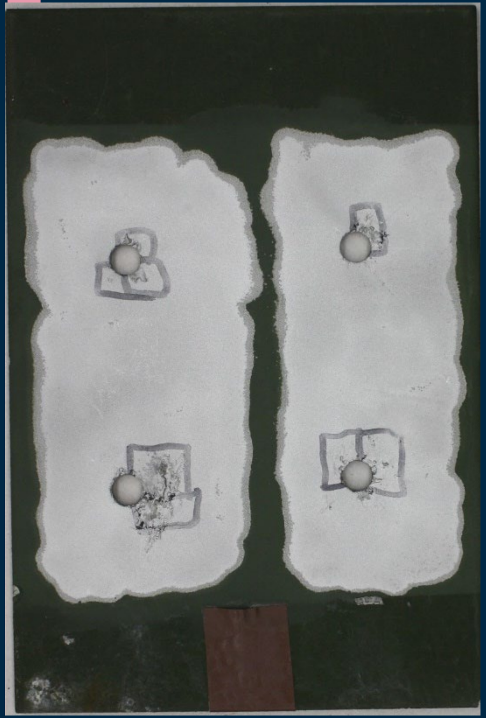
# Machine Learning Corrosion Area Analysis

Prepared by: Elzly Technology Corp

# Introduction

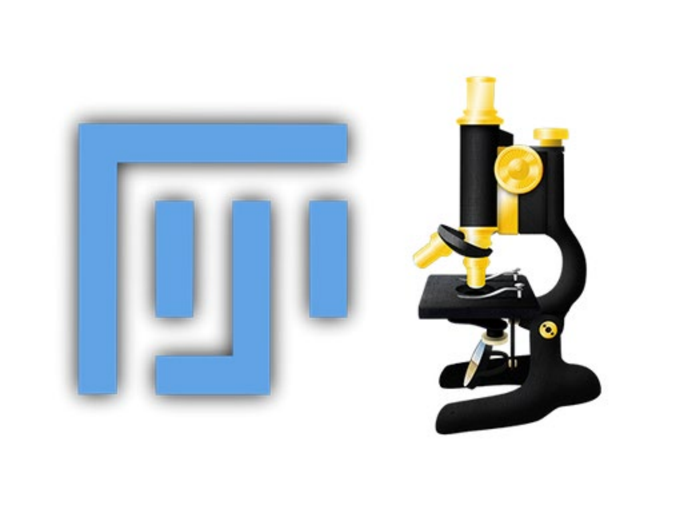
- There is a desire for a method to quickly and accurately analyze corroded areas after corrosion testing
- This analysis will add a quantifiable metric to the corrosion observed at these interfaces which has been notoriously difficult to quantify
- Physical analysis methods are inaccurate, vary from person to person, and require significant man-power for large data sets



Physical Area Measurements

# Method

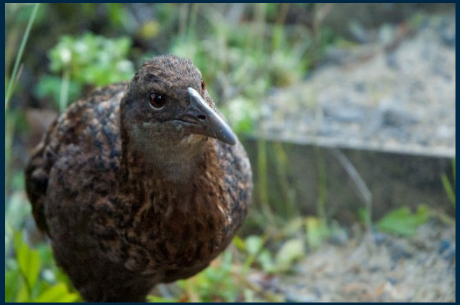
- ImageJ2 image correction and analysis program – free for use
- Utilizing Fiji
  - Advanced image processing plugins for ImageJ2
- Specifically Trainable Weka Segmentation Model



Source:
Arganda-Carreras, I., Kaynig, V., Rueden, C., Eliceiri, K. W., Schindelin, J., Cardona, A., & Sebastian Seung, H. (2017). Trainable Weka Segmentation: a machine learning tool for microscopy pixel classification. *Bioinformatics*, *33*(15), 2424–2426. doi:10.1093/bioinformatics/btx180

