SAND2018-7988PE

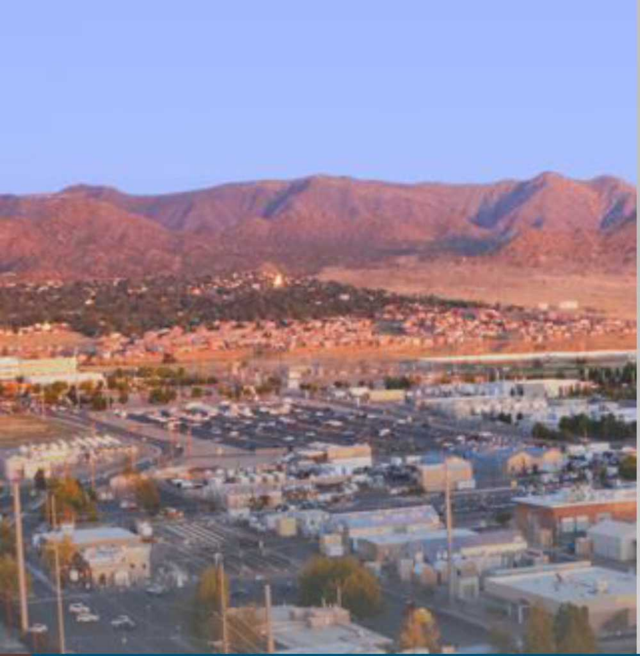


Image Analysis of Micro CT Scans of Fractures in 3D Printed Samples

Jack Ringer & Vanessa Brock –AHS and CHS
STAR High School Summer Intern Program

Hongkyu Yoon- Org. 8864

Hongkyu Yoon – Org. 8864

Biography

Jack Ringer

- Currently attending Albuquerque High School, plan to graduate in May 2019
- Plans on attending university to pursue a degree in engineering/computer science

Vanessa Brock

- Currently attending V. Sue Cleveland High School, plans to graduate in 2019
- Plans on attending NM Tech to pursue a degree in cyber security

Introduction

Why 3D printing?

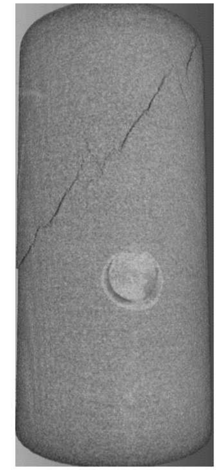
- 3D printing of fractured and porous analog geomaterials has the potential to enhance hydrogeological and mechanical interpretations by generating engineered samples in testable configurations with reproducible microstructures and tunable surface and mechanical properties.
- Overcome sample-to-sample variability for testing material response

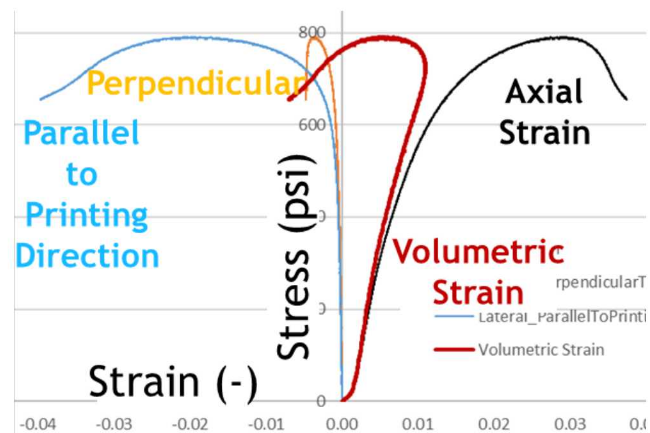
Gypsum powder-based 3D printing

- Print cylindrical core samples in three different directions to evaluate the impact of anisotropy on mechanical properties

Mechanical testing and MicroCT scanning

- Printed samples were tested for compression strength and tested samples were 3-D imaged with microCT scanning





Purpose

- Analysis of microCT (micro-computed tomography) images of fractures in tested 3D printed samples
- Quantitative image analysis of fractures is technically challenging due to complexity of fracture geometry and features of microCT images of 3D printed samples
- It is very important to evaluate the applicability of existing image analysis algorithms for this particular image sets

Method: Fiji / ImageJ

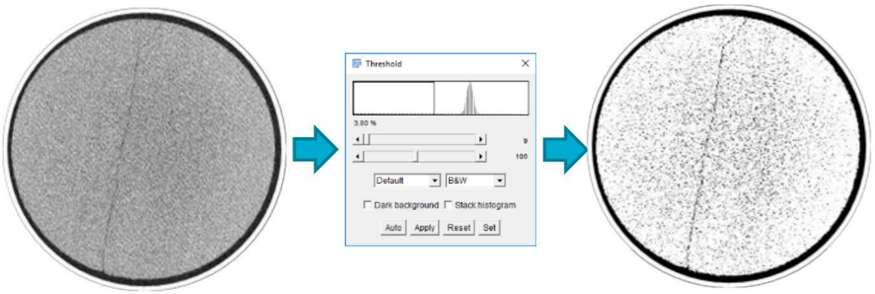
- ImageJ is an open source image processing application written in Java.
- Fiji (Fiji is Just ImageJ) is a distribution of ImageJ with several useful plugins pre-installed.

Schindelin, J.; Arganda-Carreras, I. & Frise, E. et al. (2012), "Fiji: an open-source platform for biological-image analysis", Nature methods ***9(7)****: 676-682, PMID 22743772, doi:10.1038/nmeth.2019* (on Google Scholar).

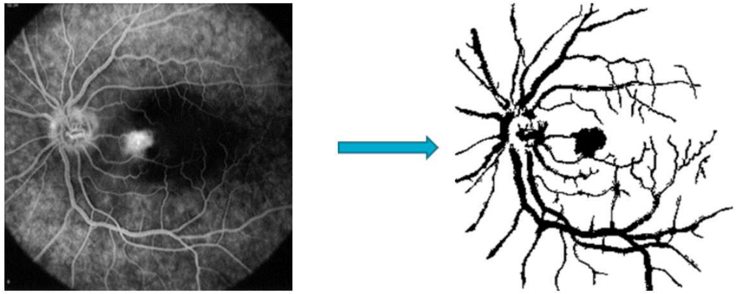
Rueden, C. T.; Schindelin, J. & Hiner, M. C. et al. (2017), "ImageJ2: ImageJ for the next generation of scientific image data", BMC Bioinformatics ***18:529****, doi:10.1186/s12859-017-1934-z.*

Image Process: Segmentation

- **Thresholding**
  - The most common way to segment an image is through thresholding
  - Thresholding segments an image into background and foreground based upon a threshold value(s) defined by the user or calculations of different algorithms
  - Thresholding is purely based upon pixel values, which is why preprocessing is often necessary prior to thresholding



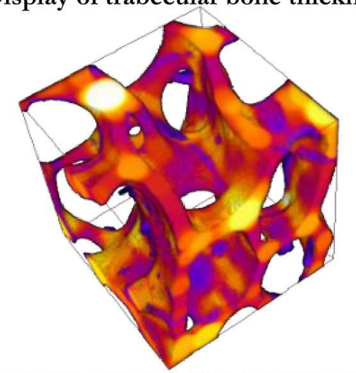
- **Trainable WEKA Segmentation**
  - WEKA stands for Waikato Environment for Knowledge Analysis
  - TWS is a tool which can take in user-provided annotations in order to create classifiers that can automatically interpret image information.
  - Utilizes machine learning algorithms in order to interpret user input and produce image segmentations.



Arganda-Carreras et al. (2017)

BoneJ Plugin

Display of trabecular bone thickness



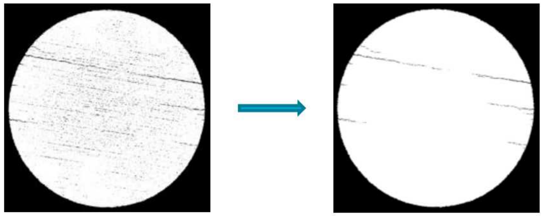
Doube M, Kłosowski MM, Arganda-Carreras I, Cordeliéres F, Dougherty RP, Jackson J, Schmid B, Hutchinson JR, Shefelbine SJ. (2010) BoneJ: free and extensible bone image analysis in ImageJ. *Bone* 47:1076-9. doi: 10.1016/j.bone.2010.08.023

BoneJ

- BoneJ is a plugin for ImageJ designed for bone image analysis.
- It has proven useful in the realm of geoscience, as bone fractures and rock cracks often share similar distinguishing features.

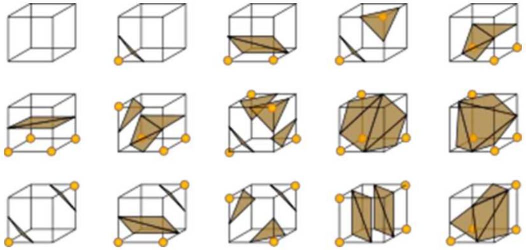
Purify

- Purify is a method within BoneJ which uses connectivity analysis (assumes there is only one foreground particle and no cavities) in order to remove small foreground and background particles.



