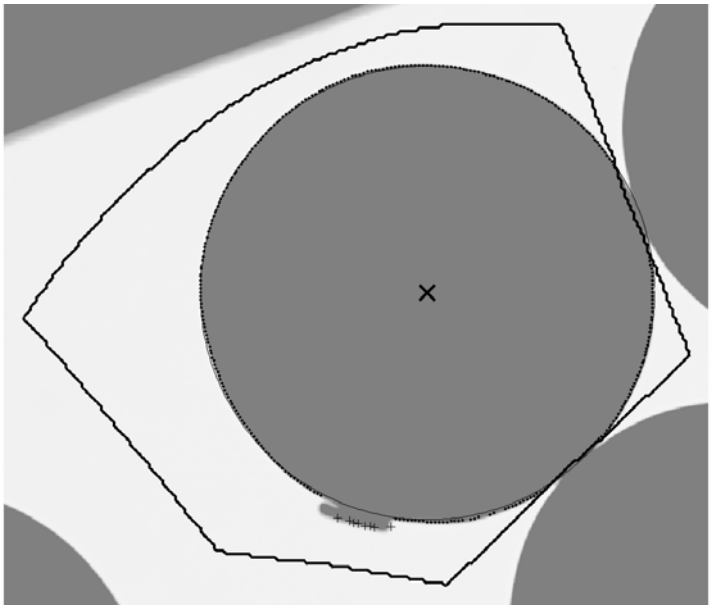

Figure 6. Example results of outer surface detection on region 6 from Fig. 4. The thick polygon represents the watershed boundary, the dots represent the retained boundary points, the thick "X" represents the particle center, the "+' marks represent points discarded as outliers, and the thin solid circle represents the Kasa-fit circle.

3. CROSS-SECTIONAL MEASUREMENTS

In this section, we discuss methods to measure cross-sectional layer thicknesses of fuel particles from images such as that shown in Fig. 7. These images are acquired by embedding the complete particles in an acrylic epoxy and then grinding them down to near the spherical center. We note that the particles cannot be ground to the true center for two reasons. First, the kernels tend to loosen inside the buffer during grinding. If the particles were ground to or just past the center, the loosened kernels might actually fall out. The kernel loosening is actually the cause of the partial black ring around the kernel of Fig. 7, which is a void caused by the kernel breaking away some of the buffer material during grinding. Secondly, stopping short of the particle center leaves some excess particle visible. This excess can be seen as the dark band with an out-of-focus outer edge outside the fifth layer in Fig. 7. The width of this excess can be used to computationally correct for the observed cross-section not passing through the true particle center. If, however, we attempt to grind to the exact center of the particle, there is a possibility that we will bypass the center and there will be no excess particle to indicate that fact or correct for the induced error.

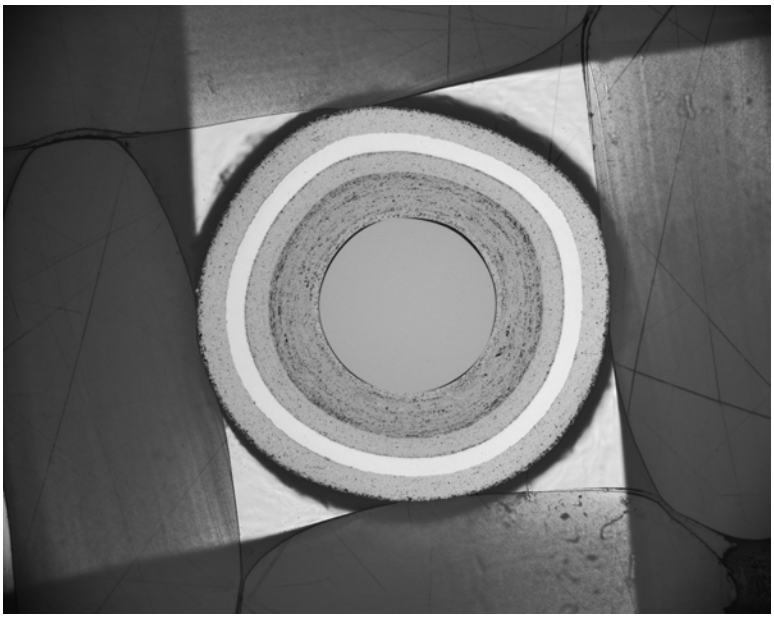

Figure 7. One example of the images used for cross-sectional layer thickness measurements. The dark region with an out-of-focus outer edge that is just outside the fifth layer is the excess particle resulting from not being able to grind the particle down to the true center.

