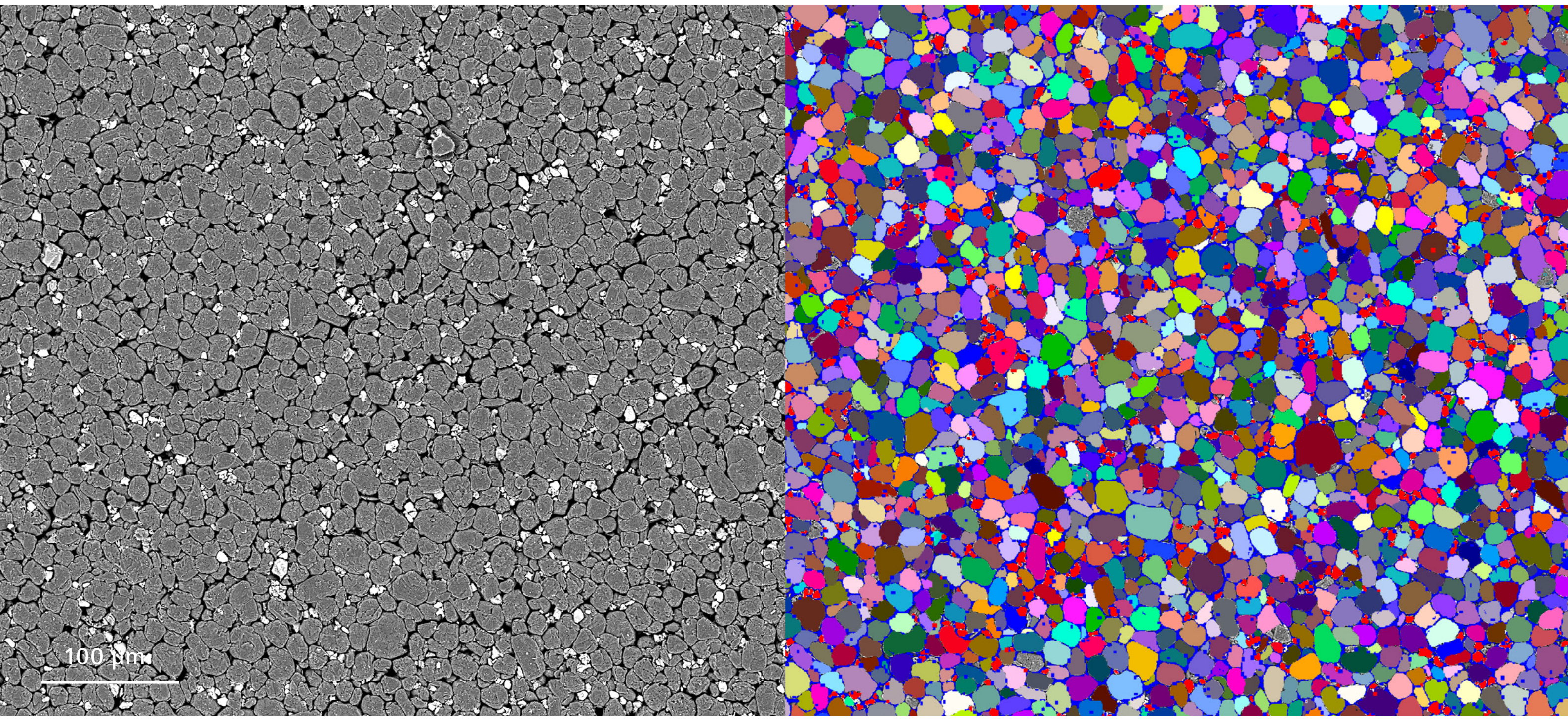
Application Note

Comparing Segmentation Methods for Particle Analysis of Lithium Ion Battery Materials using ZEN core



Seeing beyond

Author: Tim Schubert
Aalen University, Germany

Date: January 2025

Accurate particle size measurement is essential for R&D, enabling materials researchers and engineers in manufacturing and quality control to effectively analyze, predict, and control material behavior. The size and distribution of particles such as grains in construction and functional materials or active material particles in batteries strongly influence the properties of materials and components.

In the analysis of loose particles such as powders in additive manufacturing, powder metallurgy, or even in the pharmaceutical industry, laser diffraction is a widely used method for size and size-distribution measurements. This method provides insights into particle size classes and the size distributions, but usually does not provide geometric features of individual particles. Laser diffraction is also not suitable for grain size analysis in bulk polycrystalline materials. The quantitative analysis of images from light optical microscopes (LOM) or scanning electron microscopes (SEM) is a commonly used method to evaluate the geometric features of particles and to analyze grain size, shape, and distribution in polycrystalline materials. The challenge, however, is the segmentation process and especially the separation of individual particles or touching grains.

The software ZEN core from ZEISS, where ZEN stands for ZEISS Efficient Navigation, offers various types of segmentation methods for all kinds of images. The three most common types will be compared in this paper using the example of a lithium-ion battery anode. The three types are a global thresholding method, machine-learning-based segmentation utilizing the module ZEN Intellesis, and a cloud-trained deep-neural-network instance segmentation method using the platform ZEISS arivis Cloud and the model import into ZEN core.

Introduction

Analyzing the phase fractions and the particle size distribution in lithium-ion battery electrodes is crucial for both R&D as well as manufacturing and quality control, as it directly affects battery performance, stability, and lifetime.

Phase composition determines how effectively lithium ions can be stored and transported within the electrode, affecting capacity and charge/discharge rates. Particle size distribution affects the surface area available for reactions as well as electrode porosity, which influences the rate of ion diffusion and overall battery efficiency. Consistent and optimal particle size also minimizes degradation, improving battery life and safety. Therefore, understanding these parameters helps to optimize electrode design for enhanced battery performance and ensures product consistency during manufacturing. This applies to both the anode and cathode sides of lithium-ion batteries.