# Trainable Weka Segmentation

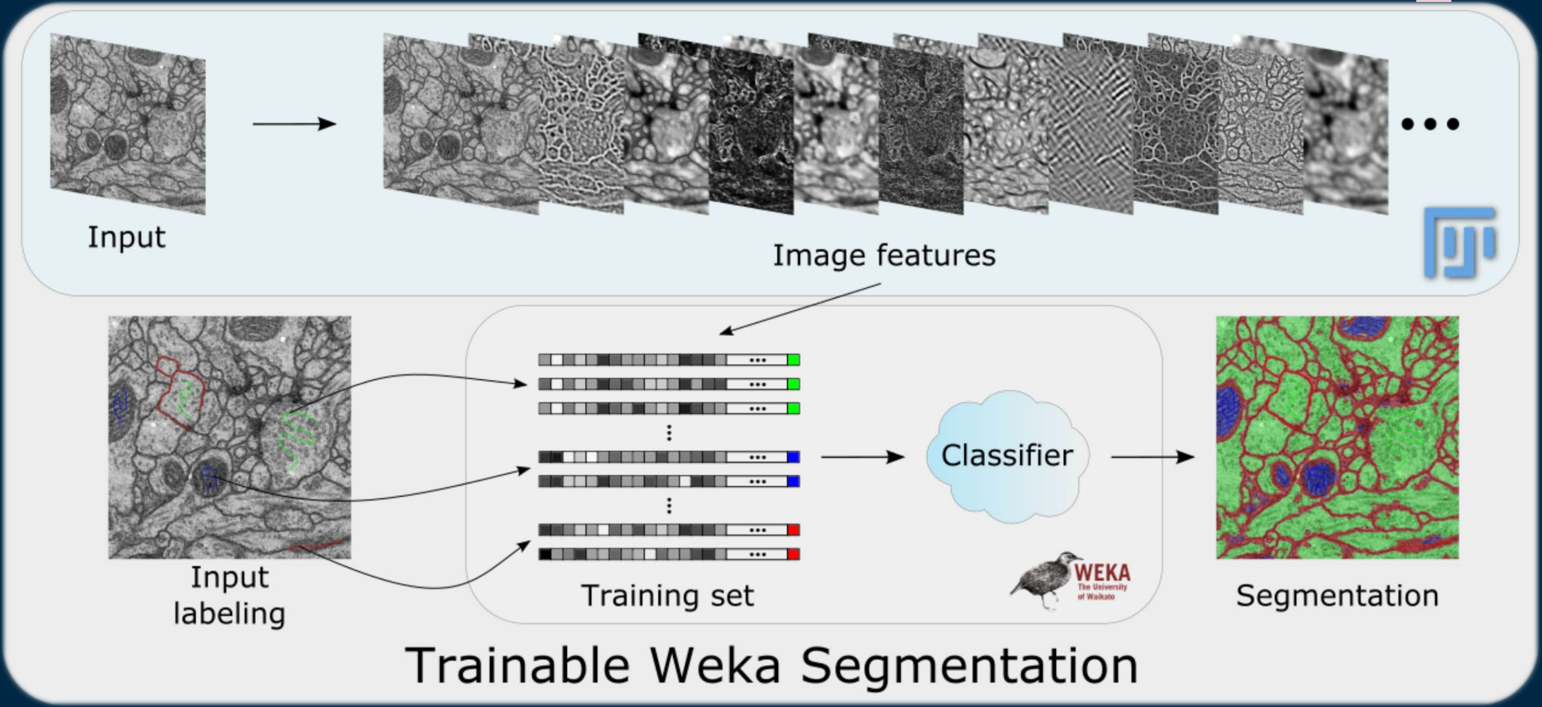
- WEKA (Waikato Environment for Knowledge Analysis)
  - Open-source ML algorithms used for data classification and regression
  - Reads images as 8-bit greyscale or 8-bit RGB pixel data
- Trainable
  - Plugin can be taught to identify different aspects of an image based on different pixel patterns
- Segmentation
  - Algorithm chooses a category label for every pixel in the image
  - Labeled results are based on the way it has been trained by the user



Weka (bird)

Weka (ML software logo)

# Trainable Weka Segmentation



# Pre Processing

## Contrast Boost

Makes blacks and whites in image more pronounced increasing model accuracy

## Remove Outliers

Removes any excesively bright areas that would be difficult for the model to process