For the cross-sectional measurements, we seek to sample the five material interface boundaries and the out-of-focus particle edge at 360 points in one-degree increments. The 3900 × 3090 images used in this phase of the inspection have a resolution of about 2.27 pixels per micron, while the particles we show in our examples are designed to have kernels of 500 microns in diameter with layer thickness of 100 microns (buffer), 45 microns (IPyC), 35 microns (SiC), and 45 microns (OPyC).

3.1. Estimating the kernel center and unwrapping the image

We first seek an initial estimate of the particle center. Because of the image acquisition process, we know that the center point of the image lies within the kernel. Furthermore, the kernels are quite homogenous. Taking advantage of these facts, we find the median gray value in a 21 × 21 window about the image center and then set all image points to one if they are within 20 gray-levels of this median, and to zero otherwise. The resulting binary image has the kernel region, along with some additional regions that happen to have similar gray values, all set to one. We then perform a morphological erosion to ensure that the kernel region is detached from any other regions that may also have been set to one. We follow this with a morphological closing to fill any holes in the kernel region that may have arisen from the thresholding and erosion. The result of this process can be seen in the example of Fig. 8, where the contiguous circular region in the center represents the kernel. We take the region that coincides with the image center to be the kernel and

fit a circle to its perimeter points using the Kasa method. This estimated center of the kernel region serves as our initial estimate of the particle center.

Figure 8. Finding the kernel center, from the image of Fig. 7, via thresholding, erosion, and dilation. Note that only a center sub-image of twice the kernel design diameter on each side is processed. The perimeter points of the center white region (the kernel) are used to provide an initial estimate of the particle center.

Using the estimated center point, we then resample the image on a polar grid at one-degree angular resolution and one-pixel radial resolution. This results in an "unwrapped" version of the image as illustrated in the example of Fig. 9, where the vertical axis represents angle in degrees and the horizontal axis represents radius in microns. When performing layer boundary detection, as described in Section 3.2 below, we work only with this unwrapped image.

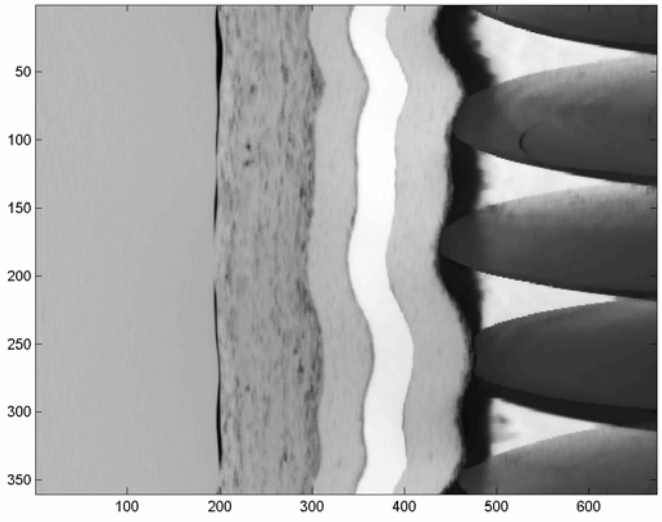

Figure 9. "Unwrapped" image computed by sampling image of Fig. 7 on a polar grid centered on the estimated particle center. Vertical scale represents angle in degrees and horizontal scale represents microns.

3.2. Layer boundary detection: marching points

Given the unwrapped image of Fig. 9, we now seek to locate the layer boundaries (and the out-of-focus particle boundary) so that we can compute and store layer thickness information. To locate the boundaries, we adopt an approach that we refer to as "marching points," which can be summarized as follows. For each successive layer, beginning with the kernel, a front of 360 points – one for each angular sampling interval – is initialized within the layer at some radius (recall that the horizontal axis of Fig. 9 represents radius). Each point then marches left-to-right, one pixel at a time, across the unwrapped image. The marching continues, according to certain smoothness constraints, until the layer boundary is identified as determined by objective-based stopping criteria that are specific to the image characteristics of the boundary in question. After first finding the kernel boundary in this manner, we map the kernel boundary points back to the original image coordinate system and then compute a new estimate of the particle center by again applying the Kasa circle-fit. A new unwrapped image is computed from the updated center point and all material boundaries are found from this unwrapped image. In the following paragraphs, we will discuss in more detail the initialization of the marching points, the smoothness constraints, and the objective-based stopping criteria, respectively.