The anode material investigated in this paper is composed of graphite particles and finely dispersed silicon particles. Silicon particles are added to anodes to increase the energy density of batteries because silicon has a much higher theoretical lithium storage capacity than graphite. However, silicon also expands and contracts much more than graphite during charging and discharging of the battery, causing mechanical stress and degradation. To address this, silicon is blended with graphite to balance high capacity and structural integrity and stability. The combination enhances the overall capacity of the battery while maintaining the durability required for most commercial applications.

Figure 1 shows a top-view SEM image of the anode surface in backscattered electron contrast visualizing the difference in chemical composition as well as the morphology of the particles. Silicon looks bright, the dark areas represent porosity and space between the particles, and graphite looks gray with small white speckles on the surface. These white speckles are a feature of the graphite particles and do not belong to the silicon phase in the sample.

The analysis of this particular sample aimed to determine the graphite and silicon fractions as well as the size distribution of both phases.

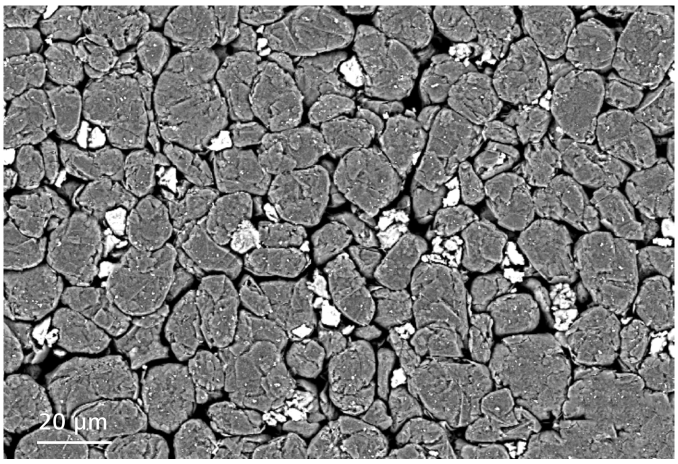


Figure 1: *SEM overview of the anode surface in backscattered electron contrast (BSE). White particles represent silicon, gray particles correspond to graphite, and black areas indicate porosity or spaces between particles. The image has a pixel size of 279 nm/pixel and a width of 203 µm.*

Segmentation Methods

For the measurement of geometric features, each particle has to be segmented as a separate object, whereas for the analysis of the phase fraction each image pixel belonging to a specific phase can be segmented together. For this purpose, a first differentiation has to be made between semantic and instance segmentation.

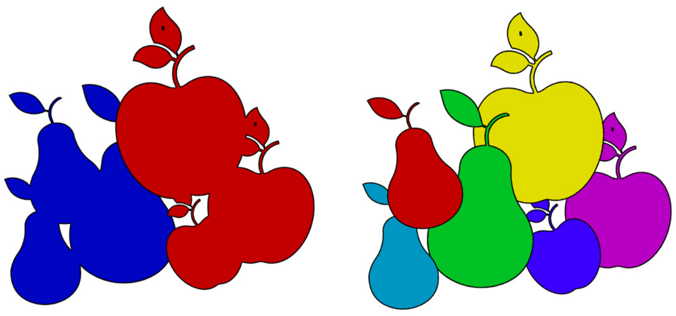


Figure 2: *Left: Semantic segmentation, assigning each image pixel to a phase, differentiating between apples and pears. Right: Instance segmentation, assigning objects to phases, distinguishing individual apples from individual pears.*

Figure 2 shows a schematic drawing of semantic and instance segmentation. While semantic segmentation classifies each pixel of the image into a specific class (or phase such as apples and pears) without differentiating between objects of the same type, instance segmentation is a more advanced type of segmentation that not only classifies each pixel but also differentiates between instances of the same class (or phase). In semantic segmentation each pixel belonging to a class is labeled the same, regardless of how many individual objects of that class are present in the image. In instance segmentation, however, individual objects of the same class are labeled separately. For the schematic drawing in Figure 2, this means semantic segmentation sees only apples and pears whereas instance segmentation sees three apples and three pears.

The three segmentation methods presented in this paper will show two simple semantic methods that require a secondary step to separate objects and one more advanced instance segmentation method that does not require a post-processing step. The three individual methods are described below.

To better describe the three segmentation methods, an artificial representation of the anode microstructure is used. This artificial representation is shown in Figure 3.

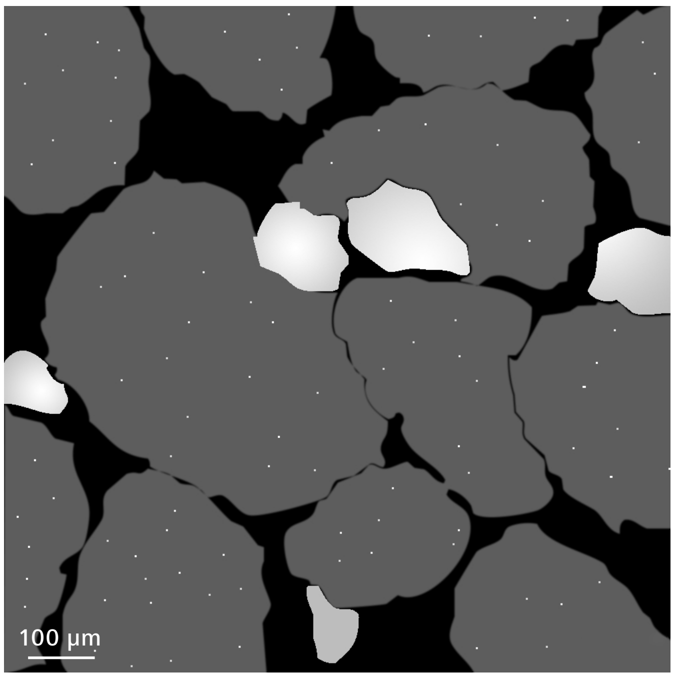


Figure 3: *Artificial representation of the anode surface: white particles represent silicon, dark areas represent pores, and gray particles with white speckles represent graphite.*

Threshold Segmentation

Threshold segmentation is a simple but effective technique for labeling pixels or objects in an image based on the pixel intensity values. The process involves selecting different threshold values for each class, and then classifying each pixel in the image into a class based on whether the intensity is above or below the set threshold.