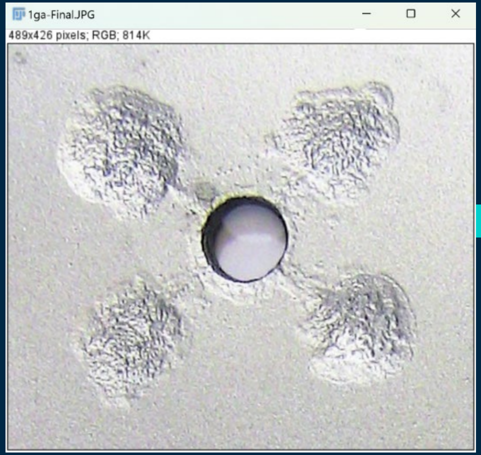
## Unsharp Mask

Increased the sharpness of the image without reducing clarity

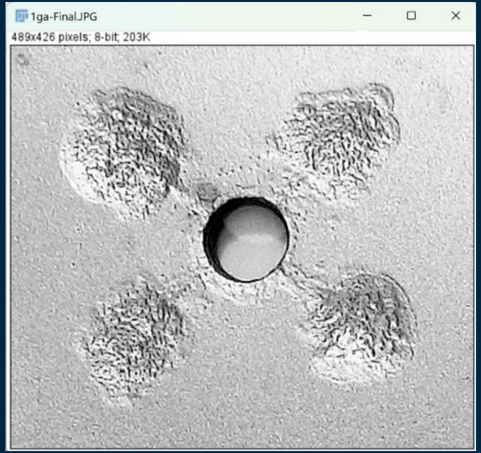
## Make 8bit

Gives us a pseudo mono-chrome image that is easier for the model to process