The initial radius for each marching point in a layer is set according to the following equation:

$$r_k^i(\theta) = r_{k-1}^f(\theta) + 0.7L_k \,, \tag{1}$$

where $r_k^i(\theta)$ is the initial radius in layer k at angle θ, $r_{k-1}^f(\theta)$ is the final radius in layer $k-1$ at angle θ, L_k represents the design thickness for layer k, and $k = 1,...,5$ (if $k = 1$, the layer is the kernel and $r_0^f(\theta) = 0$ for all θ). Due to the manufacturing process, we do not expect to see layer thicknesses less than 0.8 of the design. Hence, initializing each point to be only 0.7 times the design layer thickness into the next layer ensures, with an extra margin of safety, that we will not accidentally initialize past the boundary we are seeking. A point is allowed to march left-to-right only, i.e., radius is only allowed to remain constant or increase at each iteration. During the marching, smoothness rules are applied that prevent a point from getting too far ahead or being left too far behind relative to its neighbors. At each iteration, a point steps one pixel to the right if either its objective has not yet been met or if its neighbors pull it along because of smoothness constraints. A point will remain stationary at an iteration if its objective has been met or if the step will take it too far ahead of its neighbors and violate the smoothness constraints.

We now turn our attention to the smoothness constraints. Letting $r_k^n(\theta)$ represent the radial position of the point at angle θ and iteration n, $r_k^n(\theta-1)$ and $r_k^n(\theta+1)$ represent the direct neighbors of $r_k^n(\theta)$. Note that θ ranges over [0,359] and is computed modulo 360. We define S_a to be the five neighbors defined by $\theta-1$ to $\theta-5$ and S_b to be the five neighbors defined by $\theta+1$ to $\theta+5$. Referring to Fig. 10, we also define a distance threshold parameter, T_D. At the radii we will be examining, note that the angle γ is essentially a right angle. Letting φ, as shown in Fig. 10, represent the maximum angular deviation from circular that we will allow, then T_D is given simply by $T_D = C\tan\varphi$, where we define C to be the chord length using the mean of the $r_k^n(\theta-1)$ and $r_k^n(\theta+1)$ as the effective radius. Given the above definitions, $r_k^n(\theta)$ will be allowed to step forward if *all* of the following conditions are satisfied:

- $r_k^n(\theta) - r_k^n(\theta-1) < T_D$, and
- $r_k^n(\theta) - r_k^n(\theta+1) < T_D$, and
- $r_k^n(\theta) - \min_{\alpha\in S_a}\left(r_k^n(\alpha)\right) < 3T_D$, and
- $r_k^n(\theta) - \min_{\beta\in S_b}\left(r_k^n(\beta)\right) < 3T_D$.

These conditions prevent excessive protrusions in case a boundary is missed or not clearly present. An example of the effectiveness of this method is illustrated in Fig. 11, which is a magnified version of the kernel boundary from Fig. 9.

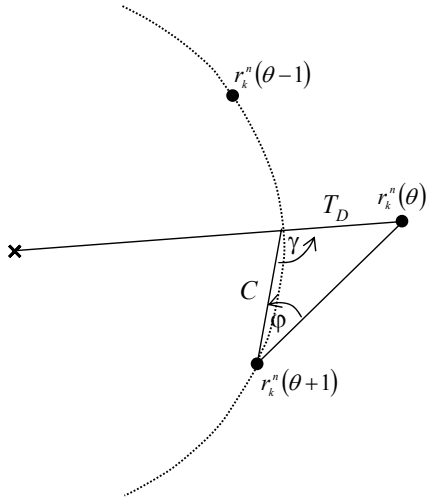

Figure 10. Illustration of how the distance threshold parameter, T_D, is defined.