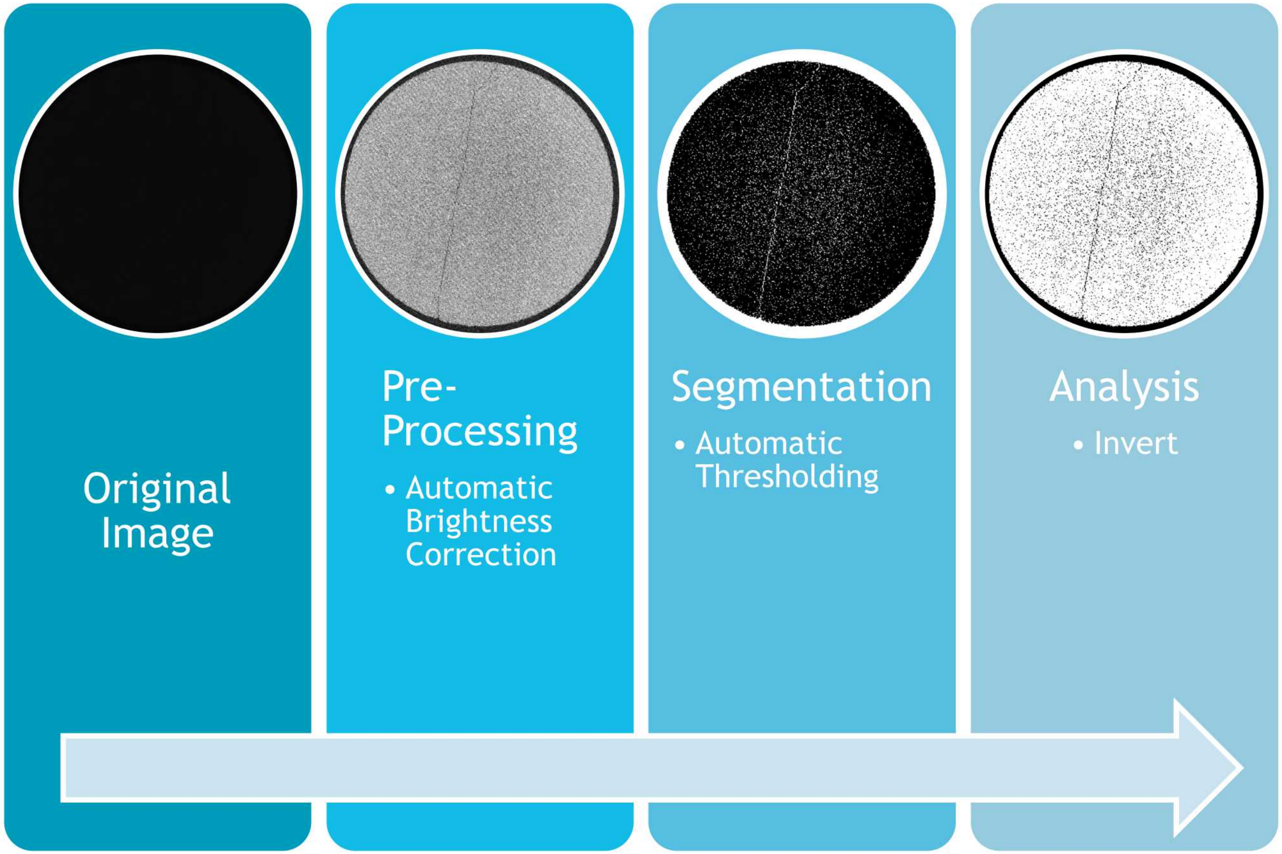
Isosurface

- Constructs a triangular surface mesh via the marching cubes algorithm (from Lorensen and Cline)
- Used to construct 3D representations of 2D data (stacks of images).



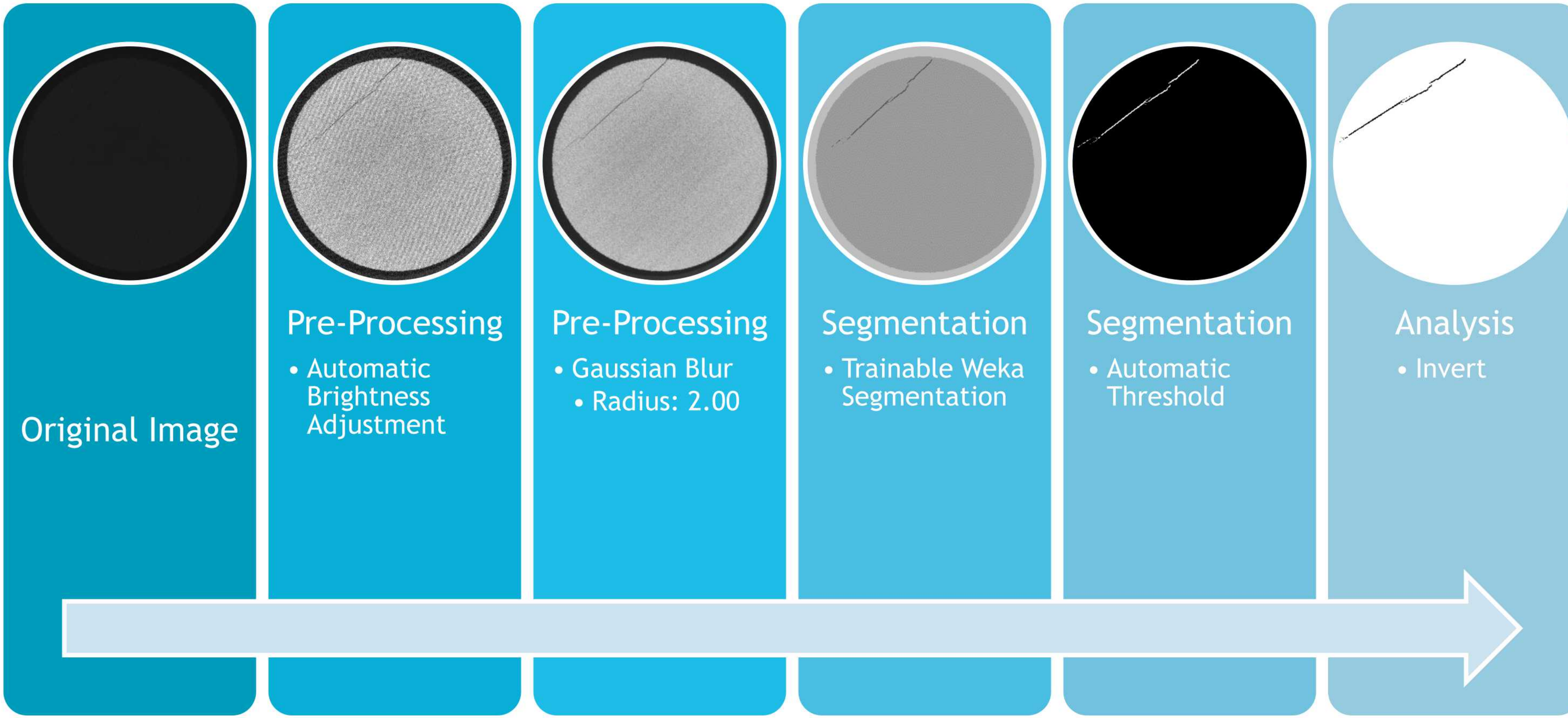
Lorensen, William E., and Harvey E. Cline. "Marching Cubes: A High Resolution 3D Surface Construction Algorithm." *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques - SIGGRAPH 87*, 1987, doi:10.1145/37401.37422.

Method 1



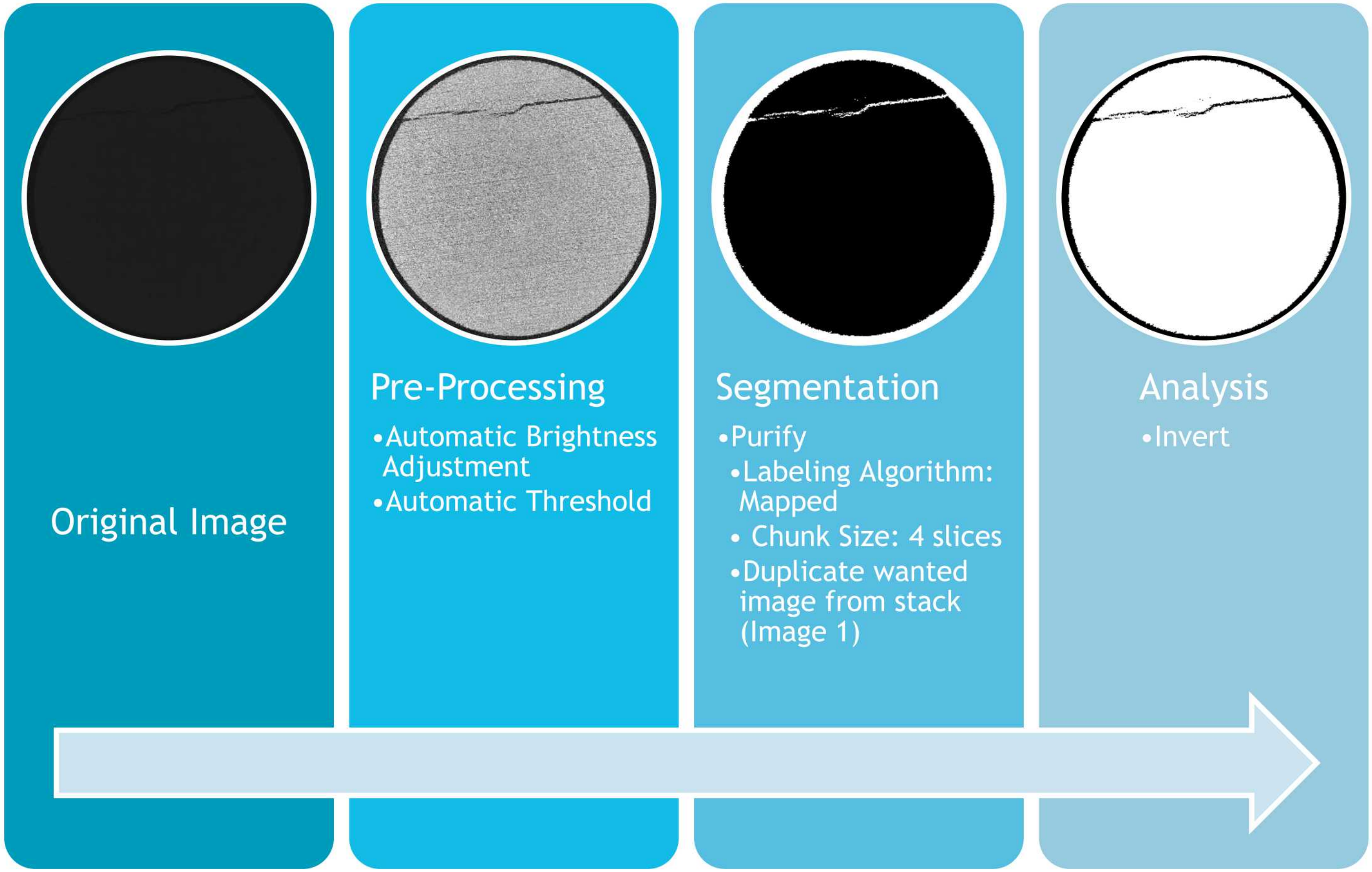
- The pixel values of the fracture and those at the center of the image are identical so pores and noise are exaggerated
- Different methods of segmentation are needed to create cleaner results