All performed automatically within macro

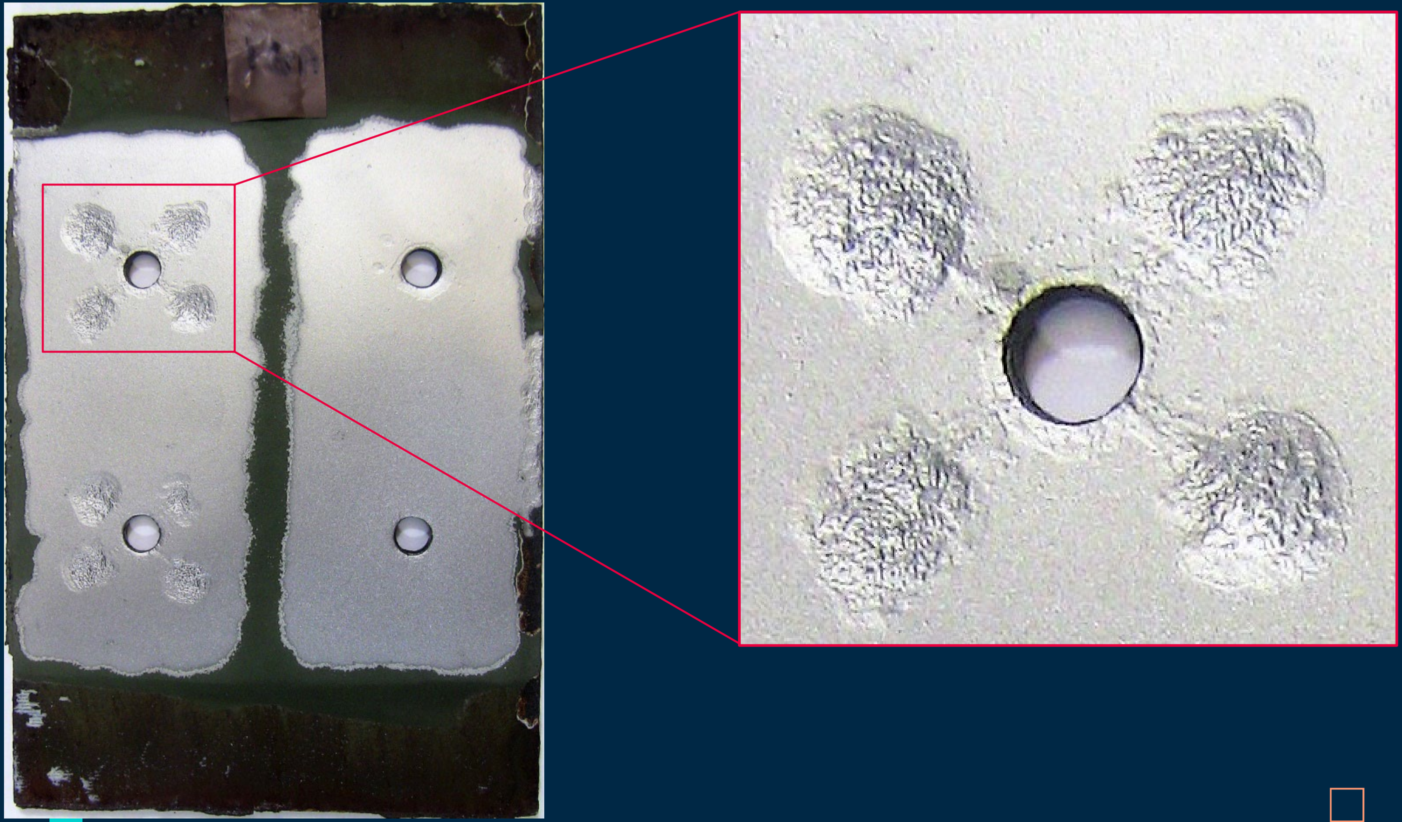



Before Pre-Processing

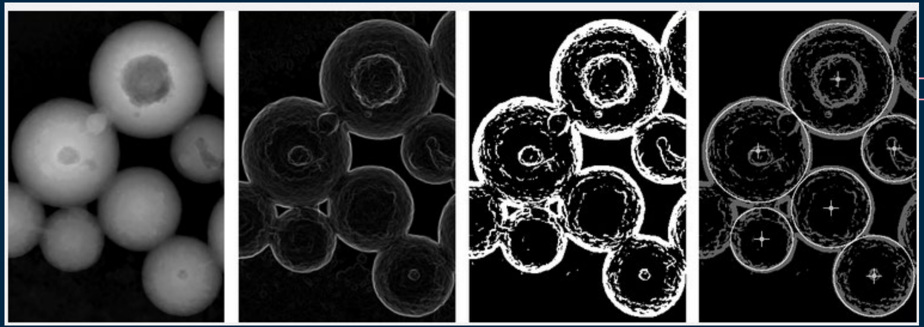



After Pre-Processing

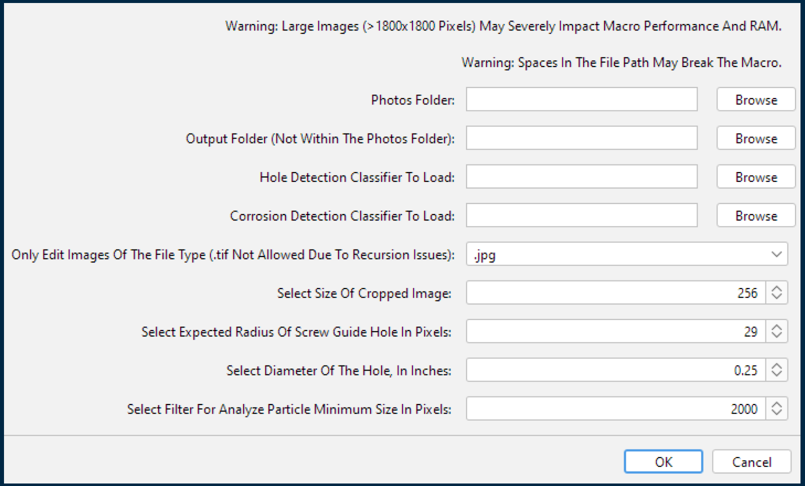
# Starting Image (Formerly)