Threshold segmentation works particularly well in images where different objects have different levels of brightness and contrast. These image types are mainly black-and-white or grayscale images. While this method is computationally efficient, it struggles with more complex images that may have intensity gradients or overlapping intensity values in individual objects.

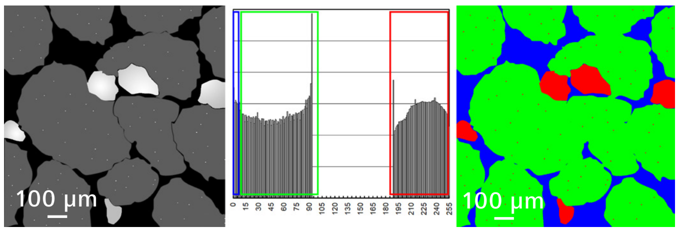


Figure 4: *Principle of threshold segmentation using an artificial anode image as an example. Left panel shows an artificial grayscale image. The center panel shows the image histogram with threshold settings. The right panel shows the threshold segmentation result.*

Figure 4 represents the principle of threshold segmentation. The input image (Figure 4 left) is the artificial image of the anode. It is an 8-bit grayscale image with a total of 256 gray values (0 = black, 255 = white). Figure 4 center shows the histogram (gray value distribution) of the input image as well as three threshold regions representing the three phases in the image. The dark background phase is covered by a threshold from 0 to 7, the gray graphite particles are covered by a threshold from 10 to 100, and the light silicon phase is covered by a threshold from 180 to 255. With this setting, each pixel in the image is assigned to one of the three phases and is labeled with the corresponding color. Note that the bright speckles on the graphite particles are assigned to the silicon phase because their intensity level is in the silicon threshold (Figure 4 right).

ZEN Intellesis Segmentation

ZEN Intellesis is a machine-learning-based segmentation tool in the ZEN ecosystem. It leverages machine learning algorithms to segment images based on models trained on user labeled data. The user can train the software to recognize and segment-specific features in microscopic images (LOM, SEM, X-ray imaging systems). Once trained, ZEN Intellesis enables automated segmentation tasks for high-throughput and reproducible image analysis. Image segmentation with ZEN Intellesis is a semantic pixel-based segmentation, where each pixel in the image has to be assigned to a class.

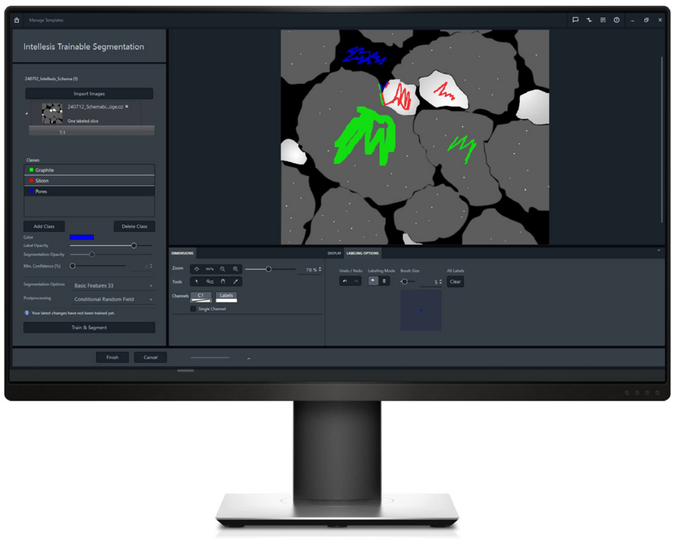


Figure 5: *ZEN Intellesis training – training image loaded, labeling of the individual phases.*

Figure 5 shows the training process in ZEN Intellesis. First, a set of training images is loaded into the training environment. The more training images with different imaging conditions, the more robust the model will be. Once the images are loaded, the user can create as many individual classes as are present in the image and start labeling the pixels belonging to each class. The labeling process basically involves painting the pixels belonging to a phase with a brush tool. The size of the brush tool can be adjusted according to the user's needs. In a subsequent step, the user can select from a range of feature sets, starting with 25 basic features up to 256 deep features. Note that a higher number of features will use more computational time and power. The feature set should be chosen according to the complexity of the image in a trial-and-error process. The final step is to click the "Train & Segment" button. Training then begins based on the labeled pixels.