Method 2



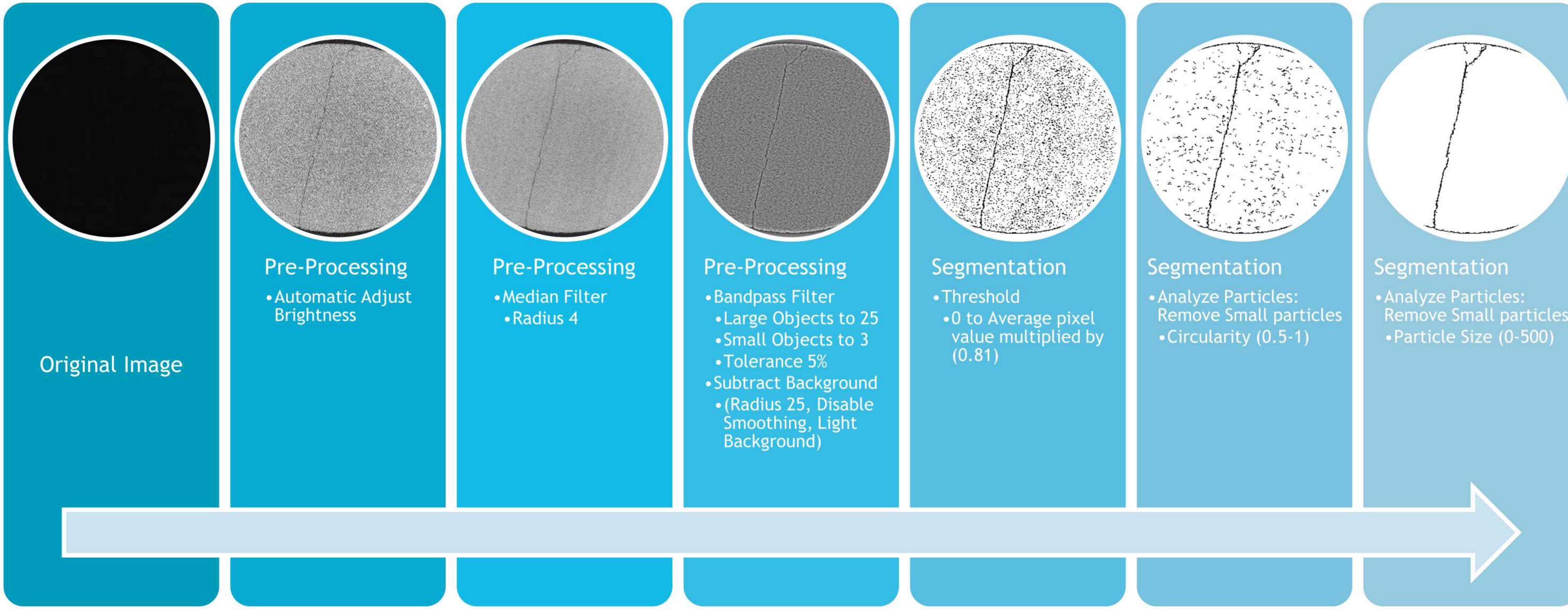
- This method consistently produces more accurate results as well as reducing the noise almost entirely as opposed to first applying a threshold.
- Takes a longer time to process and is not ideal for stacks as it requires more processing power.
- This is a more hands-on approach as it requires the user to train the classifier.

Method 3



- This method is ideal for image stacks as it is a 3D filter and removes particles that are not present in multiple images.
- Has a long processing time but is far less user reliant.

Method 4

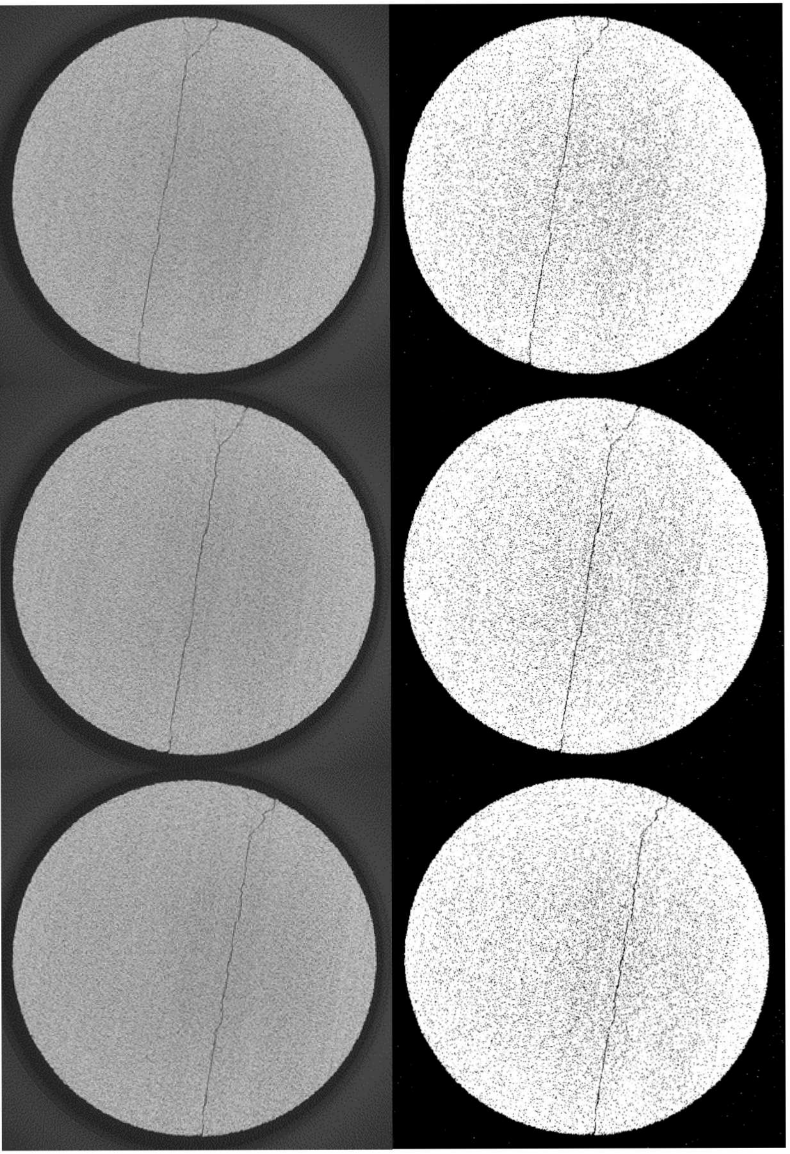


- This method is time consuming and requires many steps
- Large particles remained and branches formed
- Maintained overall connectivity

MicroCT image datasets

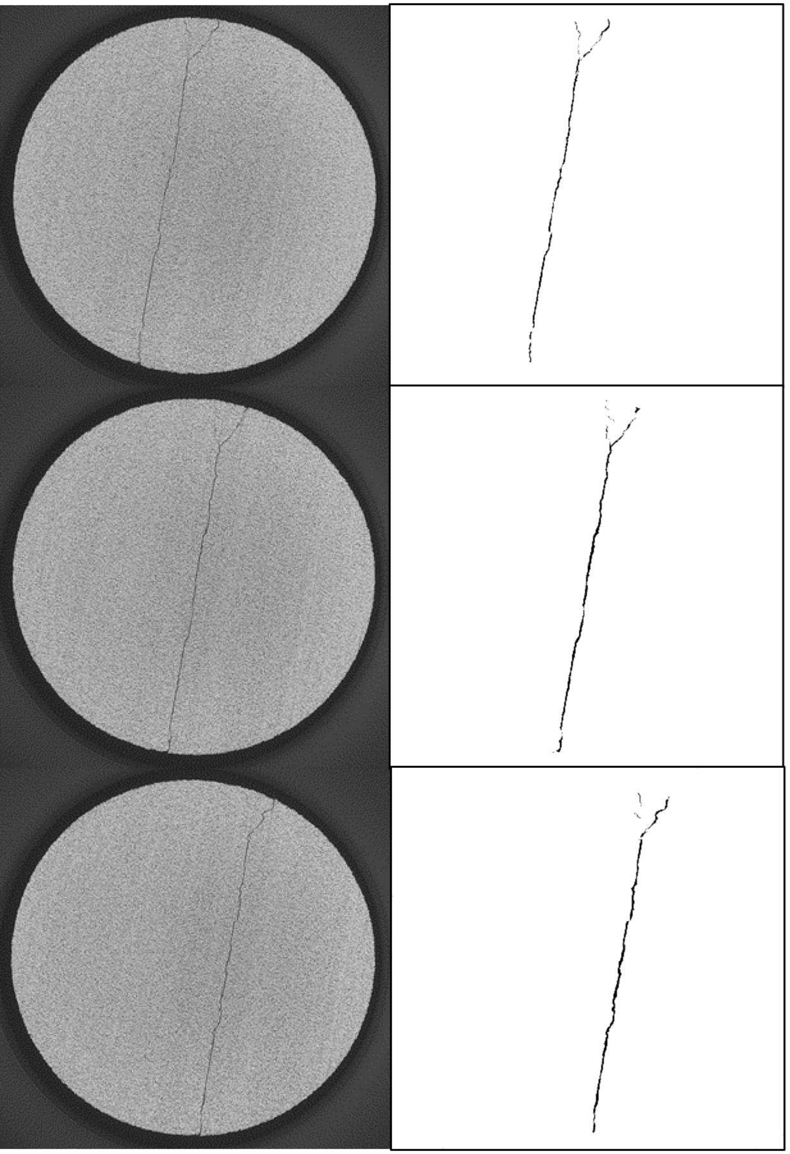
- 22.5 micron resolution per each voxel
- ~3000 image slices per each sample
- Three different datasets are presented (a total of 7 sets were analyzed)
- I45 – printed at 45 degree, H – horizontal, and V - Vertically printed samples
- Image #2501, #2650, and #2800 out of ~3000 images were processed for each stack