# Automation



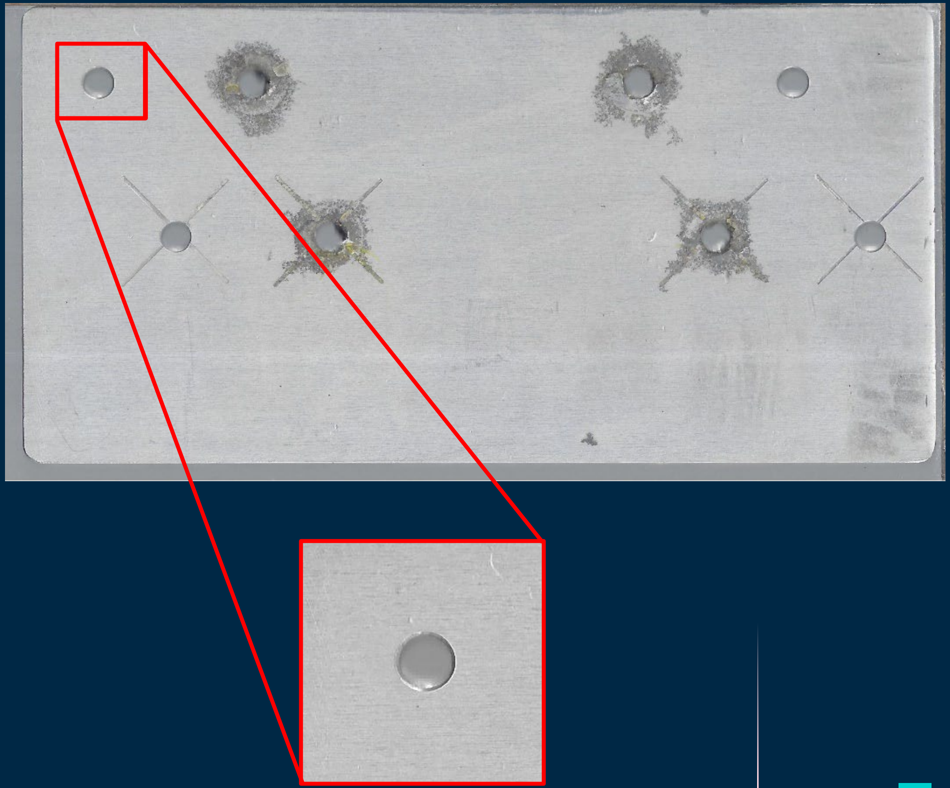
- FIJI supports macro scripting
  - Allows for a user interface to be generated, allowing the user to provide information such as the hole diameter, desired image size, etc.
- Hough Circle Transform
  - FIJI Plugin offered by the UCB Vision Sciences library
  - Can be used to identify circles, can be constrained to a radius
  - Useful but ran into problems with false-positive circles, shadows or corrosion cause holes to sometimes not be circular enough
- Switched to a two -stage Weka segmentation method



# Starting Image

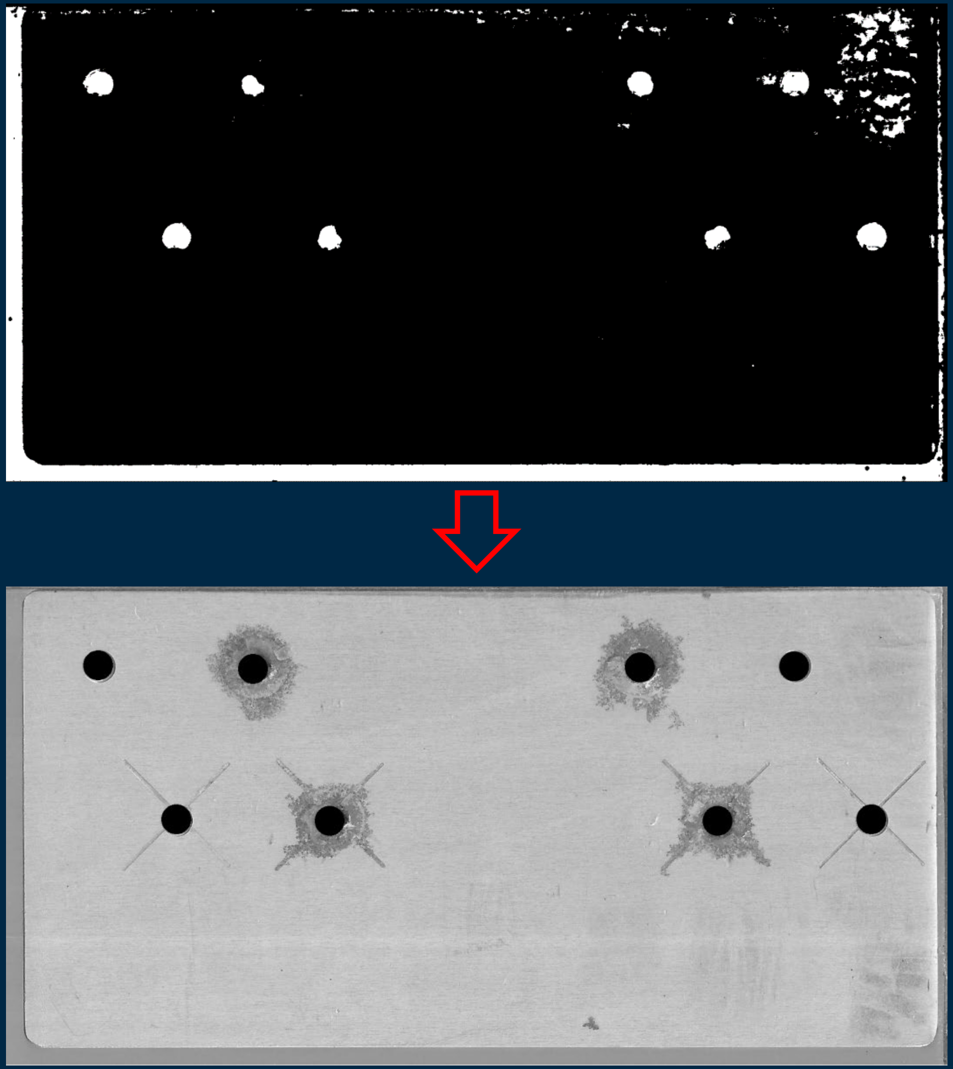
- Automatic pre-processing, hole identification and cropping