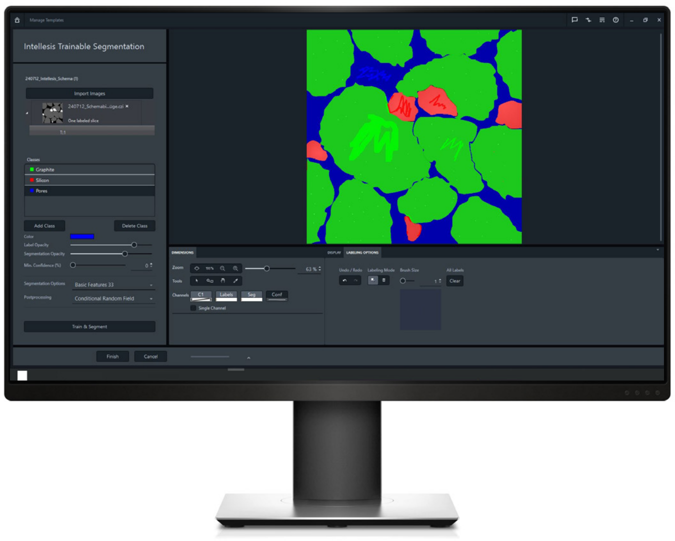


Figure 6: *ZEN Intellesis training result: Each pixel in the image is segmented according to the feature set learned in the training process.*

Figure 6 shows the training result for the ZEN Intellesis model. Each pixel in the image is assigned to a class that was found to best match the trained model based on the selected feature set. Note that the bright speckles on the graphite particles are assigned to the graphite and not to the silicon phase. If the user is satisfied with the training results and each pixel is assigned to the correct class, the model can be saved for further use in the ZEN core image-analysis workflow. If some pixels are still assigned to the wrong class, the user can continue the labeling process and perform more training steps. With each training step and more labeling (and even more training images), the model learns more and becomes more robust. It should be noted, however, that if the labeling results are not accurate or the training images become too complex, the results may become worse again. ZEN Intellesis training is a process of trial and repeat until the results meet the user's expectations. In the case of the artificial image, a short training with a basic set of features already shows very accurate segmentation results.

arivis Cloud Instance Segmentation

ZEISS arivis Cloud is a powerful platform for training custom deep-learning models for image segmentation. It provides intuitive tools for annotating images and supports both semantic and instance segmentation. This study leverages its instance segmentation capabilities to accurately segment particles as individual objects, using deep learning to tackle complex segmentation tasks. Models trained with arivis Cloud can be downloaded into the ZEN ecosystem to utilize powerful image-analysis modules (e.g., in ZEN core) to perform image analyses locally. For optimal performance during local analysis, a robust workstation and Docker software are recommended.

After labeling, the training process begins and runs unsupervised on the platform. Once training is complete, the platform notifies the user via email, and the evaluation of the training results can proceed. If the predictions from the initial training are unsatisfactory, additional labels can be added, and the training process can be continued. When the results meet expectations, the model can be downloaded and imported into the ZEN ecosystem for further analysis. Figure 8 illustrates the training results, showing predictions of the identified objects. Note that each object is segmented individually, including the bright speckles on the graphite particles, which are incorporated into the predicted graphite objects.

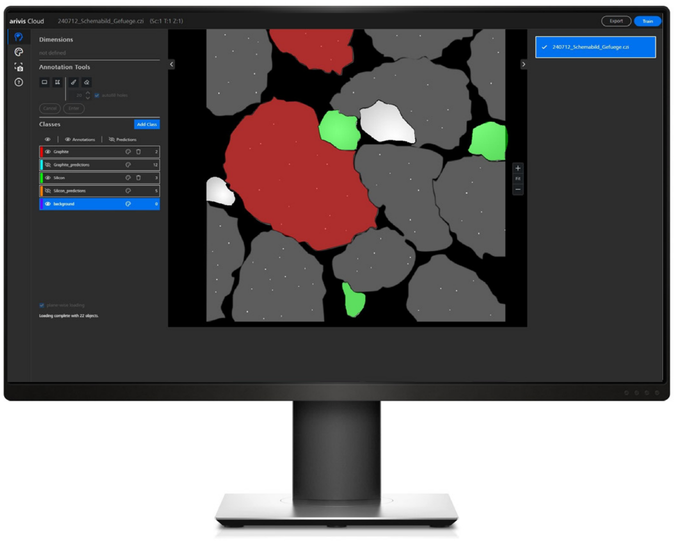


__Figure 7:__ arivis Cloud training – training image loaded, number of classes set, labeling of individual objects.

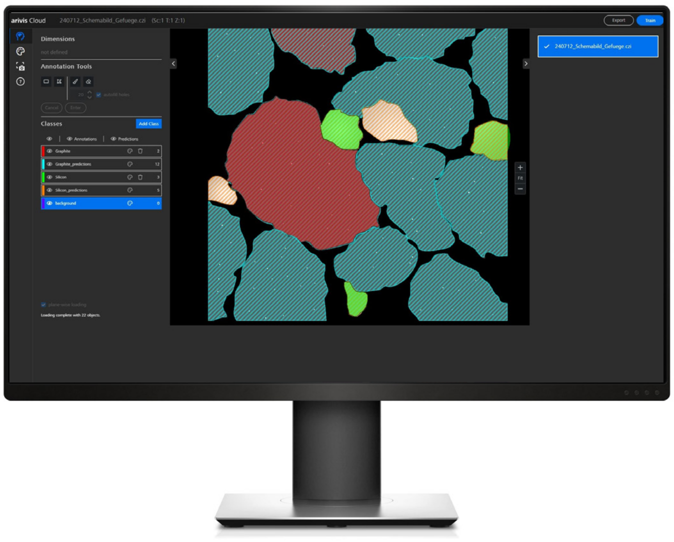


__Figure 8:__ arivis Cloud training result – particles with satisfactory segmentation and separation.

Figure 7 shows the labeling process using arivis Cloud. Similar to the process in ZEN Intellesis, a set of training images is uploaded to the cloud. After uploading, the classes to be analyzed are created. For this segmentation technique, only the classes that are to be analyzed need to be created and labeled. The labeling process then works like the one in ZEN Intellesis with an adjustable brush tool or a polygon tool. When labeling instance segmentation for example, it is crucial to precisely trace the individual objects with the brush or polygon tool, but objects can also overlap. For arivis Cloud training, more labels mean a more robust model. It is therefore recommended to label at least 50 objects per class.