3DP-25_145-9 Method 1



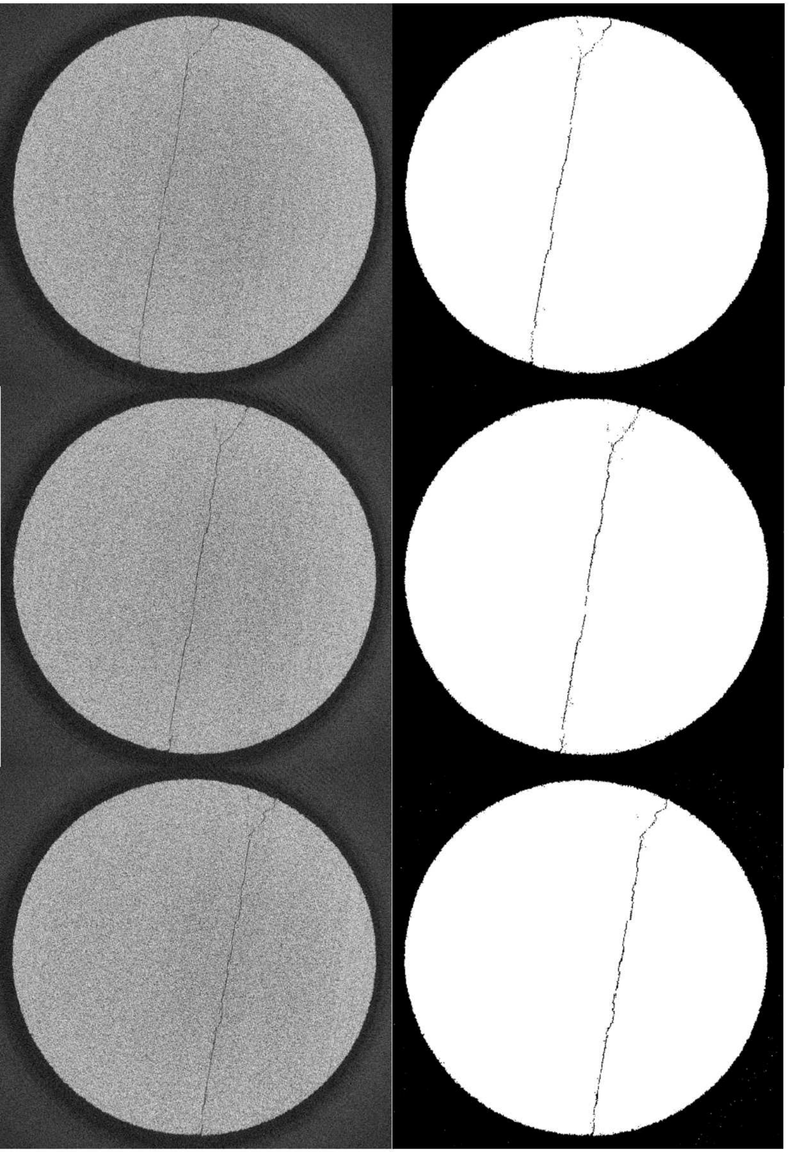
- **Method**
  1. Automatic brightness correction
  2. Automatic threshold
  3. Invert
- **Results**
  - Large amounts of noise and pores remain
  - The noise regions are heavily emphasized around the center region of the image
  - The fracture has maintained connectivity and no regions were lost
- **Discussion**
  - With this image set a large portion of unwanted particles matched the value of shallow fracture regions which created difficulties.
  - Although the fracture is narrow throughout, the width changes are still present

3DP-25_I45-9 Method 2



- **Method**
  1. Automatic brightness correction
  2. Gaussian Blur (Radius: 2)
  3. Trainable Weka Segmentation
     1. Use 3 Classifications (fracture, rock, and void)
     2. Trace regions and classify them accordingly
     3. Repeat as needed
  4. Get Result
  5. Convert to 8-bit
  6. Apply Automatic Threshold
  7. Crop (using polygon selections)
  8. Invert
- **Results**
  - Connectivity was maintained in the wider regions of the fractures
  - The shallow upper areas of the fracture were preserved
  - Width became uniform throughout the wider regions of the fracture
- **Discussion**
  - This method maintained connectivity better than the others although the width discrepancies are more obvious here.
  - All the noise was removed but no regions were lost.

3DP-25_I45-9 Method 3



•Method

1. Automatic brightness correction
2. Automatic threshold
3. Purify
4. Analyze particles (size 0-20, show masks)
5. Image calculator (Purified image, subtract, Mask)
6. Invert
7. Duplicate single images from stack (1, 150, 300)

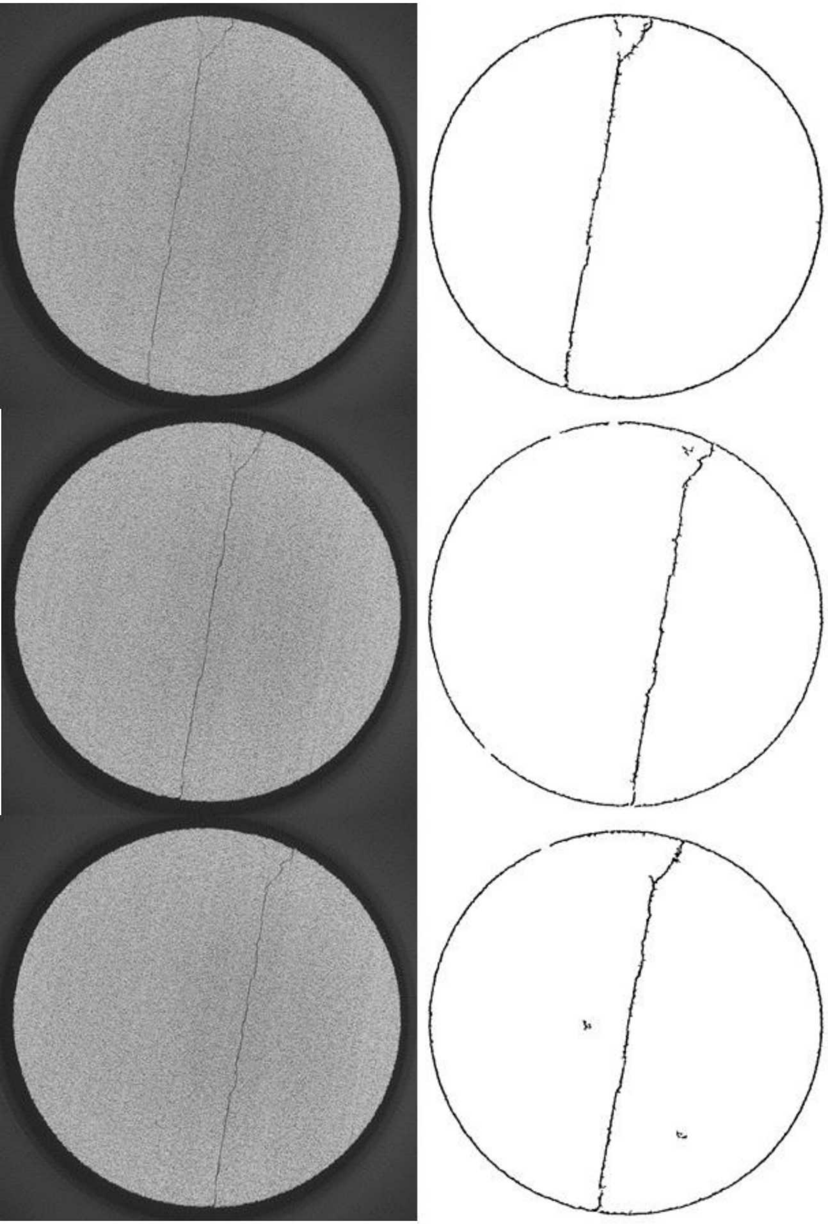
•Results

- Main fracture area preserved entirely and overall connectivity was maintained
- Small amounts of particles remain surrounding the top regions of the fracture

•Discussion

- Top region of the fracture was difficult to segment as the particles share many values of the crack
- In the center of Image 3 a small area of the crack was lost in the analyze particles step as the width was very small after purifying
- Many noise particles remained after purification
- Biggest issue was maintaining connectivity

3DP-25_I45-9 Method 4

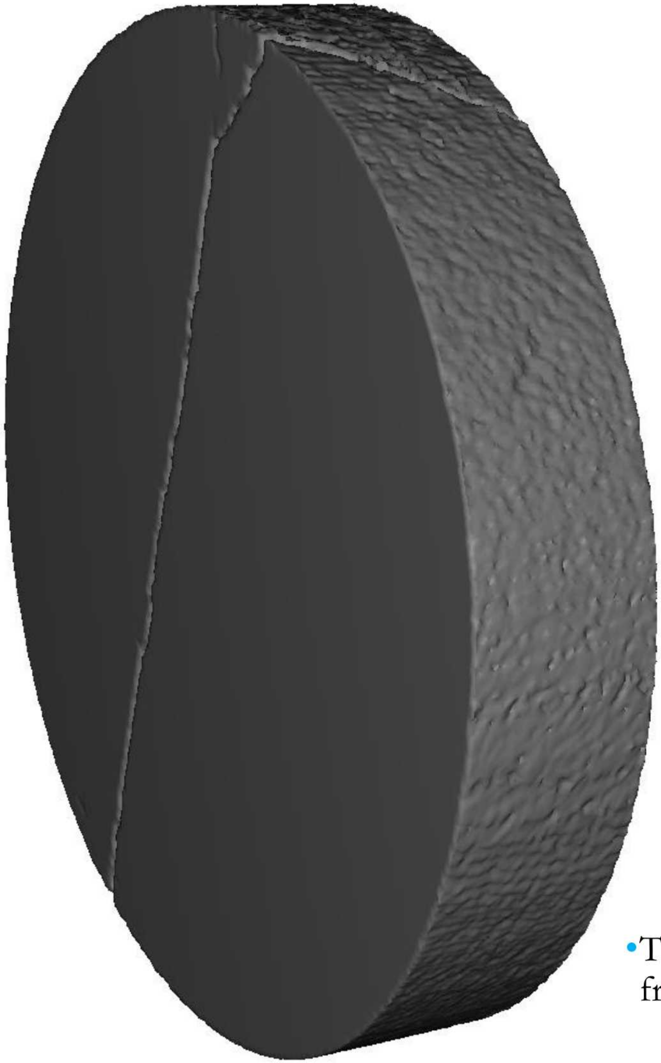


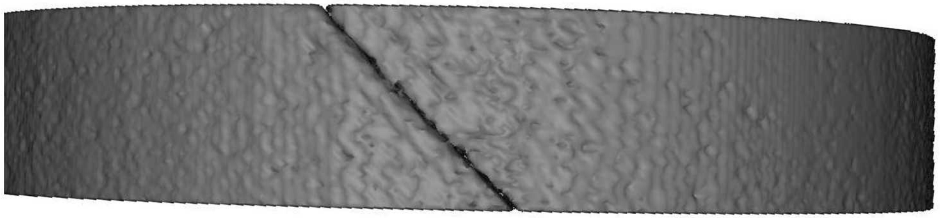
- **Method**
  1. Automatic brightness correction
  2. Median Filter (radius 4)
  3. Bandpass Filter (Small Objects to 3, Large Objects to 25, Tolerance 5%)
  4. Subtract Background (Radius 25, Light Background, Disable Smoothing)
  5. Threshold (0 -average pixel value multiplied by 0.81)
  6. Remove Small Particles (circularity 0.5-1, size 0-500)