- User decides the size of the square cropped photo
    - Requires all regions to be around the same size



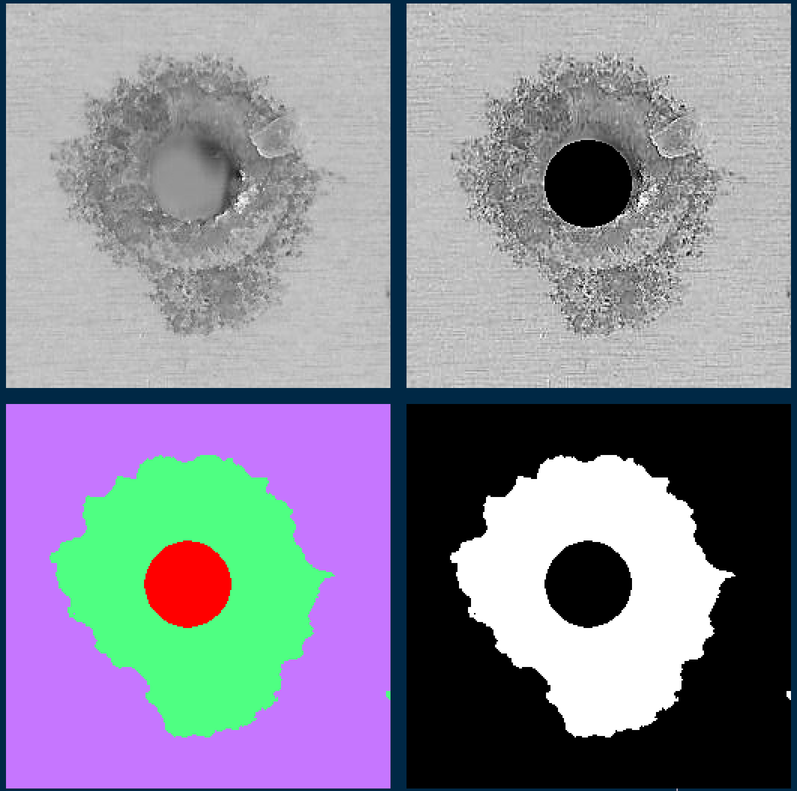
# Process (Full Image)

- Weka
  - Identifies holes and other areas
  - Filter 'particles' by circularity
- Masking
  - Perfect circle mask over the screw holes
  - Less sensitive to distortion from lighting and corrosion effects than Hough Circle Transform