Object Separation

As mentioned above, thresholding and ZEN Intellesis segmentation are both semantic segmentation techniques where each pixel in the image is assigned to a phase and not to an individual object. This problem, along with the fact that some individual particles touch or overlap each other, inevitably leads to merging of particles in the segmentation results. To illustrate this issue, Figure 9a and 9b show the results from threshold segmentation (Figure 9a) and ZEN Intellesis segmentation (Figure 9b) where the graphite particles are randomly colored to visualize the merging of some individual particles.

Object Separation

As mentioned above, thresholding and ZEN Intellesis segmentation are both semantic segmentation techniques where each pixel in the image is assigned to a phase and not to an individual object. This problem, along with the fact that some individual particles touch or overlap each other, inevitably leads to merging of particles in the segmentation results. To illustrate this issue, Figure 9a and 9b show the results from threshold segmentation (Figure 9a) and ZEN Intellesis segmentation (Figure 9b) where the graphite particles are randomly colored to visualize the merging of some individual particles.

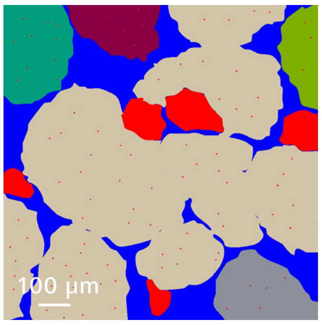


Figure 9a: *Threshold initial segmentation – graphite particles with random coloring, many particles segmented as one.*

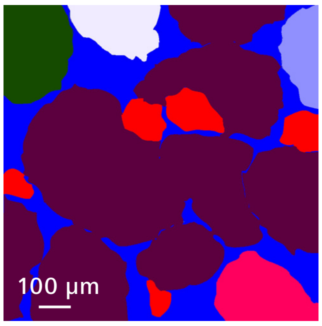


Figure 9b: *Intellesis initial segmentation – graphite particles with random coloring, many particles segmented as one.*

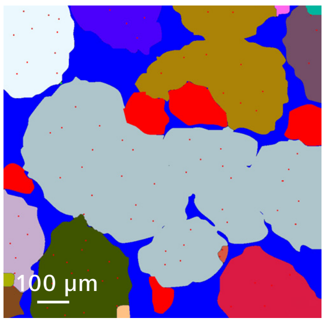


Figure 9c: *Threshold segmentation with Morphology separation filter – only some graphite particles separated, some particles split into multiple.*

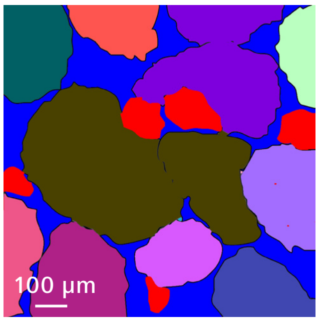


Figure 9d: *Intellesis segmentation with Morphology separation filter – only some graphite particles separated, some particles split into multiple. multiple.*

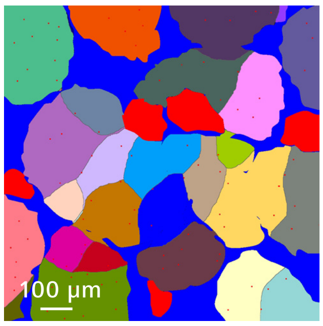


Figure 9e: *Threshold segmentation with Watershed separation filter – individual graphite particles split into multiple.*

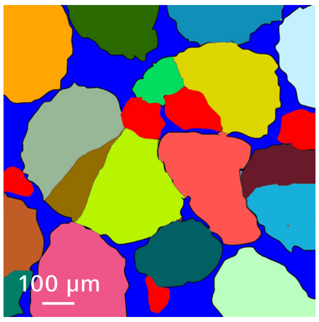


Figure 9d: *Intellesis segmentation with Watershed separation filter – some particles separated correctly, individual graphite particles split into multiple.*

To separate particles merged due to limitations in the semantic segmentation techniques, a post-processing step can be applied to reseparate the particles. The ZEN ecosystem provides two separation methods for this purpose, namely the Morphology and the Watershed function.

The Morphology function uses binary image operations to first reduce the objects until they separate and then enlarges the objects again, making sure they do not merge again. Figures 9c and 9d show the results of the Morphology function for the threshold segmentation (Figure 9c) and the ZEN Intellesis segmentation (Figure 9d). Compared to Figures 9a and 9b, there are more separated graphite particles in the image, but still many particles are merged together. Also, in the threshold segmentation, the Morphology function results in the splitting of objects that are one.

The Watershed function separates objects that have approximately the same shape. The result is two objects/ shapes separated by a 1-pixel boundary, while the rest of the perimeter remains unchanged. The Watershed method almost always results in the separation of elongated objects. This can be clearly seen in Figure 9e with the threshold segmentation results. Here, the bright speckles on the graphite particles are assigned to the silicon phase and thus create holes in the graphite particles, which the Watershed function interprets as individual shapes and separates. For the ZEN Intellesis segmentation (Figure 9f), the Watershed function also leads to the splitting of individual graphite particles, but the overall result is much better compared to the threshold segmentation.

Results

To show the differences of the three presented segmentation techniques on real-world samples, an image analysis was performed on a 1.2 × 0.8 mm² area of the anode shown in the first section. The analyzed image was acquired in a ZEISS FE-SEM with backscattered electron contrast. The resolution of the image was 279 nm/Pixel. Figure 10 shows a direct comparison of the three segmentation techniques after image analysis in the ZEN core Multi-Image View.