- **Results**
  - Connectivity retained
  - Some branching/clumping occurs
  - Fails to capture smaller cracks

- **Discussion**
  - The FFT / Median filter are responsible for the branching, altering their settings could potentially resolve the issue
  - Increasing the threshold would allow for the smaller features to be captured

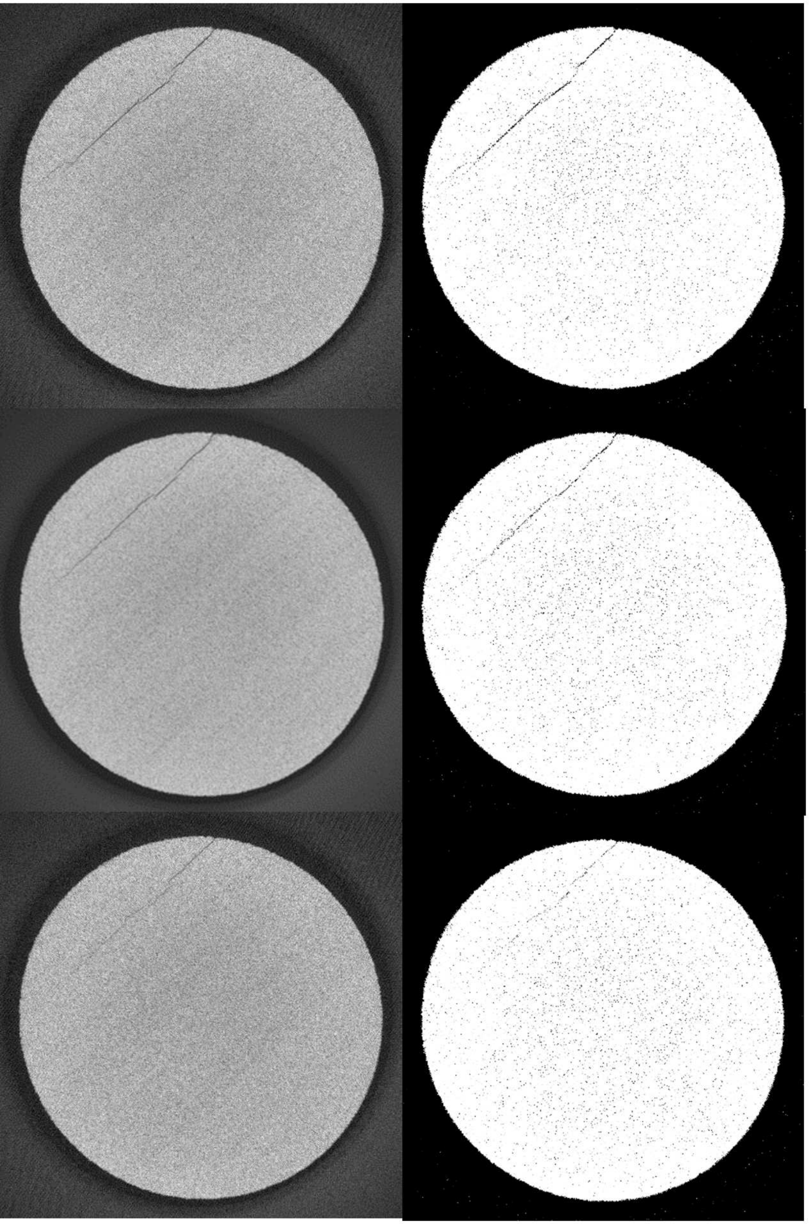
3DP-25_I45-9





- This sample was printed at an incline of 45 degrees which resulted in a fracture going diagonally down the side of the core and a clean break

3DP-28_H-19 Method 1



Method

1. Automatic brightness correction
2. Apply Automatic Threshold
3. Invert

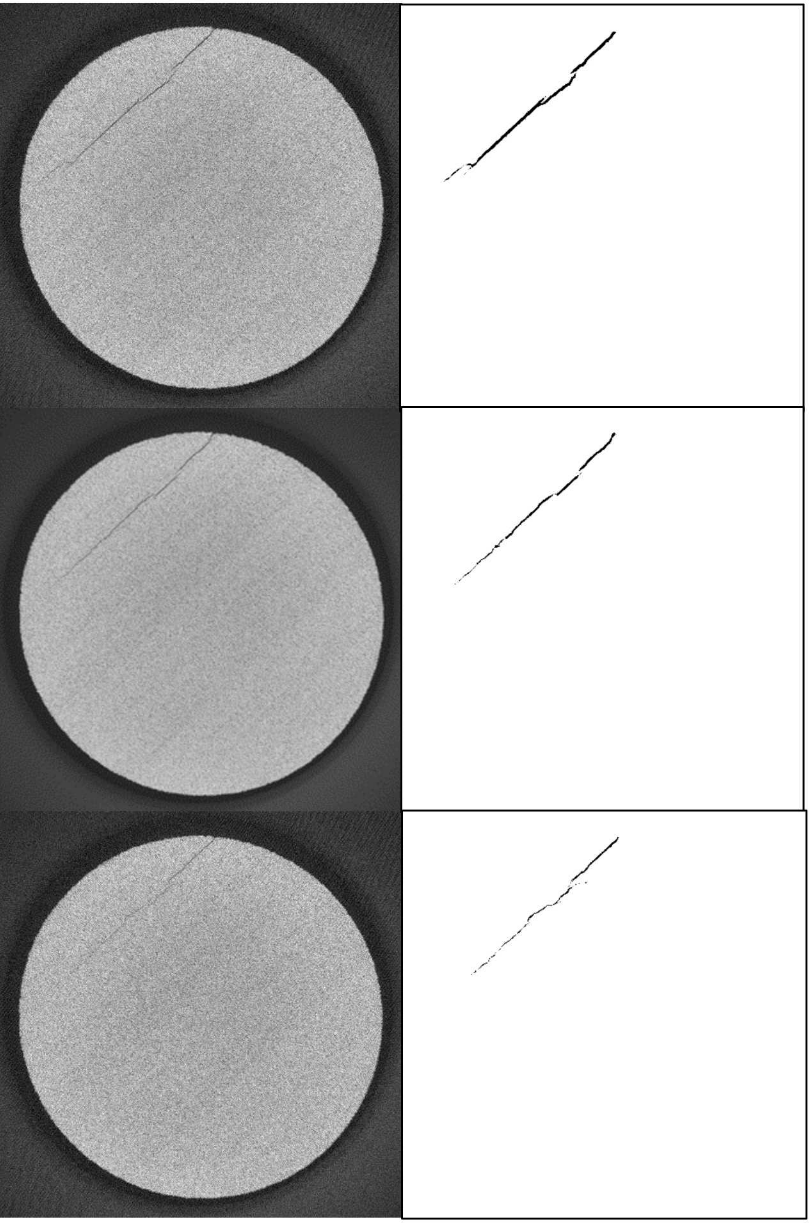
Results

- Many pores and noise particles remain specifically in the center region
- Some very shallow disconnected regions were filtered out by the threshold
- The fracture is not clear

Discussion

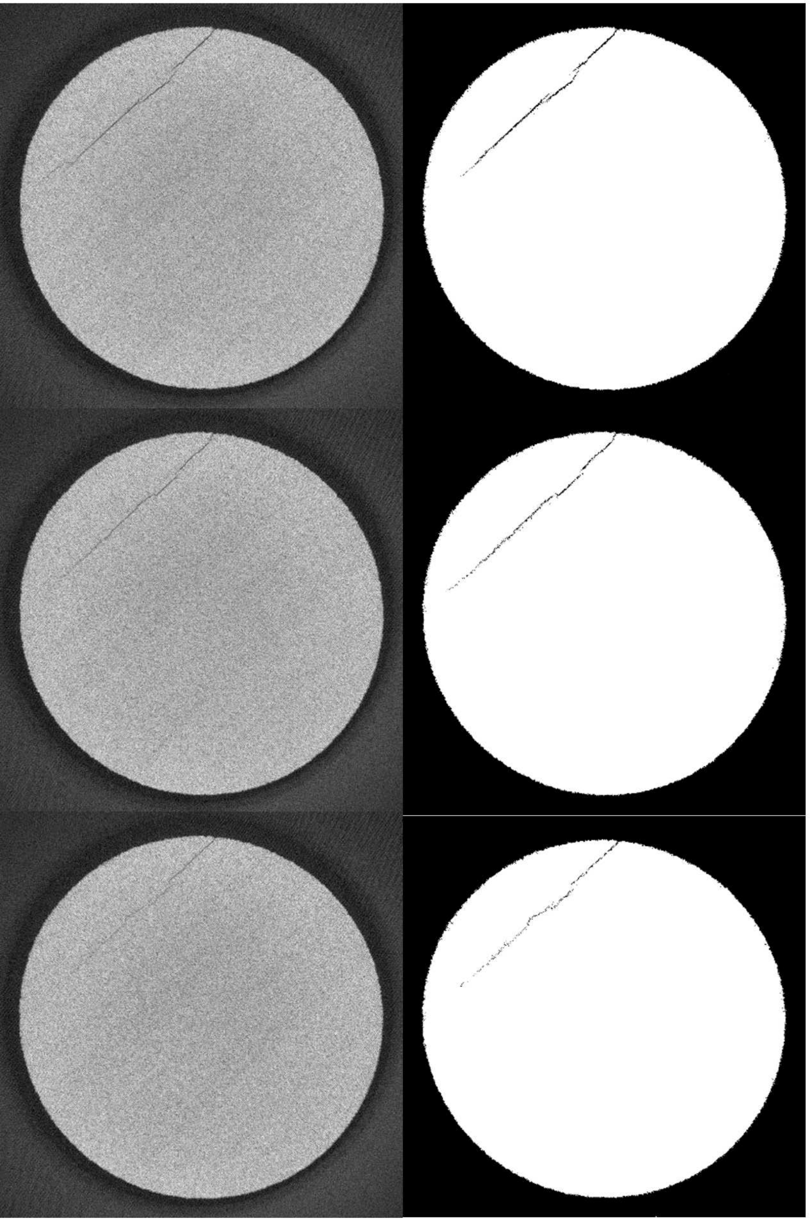
- Many regions of the fracture in this image set have matching values to those of unwanted particles which results in being unable to remove noise.
- Areas of Image 3 appear gray and difficult to see in the final result.