# Process (Cropped Image)

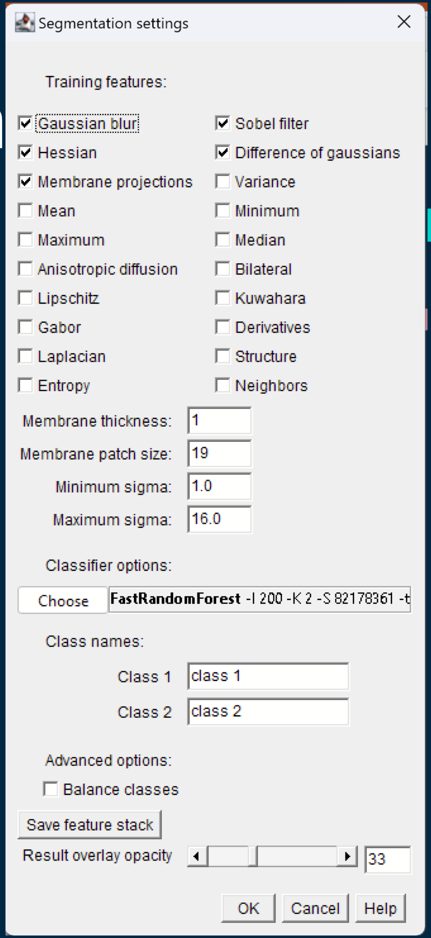
- Weka segmentation performed using 3 classes/categories
- Binary filter applied to isolate only corrosion area for analysis
- Scale determined by known, user-provided screw hole mask diameter



# Reevaluation of Segmentation Settings

- By default, FIJI uses edge detection methods that are better for sharp contrast differences
  - Significant training data required
- Training using texture description focuses on variations in pixel patterns and noise removal features
  - Both improve the model and reduce the training needed
  - Significant improvements in classification speeds*

    *slowed down due to full -image segmentation in 1st stage
- Additional classifier options for the decision trees in the algorithm also available




Source: Trainable Weka Segmentation User Manual, https://oup.silverchair-cdn.com/oup/backfile/Content_public/Journal/bioinformatics/33/15/10.1093_bioinformatics_btx180/4/bioinformatics_33_15_2424_s1.pdf?Expires=1695735312&Signature=g6wuu3Y0bWsRBzo4cf9ZDF0qrZMh3yVIC~Fq0jTDlyjyXhK4BSohfWP~lotIPvqcqqRsJlSpeLHhUQ086N4SXrPtPsvUF7Vrv1VcqJ4UDGyCXZBzHx~FJXw6sipLAGxKavjzzFWsMkjY~K9ftBmdz~Uq5t4Z5G0w-w7rS8jHIXk-oAORERpNvzARQuy1jSOO4O0sqgUMXrUSZAwqU42ZxxM1gavs5PM7cl7Na0~OF5QqCrc8J6IBflJwcsgiC32thVSKDJyaCBOsOL0yVMORKYXSagep85JWrTCyGs1p0l8T4dlW7Rg6bKefqbmadZN5-0itWZYCDWcUzeJWOcBIlA__&Key-Pair-Id=APKAIE5G5CRDK6RD3PGA

# Recent Developments

Auto-crops photos centered on t he hole locations for any size necessary

Autonomous macro to analyze a directory of photos instead of one at a time

Automatically identifies and masks over the drilled screw holes

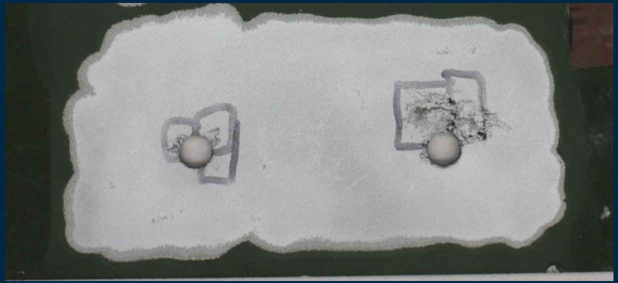
Analysis of ~2MP images without need for cropping or downscaling