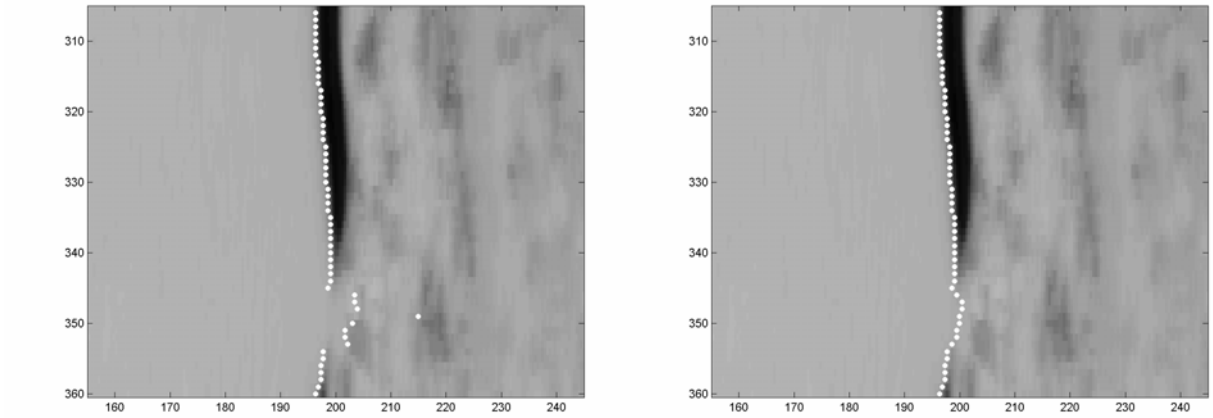

Figure 11. Smoothness constraints prevent protrusions. Marching points results (a) with and (b) without smoothness constraints. Vertical scale represents angle in degrees and horizontal scale represents microns.

Continuing our discussion of the smoothness constraints, $r_k^n(\theta)$ will be forced to step forward (even if its objective has been met), if *all* of the following conditions are satisfied:

- At least four of the five neighbors in S_a satisfy $r_k^n(\alpha) - r_k^n(\theta) > 0$, and
- At least four of the five neighbors in S_b satisfy $r_k^n(\beta) - r_k^n(\theta) > 0$, and
- $\underset{\alpha \in S_a}{\text{median}} \{ r_k^n(\alpha) - r_k^n(\theta) \} > T_D$, and
- $\underset{\beta \in S_b}{\text{median}} \{ r_k^n(\beta) - r_k^n(\theta) \} > T_D$.

These conditions effectively prevent a point from becoming stuck because it has mistakenly identified its objective as being met. An example of the effectiveness of this approach is illustrated in Fig. 12.

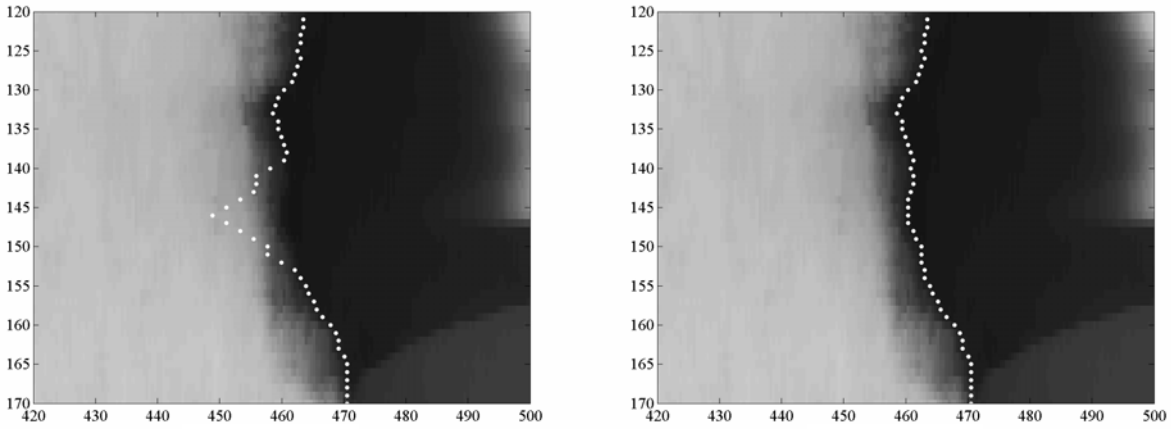

Figure 12. Smoothness constraints prevent points from being left behind. Marching point results (a) with and (b) without smoothness constraints. Vertical scale represents angle in degrees and horizontal scale represents microns.

We now briefly discuss the objective-based (as opposed to smoothness-based) stopping criteria of the marching points. As we mentioned earlier, each point marches through its layer seeking a boundary. A point determines that a boundary has been encountered when an objective, tailored to the given layer boundary, has been obtained. These objectives, though varying in minor details for each layer, are based primarily on seeking a significant change in gray value and/or an appropriately signed gradient (along the horizontal direction in the unwrapped image) of a significant magnitude. Recalling Equation (1), we characterize both the gray values and gradient values of a Layer *k* in a given image by computing the statistics of the unwrapped image pixels between the previous boundary, $r_{k-1}^{f}(\theta)$, and the initialization of the current boundary, $r_{k}^{i}(\theta)$.