3DP-28_H-19 Method 2



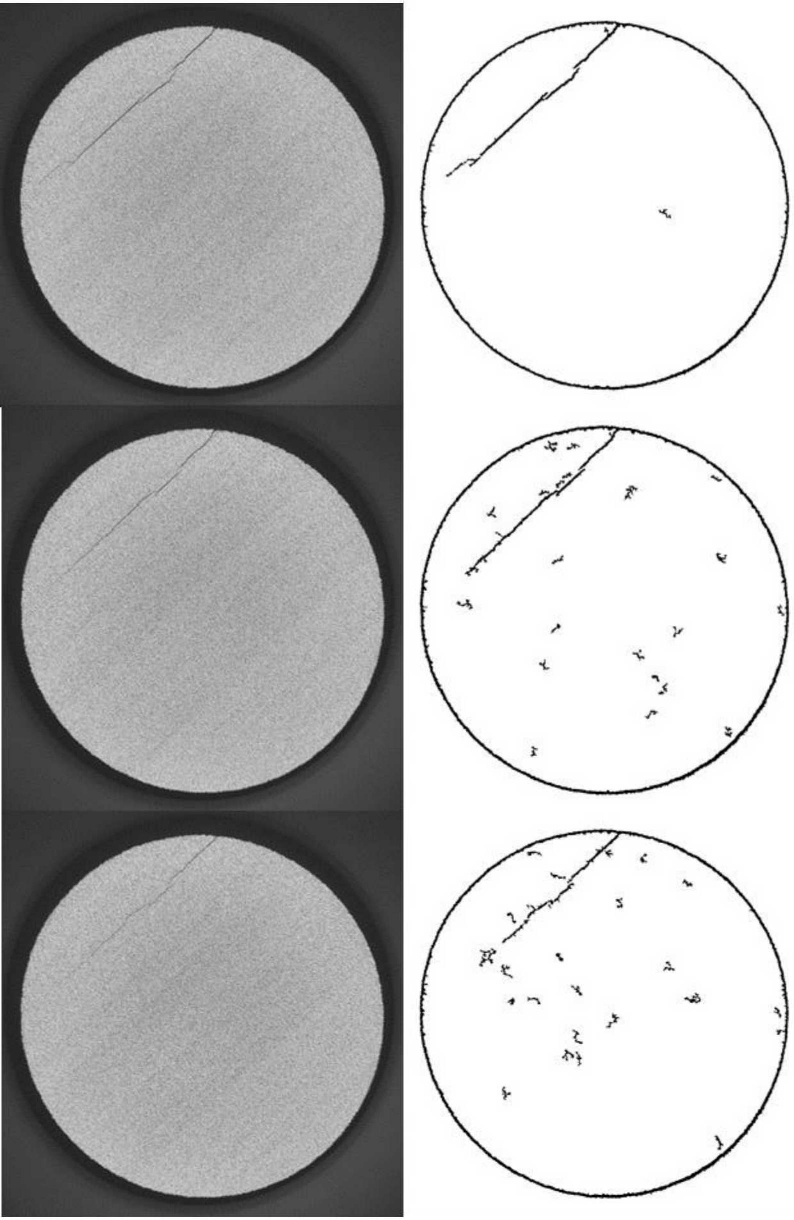
- **Method**
  1. Automatic brightness correction
  2. Gaussian Blur (Radius: 2)
  3. Trainable Weka Segmentation
     1. Use 3 Classifications (fracture, rock, and void)
     2. Trace regions and classify them accordingly
     3. Repeat as needed
  4. Get Result
  5. Convert to 8-bit
  6. Apply Automatic Threshold
  7. Crop (using polygon selections)
  8. Invert
- **Results**
  - Connectivity was lost in shallow regions of the fracture
  - No noise is remaining
  - Fracture is emphasized however most of the width has become uniform
- **Discussion**
  - The non-connected areas of the fracture were highlighted.
  - Shallow regions ended with the same width as the much deeper regions.
  - Some narrow areas were maintained but cannot be easily seen.

3DP-28_H-19 Method 3



- **Method**
  1. Automatic brightness correction
  2. Automatic threshold
  3. Purify
  4. Invert
  5. Duplicate single images from stack (1, 150, 300)
- **Results**
  - Changes in width were maintained throughout the stack
  - Noise was entirely eliminated in the purify step
  - Top left region of the fracture was filtered out
- **Discussion**
  - Because of the light and distance particles making up the top left fracture in Image 1, a region was lost with the purify filter
  - The connectivity was maintained throughout the stack well as a result of deep fracture

3DP-28_H-19 Method 4

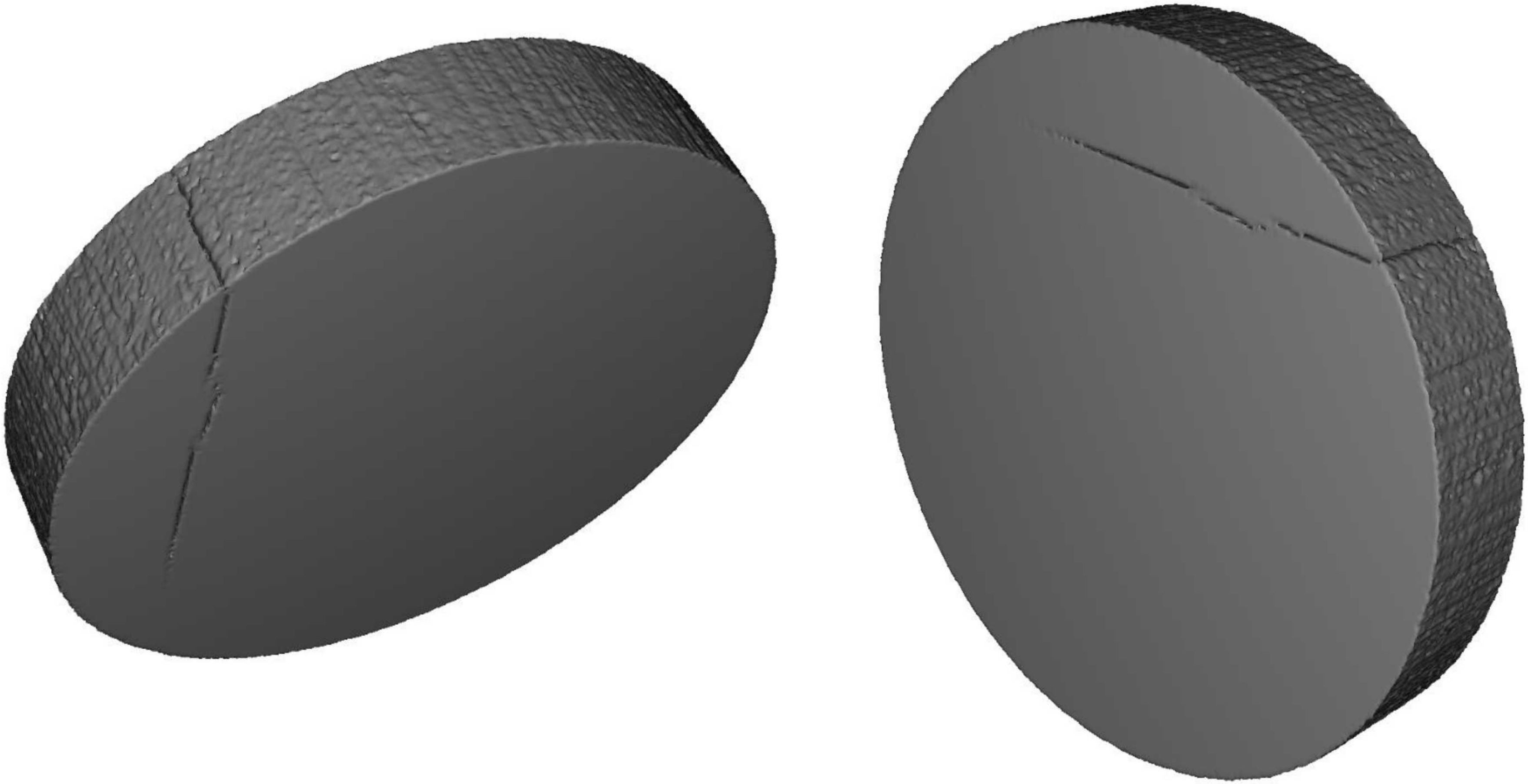


- **Method**
  1. Automatic brightness correction
  2. Median Filter (radius 4)
  3. Bandpass Filter (Small Objects to 3, Large Objects to 25, Tolerance 5%)
  4. Subtract Background (Radius 25, Light Background, Disable Smoothing)
  5. Threshold (0 -average pixel value multiplied by 0.81)
  6. Remove Small Particles (circularity 0.5-1, size 0-500)

- **Results**
  - The resulting images still have a considerable amount of leftover particles
  - Images near the end of the stack lose features.
  - Branching becomes more exaggerated near the end of the stack.

