doi:10.1017/S1431927617001349

Dragonfly SegmentationTrainer - A General and User-Friendly Machine Learning Image Segmentation Solution

Nicolas Piche[1], Isabelle Bouchard[1], Mike Marsh[1]

[1.]Object Research Systems. Montreal, Canada

For many microscopy imaging techniques, image segmentation remains a bottleneck that impedes efficient quantitative analysis, and, consequently, it slows many imaging-based research projects. A number of sophisticated segmentation techniques have been proposed, both manual and automated. Even in cases where they have relied on advanced techniques [1], they are often highly-specialized and usually tuned to meet the needs of highly constrained image analysis problems. To address the shortcoming, we present a generalized machine learning platform that is suitable for a large class of research imaging applications.

Incorporated into our image processing platform, Dragonfly, the Segmentation Trainer is a machine learning tool that utilizes textural details in the image to discriminate between different material phases. The platform provides users a solution for developing, training, and applying classifiers to segment images. A classifier can be treated as a black box that takes one or more images as input and returns a complete image segmentation. Once developed and trained, a classifier can be endlessly reused and its internal details can be ignored.

A classifier is defined by: a fixed number of named input images, textural features associated with the input images, a pixel gridding strategy, an input reference segmentation, and a machine learning engine. Once all of these parameters have been set, the classifier is trained and ready for use.

It's helpful to consider a specific example. Suppose a user has an imaging application that requires both secondary electron (SE) and backscattered electron (BSE) SEM images for optimal segmentation because the BSE image offers superior material contrast, but the SE image offers superior edge contrast. After naming the classifier, a user would start by specifying that his or her classifier requires two input images, SE and BSE. Next, in order to harness the edge details, the user may associate edge texture features with the SE image and intensity texture features with the BSE image. Any of Dragonfly's available image filters can be used (with various parameters) to define feature presets, thus building up a texture bank of features for reuse with different classifiers. These features can be associated directly with the image pixels, or with image pixel regions, such as superpixels. Using pixel regions allows for the aggregation of local statistics which can be more discriminating. These pixel gridding strategies are also modular and subject to end-user extension.

Next, the user would manually label small areas of the image—using paint tools—to identify each of the represented classes to segment. This provides a reference segmentation to train the classifier. Lastly, the user would select and parameterize the machine learning engine. The architecture of our solution permits a variety of machine learning engines to serve as the backbone of the tool. In the interest of continued innovation, we have made that backbone modular, and for our first implementation we deliver six different machine learning engines (KNeighbors, Random Forest, AdaBoost, Bagging, Extra-Trees, Gradient Boosting) based on the implementation in the open source library scikit-learn [2].

https://doi.org/10.1017/S1431927617001349 Published online by Cambridge University Press

After all of these parameters are set, the user can train the machine and visually evaluate the results. The relative weights of every feature are reported. If features carry almost no weight, they can be discarded to improve compute efficiency. If the results are unsatisfactory, different pixel gridding strategy may be selected, or different features may be added. The user may iterate until good results are achieved. When the results are satisfactory, the classifier training is complete. The user may apply his or her classifier any time he or she has BSE and SE images to provide as input.

Training classifiers can benefit other users so we have launched and online repository for sharing classifiers with others in the community for broader use and continued refinement to improve accuracy. The online repository is called Infinite Toolbox and behaves in ways similar to Android's Google Play or iOS's App Store, in that users can connect directly from the platform and browse for plugins and other extensions, particularly classifiers.

We illustrate with successful classifier examples from materials, life science, and geoscience. Deep integration of classifiers into more complicated workflows and integrations with Dragonfly's macro recording engine permit users to apply sequential actions in reproducible but flexible ways, further opening up novel ways to use this segmentation tool.

References:

[1] AJ Perez, Front. Neuroanat **8** (2007), p. 1.
[2] Pedregosa *et al*, Jnl Mach Learn Rsrch **12** (2011), p. 2825-2.

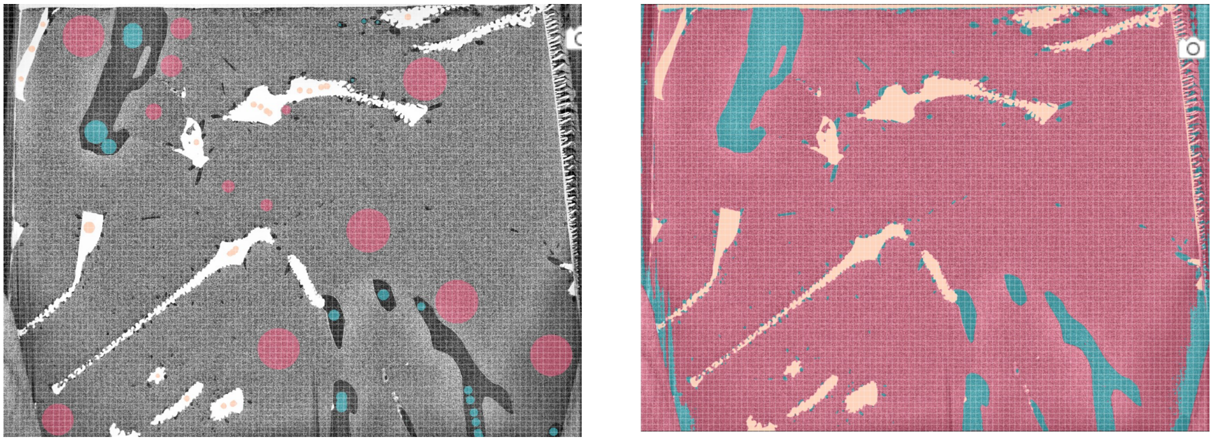

Figure 1.
Left: Greyscale micrograph with select colored areas represented the pixels that were used for training data on three different classes or materials. Right: The complete micrograph colored entirely according to the classifier built from the training data.

https://doi.org/10.1017/S1431927617001349 Published online by Cambridge University Press