To briefly demonstrate the approach, we provide details for the objective-based stopping criteria we have implemented for Layer 3 (IPyC). Note from Fig. 9, that Layer 3 is generally a mid-level gray transitioning to the nearly white Layer 4. Hence, when developing the objective for detecting the Layer 3 boundary, we are looking for a significant increase in gray level as well as a significant, positive gradient. For each marching point, $r_3^n(\theta)$, we compute statistics over the set of pixel locations in the unwrapped image defined by $\left[r_2^f(\phi), r_3^i(\phi)\right]$ where $\phi \in [\theta-15, \theta+15]$. We compute the first quartile and third quartile of the gray-level values – given by $Q_{v,1}(\theta)$ and $Q_{v,3}(\theta)$, respectively – as well as the first and third quartiles of the horizontal gradient – given by $Q_{g,1}(\theta)$ and $Q_{g,3}(\theta)$, respectively. We define the gray-value threshold $T_v(\theta)$ as

$$T_v(\theta) = Q_{v,3}(\theta) + 6\left(Q_{v,3}(\theta) - Q_{v,1}(\theta)\right), \tag{2}$$

and the gradient threshold $T_g(\theta)$ as

$$T_g(\theta) = Q_{g,3}(\theta) + 1.5\left(Q_{g,3}(\theta) - Q_{g,1}(\theta)\right). \tag{3}$$

We let $V\left(r_3^n(\theta)\right)$ and $G\left(r_3^n(\theta)\right)$ represent the gray-value and signed, horizontal gradient, respectively, of the unwrapped image at the current location of the marching point. The marching point will stop if all of the following conditions are true:

- $V\left(r_3^n(\theta)+1\right) > T_v(\theta)$, and
- $G\left(r_3^n(\theta)+1\right) > T_g(\theta)$, and

- $G\left(r_3^n(\theta)+1\right) < G\left(r_3^n(\theta)\right)$.

The first two conditions ensure that both the gray-value and gradient are significantly higher than what was observed within Layer 3, while the third condition ensures that we stop at the local peak of gradient. The points march iteratively until no points move at a given iteration. Objective-based stopping criteria for the other boundaries are quite similar. In Fig. 13, we display the final result of the marching points approach applied to the image of Fig. 9 (including the detection of the out-of-focus particle edge). We note that, for the out-of-focus particle boundary, points which do not meet their objective are discarded since the boundary in question is not present across the entire angular range.

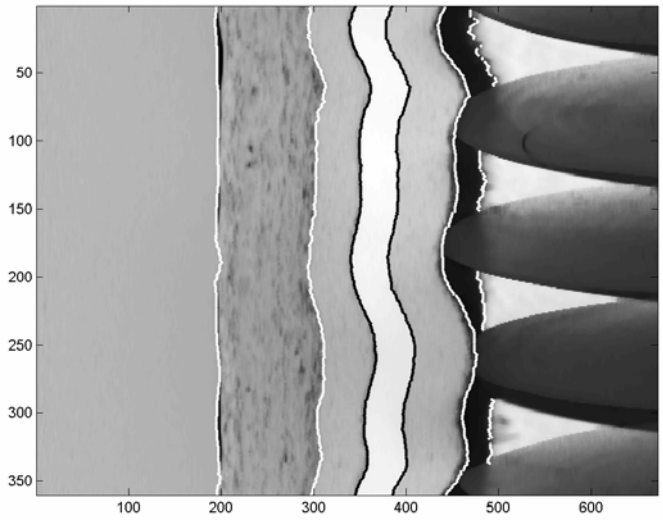

Figure 13. Result of the complete marching points approach at detecting the five layer boundaries and the out-of-focus particle edge. Vertical scale represents angle in degrees and horizontal scale represents microns.

3.3. Correcting for vertically displaced cross-section

As mentioned earlier, the particles for cross-section analysis are, by design, never ground completely down to the true center. This implies the layer radii measured from the unwrapped image are not true measurements from a great circle cross-section of the (nominally) spherical particle. Referring to Fig. 14, the vertical displacement ε from the true center can be estimated by computing the width of the excess particle, Δ, that is visible outside the outermost layer. Letting R_m represent the measured radius of the particle using the boundary of Layer 5, it is easy to show that

$$\varepsilon = \sqrt{2\Delta R_m + \Delta^2}\ , \tag{4}$$

where we set Δ to the median of the difference between the out-of-focus particle boundary radii and the Layer 5 boundary radii. Given ε, the corrections for each layer for are given by

$$\Delta_k = \left(\sqrt{\left(\bar{r}_k\right)^2 - \varepsilon^2}\right) - \bar{r}_k\ , \tag{5}$$

for $k = 1,...,5$, where $\bar{r}_k$ represents the median radius of the outer edge of Layer k. Applying this correction, the final radius for each angular sample of would be set to $r_k^f(\theta) + \Delta_k$.