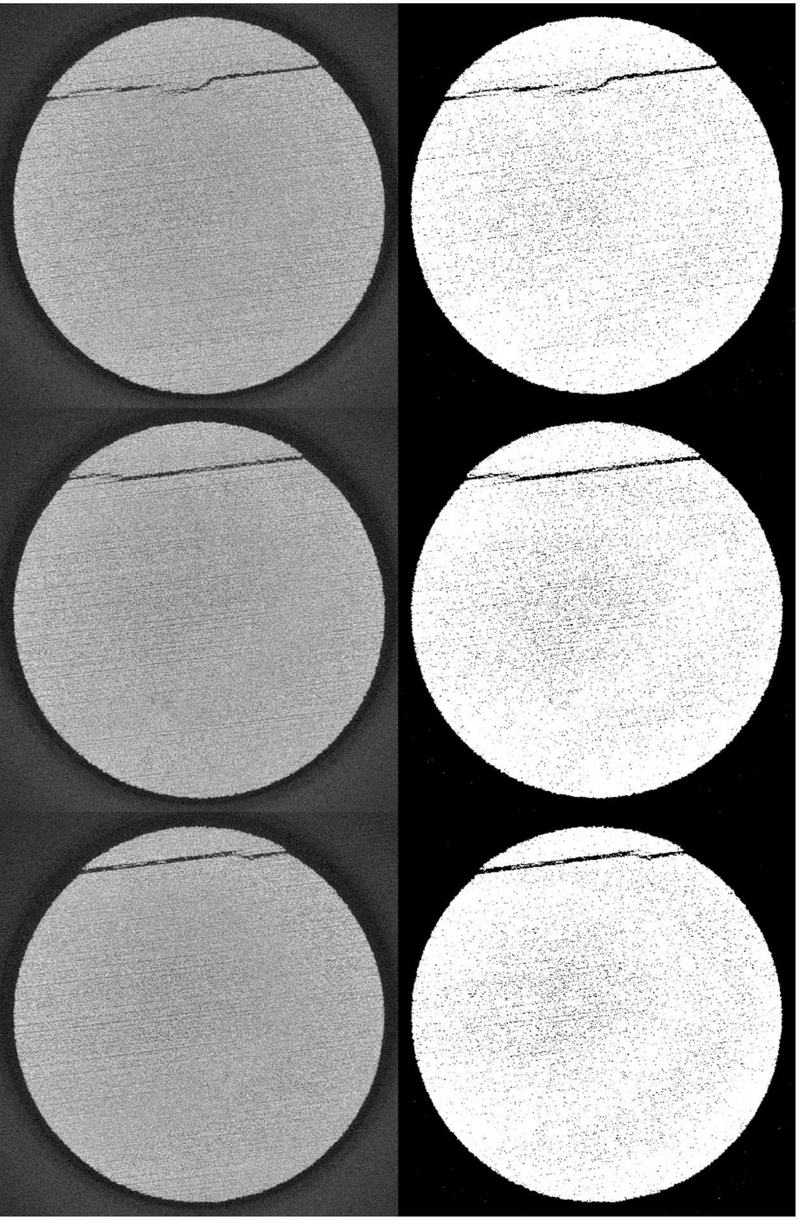
- **Discussion**
  - The loss of detail was a result of the crack becoming lighter and thinner
  - Results could potentially be improved by adding more post-thresholding steps.
  - Lowering the threshold may be necessary to prevent branching.

3DP-28_H-19 Isosurface



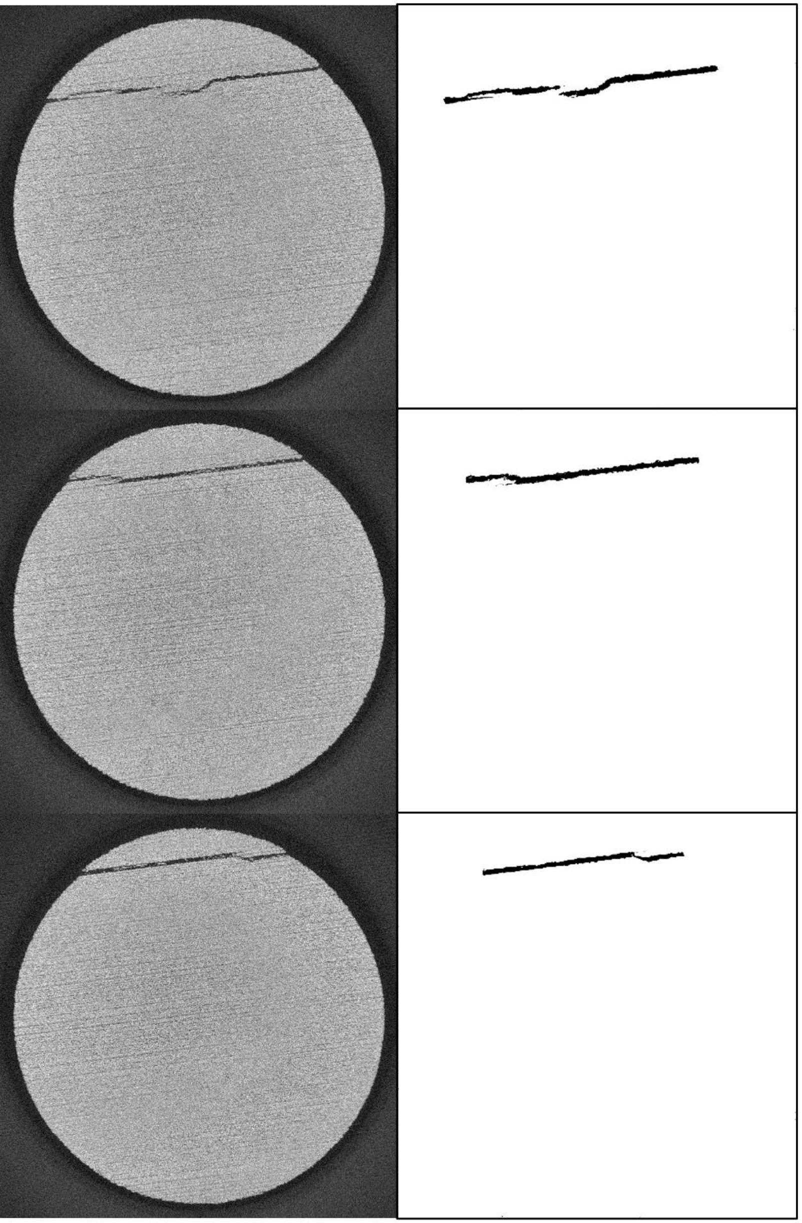
- This sample was printed horizontally which resulted in the fracture going straight down the side of the core and a cleaner break

3DP-31_V23 Method 1



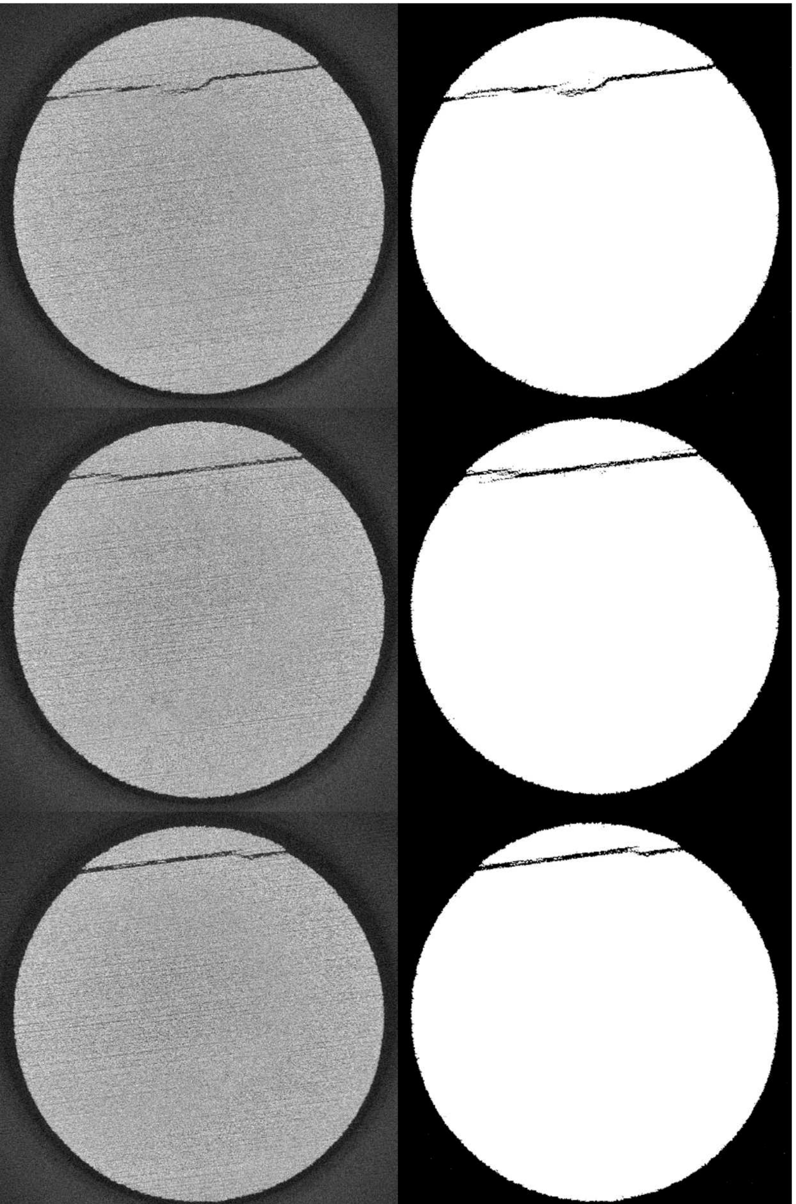
- **Method**
  1. Automatic brightness correction
  2. Automatic threshold
  3. Invert
- **Results**
  - Many pores and noise particles remain and now have the same value as the fracture
  - Horizontal lines have become emphasized in certain regions
  - Fracture has remained the same and no parts were lost
- **Discussion**
  - The horizontal lines have become more defined which takes away from the analysis of the fracture.
  - The fracture in its entirety has been preserved at the cost of many unwanted particles.
  - Method 1 works best with this image set as the fracture is originally well defined.