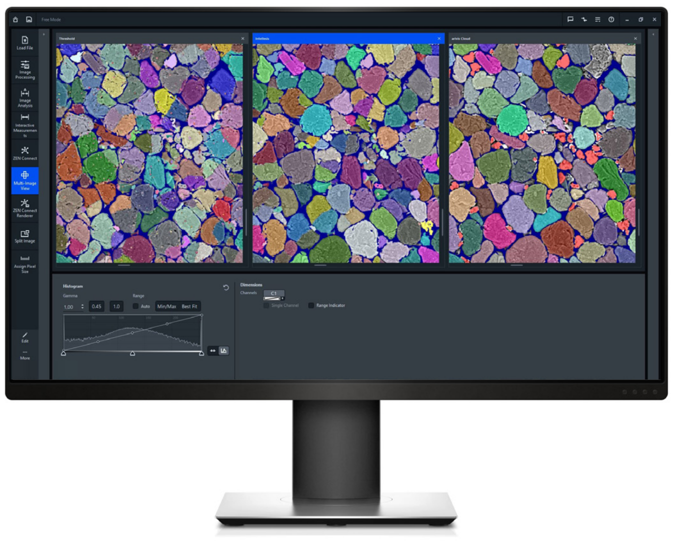


Figure 10: *ZEN core Multi-Image View of the results of the three segmentation techniques.*

From left to right (threshold segmentation, ZEN Intellesis, arivis Cloud instance segmentation) the improvement of the segmentation performance is clearly visible. The threshold segmentation shows all bright speckles as part of the silicon phase and a large number of falsely split graphite particles. The ZEN Intellesis segmentation shows a better representation of the real microstructure with the bright speckles on the graphite being segmented as part of the graphite, but there are still graphite particles falsely split into multiple objects. Only the instance segmentation results from the arivis Cloud show no false splitting and all bright speckles on the graphite are counted as part of the graphite.

For the image analysis performed in ZEN core, the analyzed parameters for each object and the whole class (field parameters) are presented in Table 1. Note that for the size distribution analysis, objects were filtered out if "is frame touched" was true for an object. The reason for this is that cut objects have to be measured for the phase fraction but cannot be measured for the size distribution.

Object parameters	Field parameters
Area	Regions count
Diameter (equivalent circular diameter)	Regions count per mm^2
FeretMax	Regions area
FeretMin	Frame area
FeretRatio	Regions area percentage
Is frame touched	

Table 1: *Measurement parameters*

First, the phase fraction and the particle/object count of each phase were analyzed. The results are shown in Figure 11. While the phase fraction does not show a significant difference between the three segmentation methods (Figure 11 left), the number of particles per phase depends very strongly on the segmentation method. For example, the high number of silicon particles in the threshold segmentation is due to the bright speckles on the graphite particles being segmented as silicon. Since these particles are very small, this false segmentation does not weigh too heavily on the phase fraction.

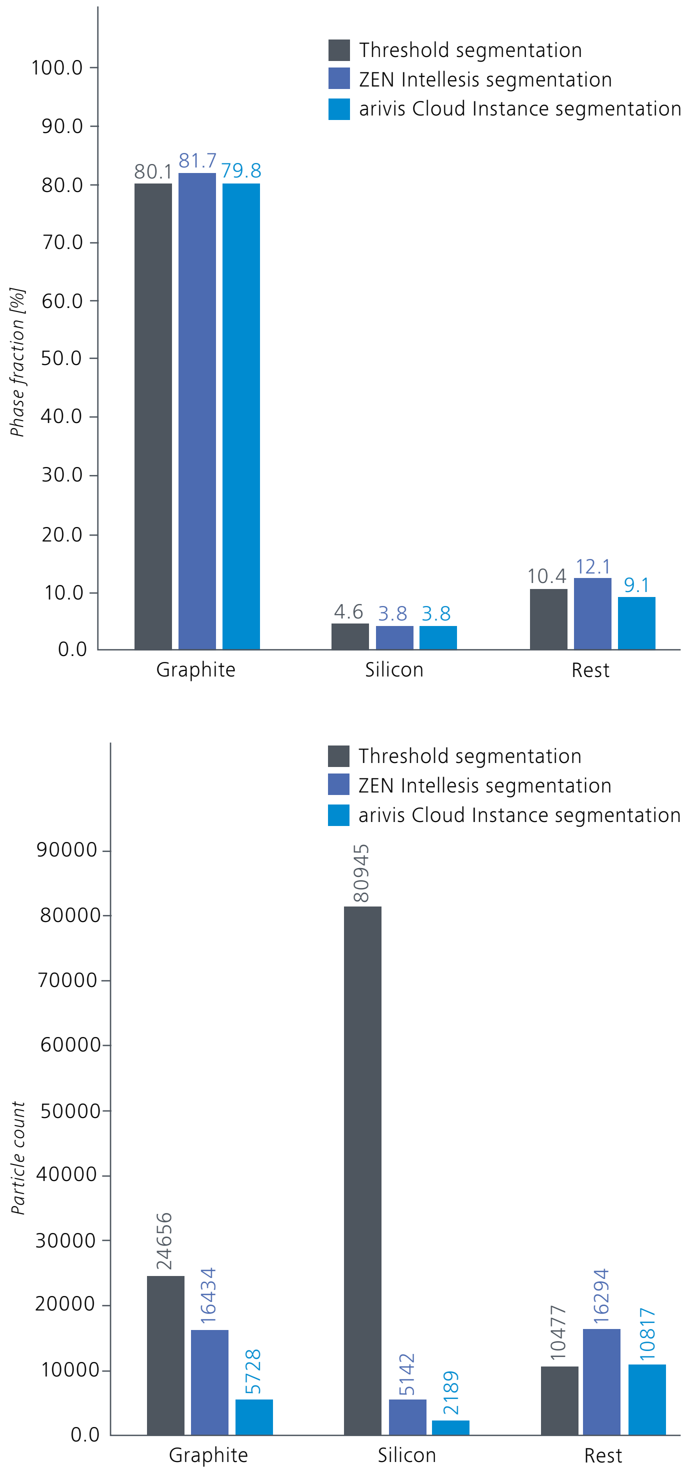


Figure 11: *Analyses of the field parameters phase fraction (area percentage) and particle count; strong influence of segmentation method on particle count.*