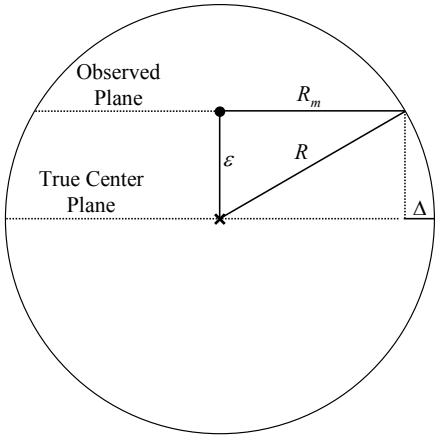

Figure 14. Measured radius, R_m, differs by Δ from the true radius, R, due to observing a cross-section that is displaced by from the true particle center by ε.

4. CONCLUSION AND FUTURE WORK

The two approaches described in Section 3 and 4 above have been implemented in MATLAB graphical user interfaces and are presently being used to gather critical measurement information necessary to refine and improve the particle development process. There are still, however, several planned improvements for the image analysis. First, we have not yet corrected for a slight deviation in magnification across the field of view caused by lens distortion. This will be a simple matter of intrinsic camera calibration. Secondly, as we noted earlier, the kernels can often loosen from the buffer during the grinding process. Furthermore, they can also rotate during the grinding. This can result in a kernel like that shown in Fig. 15. This effect can be minimized by a change in the grinding process, but we are also developing a method to recognize such "straight-edged" kernels and discard the boundary points along those edges.

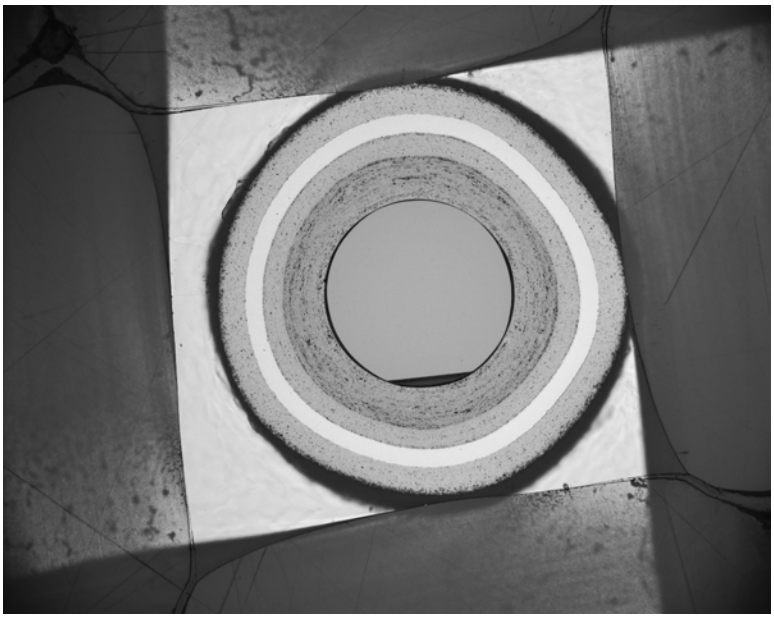

Figure 15. "Straight-edge" kernels can result when the kernel rotates during the grinding process. We are currently developing an approach to detect these straight edges and discard those measurements.

REFERENCES

1. P. Soille, *Morphological Image Analysis: Principles and Applications*, Springer-Verlag, 1999.
2. N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62-66, 1979.
3. J. Russ, *The Image Processing Handbook*, CRC Press, 1998.
4. I. Kasa, "A circle fitting procedure and its error analysis," *IEEE Transactions on Instrumentation and Measurement*, vol. 25, no. 1, pp. 8-14, March 1976.
5. C. Rusu, M. Tico, P. Kuosmanen, E. Delp, "Classical geometrical approach to circle fitting – review and new developments," *Journal of Electronic Imaging*, vol. 12, no.1, pp. 179-193, January 2003.
6. D. Moore, G. McCabe, *Introduction to the Practice of Statistics*, 3rd ed., Freeman, 1999.