Dialog box prompts user for input parameters for hands-free image processing

# Area Analysis

| Sample | Substrate | X-cut? | Avg Weka Area ($mm^2$) | Avg Physical Measure ($mm^2$) | % Difference |
|---|---|---|---|---|---|
| 1GA | Steel | Y | 369.6 | 388 | -5% |
| 1GA | Steel | N | 33.8 | 7.5 | 352% |
| 1GB | Steel | Y | 219.6 | 286.5 | -23% |
| 1GB | Steel | N | 29.0 | 5.5 | 428% |
| | | | | Avg Steel % Difference | 188% |
| 7GA | Aluminum | Y | 12.9 | 3 | 330% |
| 7GA | Aluminum | N | 104.8 | 9.5 | 1004% |
| 7GB | Aluminum | Y | 24.5 | 3 | 717% |
| 7GB | Aluminum | N | 24.8 | 2.5 | 894% |
| 7GC | Aluminum | Y | 46.4 | 2 | 2223% |
| 7GC | Aluminum | N | 36.4 | 0.5 | 7190% |
| 8GA | Aluminum | Y | 54.8 | 3.5 | 1467% |
| 8GA | Aluminum | N | 68.0 | 6 | 1034% |
| 8GB | Aluminum | Y | 123.5 | 87.5 | 41% |
| 8GB | Aluminum | N | 139.9 | 126 | 11% |
| 8GC | Aluminum | Y | 31.2 | 2 | 1465% |
| 8GC | Aluminum | N | 71.6 | 52.5 | 36% |
| | | | | Avg Aluminum % Difference | 1368% |

- Physical measurements are more accurate on steel substrates with X-cuts
- Physical measurements are more difficult on smaller areas of corrosion, this increases the % difference on aluminum substrates
- Weka model can measure corrosion that is difficult to see with the human eye, or difficult to quantify accurately



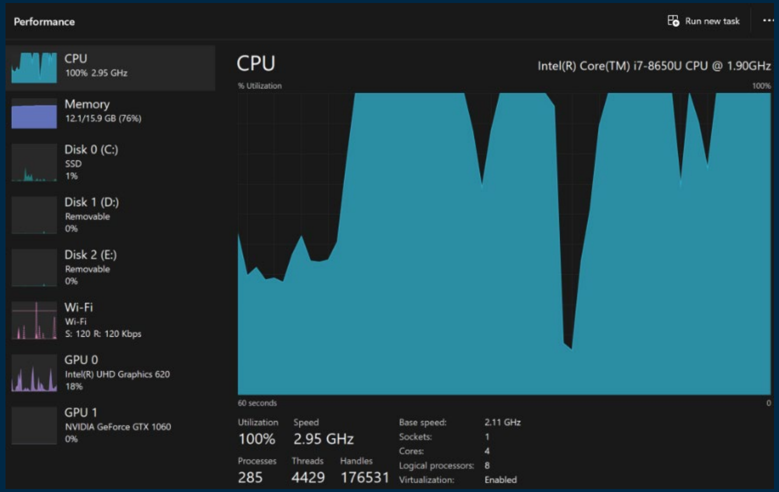
# Extended Work

- Finetuning of Weka segmentation decision tree settings
  - Have kept mostly to default
  - Difficulty in evaluating time and accuracy impacts for each parameter without a lot more testing
- PyTorch ML framework
  - Uses GPU multithreading
  - Existing open-source ML neural network models
  - Pre-trained, can be retrained on a small(-ish) dataset to reduce training requirements*

    *still generally prefers to have more data to train and test on

# Limitations of ImageJ

- CPU-based ImageJ programming
  - Caps out even on strong laptop CPUs
  - Touching any of the windows tends to crash the macro
  - Slowly does each image step-by-step
- GPU-based CLIJx Weka plugin
  - Less usage and documentation
  - Mostly just crashed during testing
- PyTorch uses both CPU and GPU
  - Hard limitation of GPU VRAM memory
  - Compensated by improved process speeds




# Future Work

- Variations in the lighting make it difficult to determine impact
  - Further work improving model to be more robust, allowing for varying lighting environments
  - May also reduce overtraining issues
- Improve PyTorch Neural Network method
  - Further work exploring confidence and accuracy of model predictions
  - Ways to allow for higher resolution images to be examined

# Alternative Methods

- High Resolution 3D surface visualization
  - GelSight
  - Optical/laser 3D Profilers
  - Lidar scanning
- These methods could increase accuracy and add Z-axis data like pit depth relatively easily
  - Access to these proprietary software's could be invaluable


https://www.youtube.com/watch?v=3VtOKSLYjsE