3DP-31_V23 Method 2



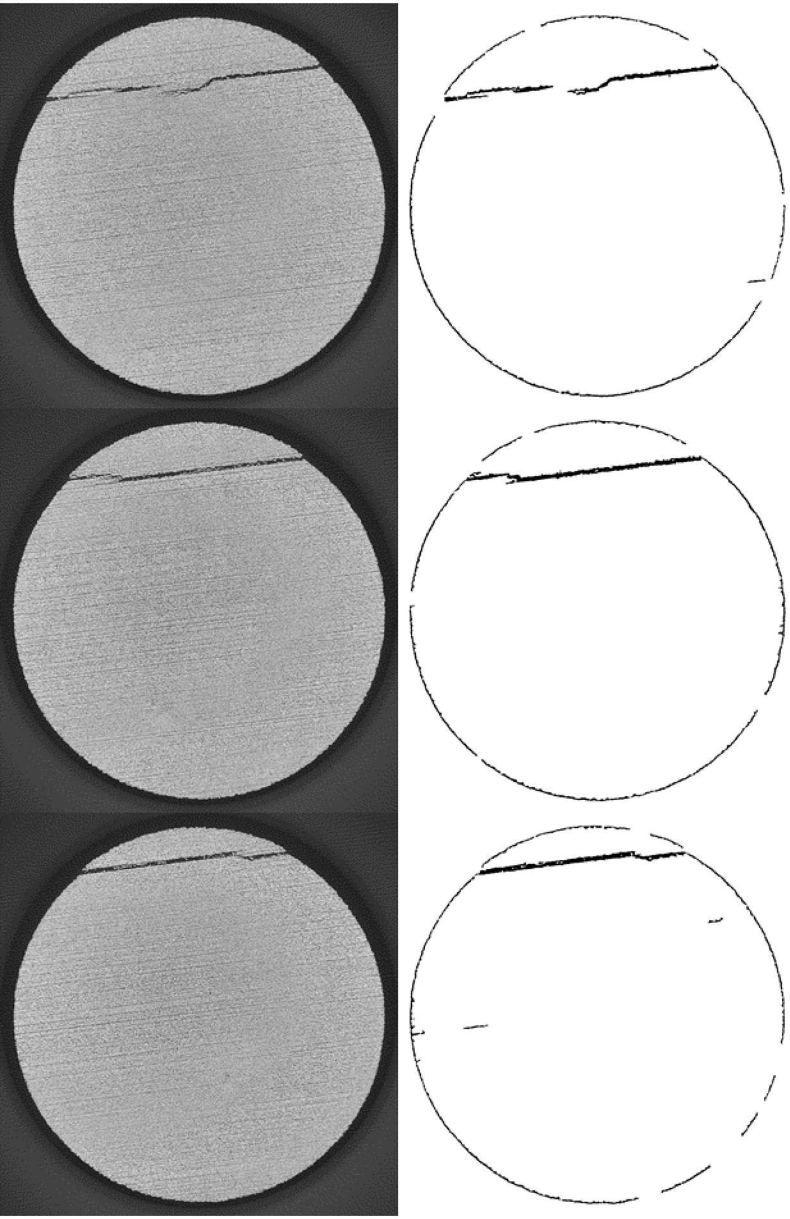
- **Method**
  1. Automatic brightness adjustment
  2. Gaussian Blur (radius 2)
  3. Trainable Weka Segmentation (repeat as many times as needed)
     1. Use 3 Classifications (fracture, rock, and void)
     2. Trace regions and classify them accordingly
     3. Repeat as needed
  4. Get Result
  5. Convert to 8-bit
  6. Apply Automatic Threshold
  7. Crop (using polygon selections)
  8. Invert
- **Results**
  - Maintained complete connectivity where it exists originally
  - Small regions of the fracture were lost
  - Swelling occurred at the end points of the fracture
  - **Discussion**
    - A region of the fracture was comprised of small particles grouped together and was subsequently lost
    - Areas within the fracture that had some rock were all classified as part of the fracture
    - Width became uniform throughout

3DP-31_V23 Method 3



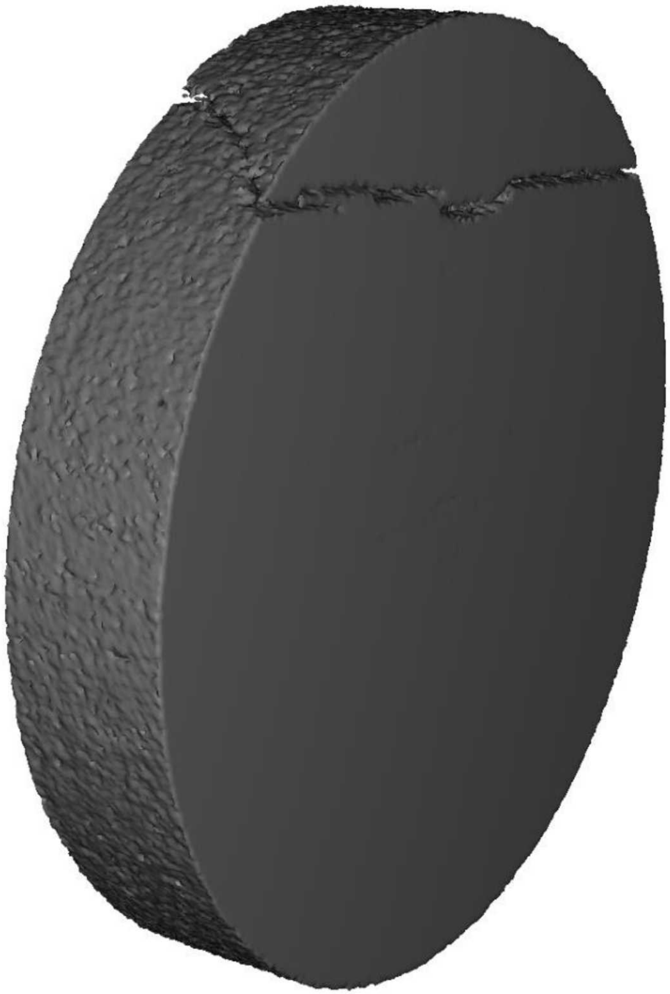
- **Method**
  1. Automatic brightness correction
  2. Automatic threshold
  3. Purify
  4. Invert
  5. Duplicate single images from stack (1, 150, 300)
- **Results**
  - Connectivity was maintained entirely
  - Changes in width and depth in the fracture remain visible
  - Areas of rock within the crack are still existent
- **Discussion**
  - The fracture was clearly defined and deep which allowed for simple segmentation
  - Lighter areas were entirely preserved although there are areas of light noise

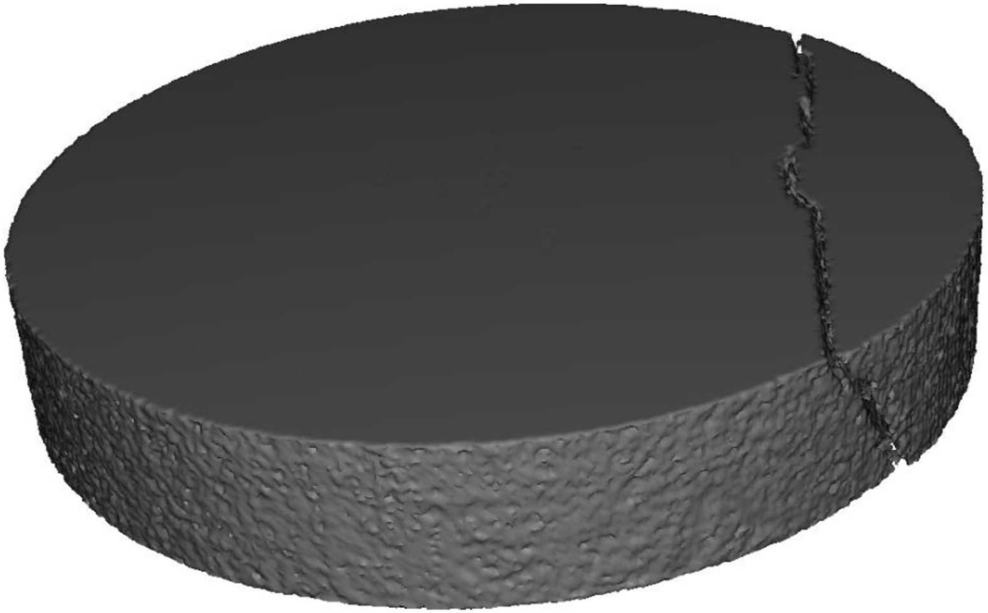
3DP-31_V23 Method 4



- **Method**
  1. Automatic brightness correction
  2. Median Filter (radius 4)
  3. Bandpass Filter (Small Objects to 3, Large Objects to 25, Tolerance 5%)
  4. Subtract Background (Radius 25, Light Background, Disable Smoothing)
  5. Threshold (0 -average pixel value multiplied by 0.81)
  6. Remove Small Particles (circularity 0.5-1, size 0-500)
- **Results**
  - Images had good connectivity and overall accurately reflected the original set
  - Cracks were slightly widened
  - Some particles remain
- **Discussion**
  - Widening of the cracks was a result of the filter and threshold used. Could potentially be resolved by an erosion algorithm.
  - Most leftover particles can be removed manually through a ROI selection.

3DP-31_V23





- This sample was printed vertically which resulted in a fracture with ‘steps’ along the side of the core and a more jagged break

Discussion

- Found that the Trainable Weka Segmentation, the FFT Bandpass Filter, and BoneJ's purify consistently produced the most accurate segmentations
- BoneJ's purify has better success on image stacks (as opposed to single images)
- Overall, these processes were able to successfully reduce noise and isolate fracture features
- However, segmentations sometimes corrupted the geometry of the original fracture (lost endpoints, creation of cavities, unwanted changes in width)

Conclusions

- Further applications include :
  - Using segmented images to create 3D models of the fracture system
  - Research regarding the effects of stress on geological samples
  - Research regarding the structure of geological samples

- Major takeaways:
  - Image processing is a vital part of the research process
  - Knowledge of various computational algorithms and their applications
  - Improved understanding of machine learning
  - Insight into the process of 3D printing

Thank You!!!

Printing with Gypsum Powder

- A thin layer of calcium sulfate hemihydrate ($CaSO_4$-0.5H_2O) power is deposited onto build chamber
- A print head with binder jets dispenses a binder material where binding is required
- Chemical reaction of powders with water (and additives) will harden the printed part over time
- The build chamber is lowered and then repeated
- The dimension of the build bed is ~20-30 cm and the resolution of inkjet print heads is 300-500 dots per inch (~100 micron thick layer)
- Cracks or flaws can be printed at ~1mm feature resolution



Methodology Overview

General Methodology:

- 1. Pre-Processing
- 2. Isolating the region of interest
- 3. Conducting analysis

Limitations:

- Most image processing algorithms rely on pixel values in order to produce a result. This causes problems when a region of interest (i.e., a fracture) shares similar values with the surrounding image.
- Post-processing analysis assumes the segmented image is an accurate representation of the original.
- Results are greatly affected by the quality and resolution of the original image set.

Benefits:

- Image processing reduces bias
- Results can be replicated exactly
- Clarifies and enhances the region of interest (ROI)
- Segmentation allows for automated and precise measurements of the ROI