The number-weighted equivalent circular diameter (ECD) analysis of the graphite particles in Figure 12 also shows the strong influence of the segmentation technique on the results. All three segmentation techniques reveal a large number of small graphite particles. Only the arivis Cloud instance segmentation shows an almost Gaussian distribution of the particle size classes, but still presents a distinct peak of small particles. For the graphite class, this difference results from the object separation functions performed on the two semantic segmentation methods, where many graphite particles are split into a number of smaller objects. The peak of small particles in the arivis Cloud segmentation will be discussed in the following section.

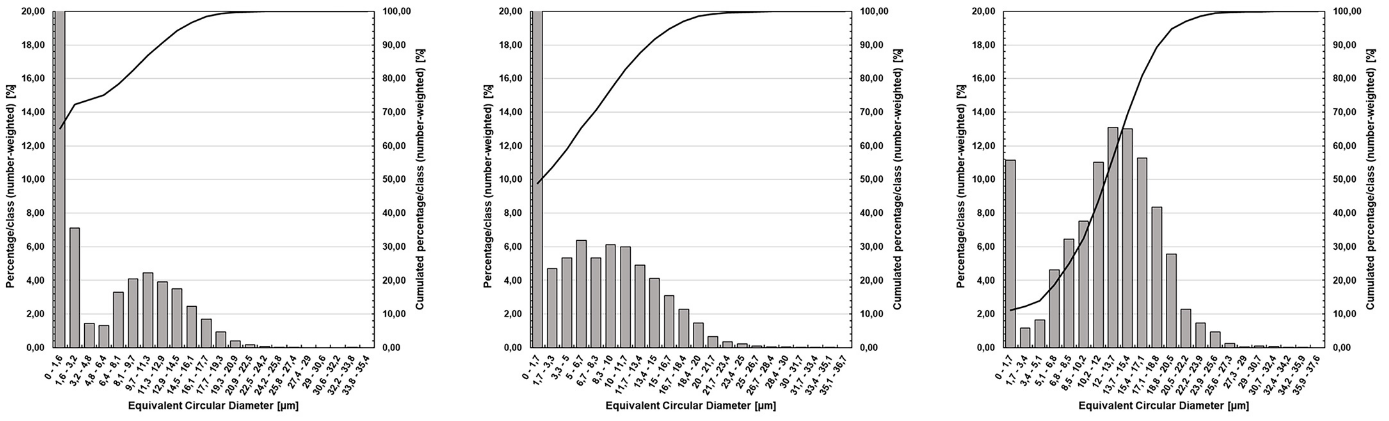


Figure 12: Comparison of the number-weighted analysis of the equivalent circular diameter (ECD) of the graphite particles.

Refining Analysis Results

As shown in the previous section, the two semantic segmentation techniques in combination with a separation function do not lead to viable results with regard to particle size distribution. The analysis from the arivis Cloud segmentation also shows a significant peak in small graphite particles, although it does not have the problem of false object splitting.

As a first step to look deeper into the analysis results and the possible cause of the large number of small objects, the size class range was increased from 10 to 100, resulting in a more detailed view of the size distribution of the graphite particles. Figure 13 shows the number and area-weighted results of the arivis Cloud segmentation results with 100 ECD size classes.

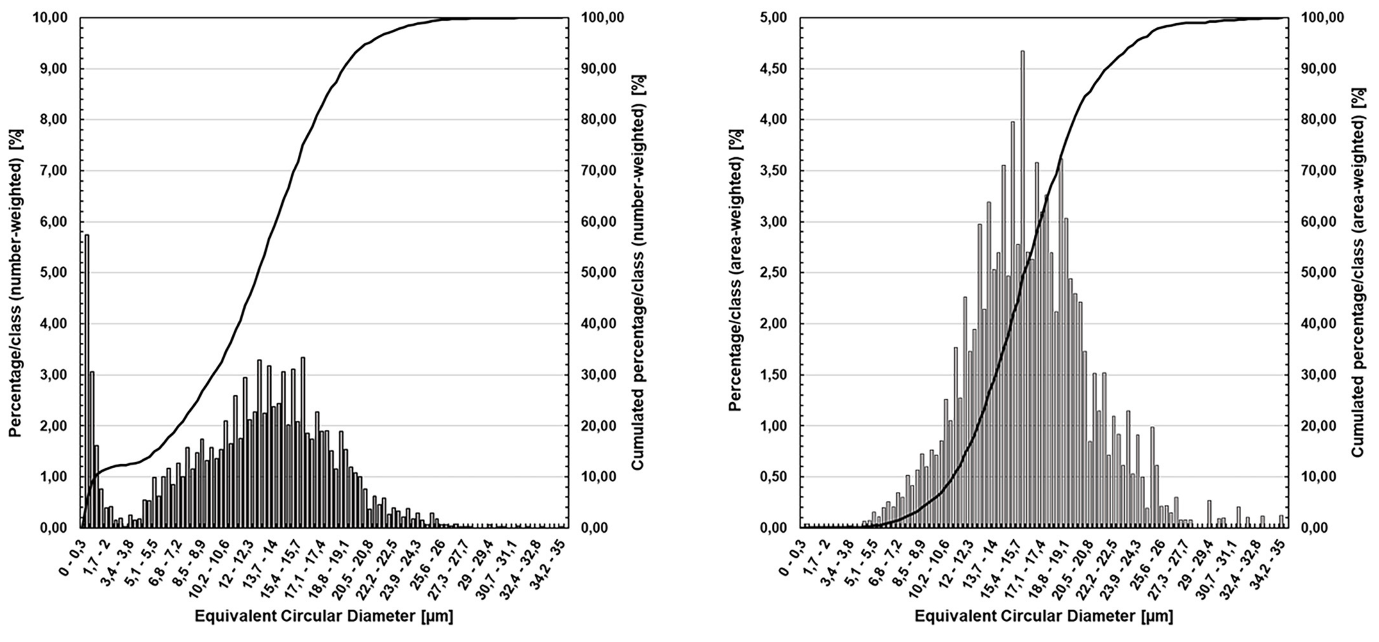


Figure 13: Graphite particle size distribution analysis with 100 ECD size classes for arivis Cloud segmentation.

The higher-resolution particle size distribution with 100 ECD size classes reveals an almost bimodal size distribution of the graphite particles with nearly 6% of the particles being in the <0.3 µm ECD size class. This is not visible in the area-weighted analysis because even a high number of very small particles does occupy much area. Given the high number of small particles in the <0.3 µm class, and knowing that the image pixel size is 0.279 µm/Pixel, it must be assumed that these particles are 1-Pixel objects resulting from a segmentation error.

With this in mind, and the rule of thumb that an object to be measured has to be represented by at least 10 pixels to minimize the measurement error (10-Pixel-Rule), the raw data from the ZEN core image-analysis workflow needs to be filtered and replotted again with the smallest class starting at 3 µm ECD. The fitted analysis results are displayed in Figure 14.

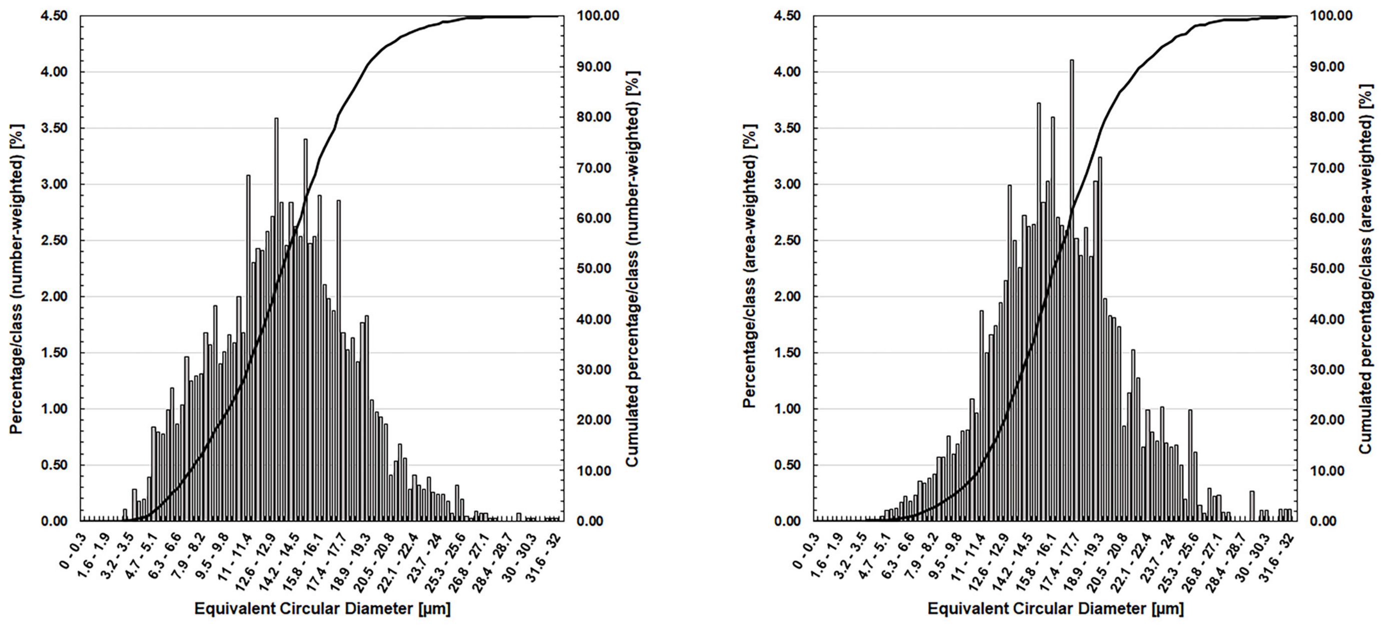


Figure 14: *Graphite particle size distribution analysis with 100 ECD size classes and filtered for objects >3 µm ECD for arivis Cloud segmentation.*

The final result shows a wide but homogeneous distribution of graphite particle sizes with no outliers. The results obtained can now be used for quality control and or process development.

Conclusion

This paper shows that high-performance instance segmentation using arivis Cloud is a highly suitable tool for the grain or particle size analysis of microscopy images. With an appropriately trained model, it can segment individual objects, grains, and particles easily and reproducibly, without the need for object separation functions that are difficult to adapt. This approach can also be used for all kinds of grain size analyses in material science that were previously not possible due to limitations in segmentation techniques.

Carl Zeiss Microscopy GmbH
07745 Jena, Germany
microscopy@zeiss.com
www.zeiss.com/microscopy

Follow us on social media: