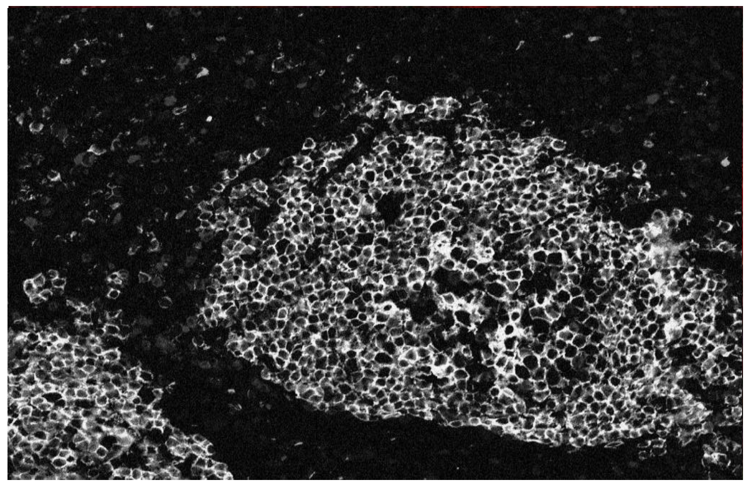
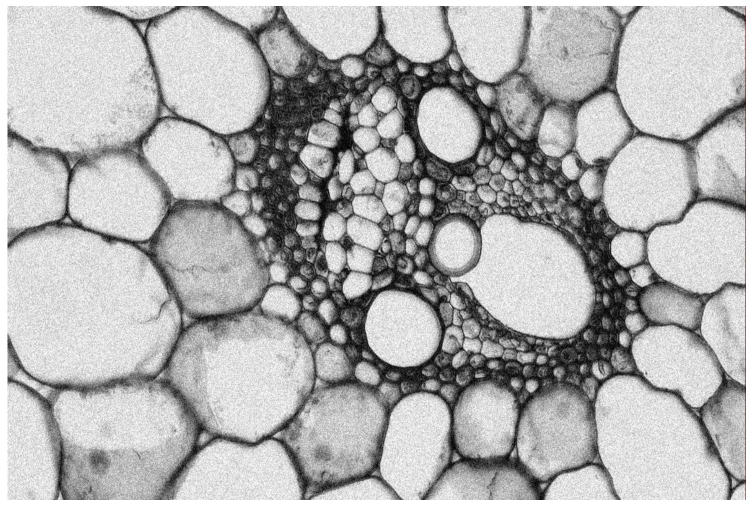
FEATURE EXTRACTION USING IMAGEJ

IMAGE AND VIDEO PROCESSING – MODULE 1





This Photo by Unknown Author is licensed under CC BY-NC-ND

NINO PHILIP RAMONES | GITHUB

2020 - 05616

APRIL 21, 2023

OBJECTIVES

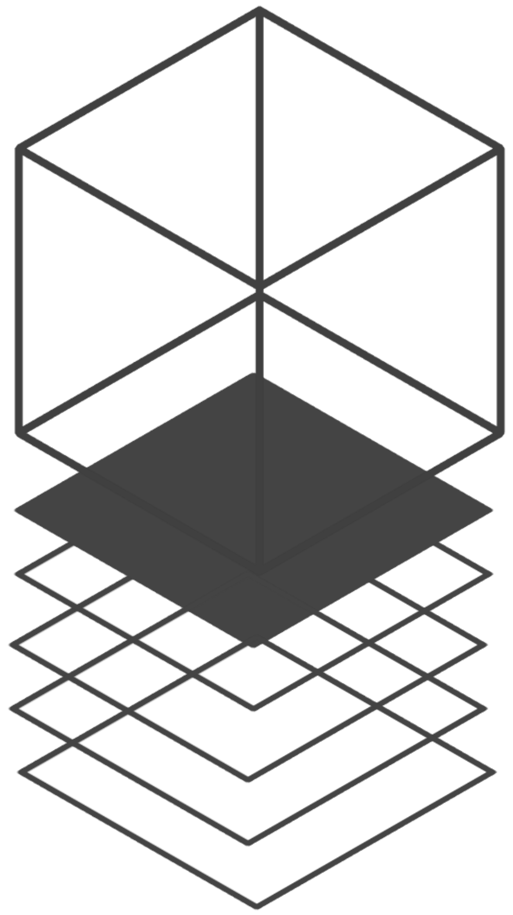
- Segment images using ImageJ and derive some noticeable features from the processed image
- Compare how ImageJ segments and reports values from actual scales and measurements of some real life objects

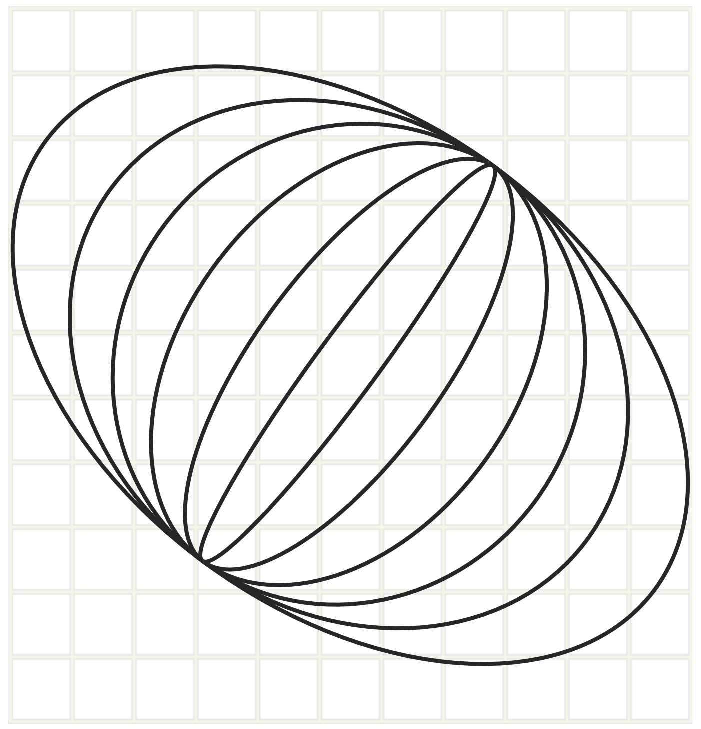
KEY TAKEAWAYS

- Thresholding is useful in image segmentation since it is the simplest way to separate blobs or cells of interest from the background
- Setting a scale to your image is crucial in obtaining some measurements

SOME PITFALLS

- Image must not be inverted after binarizing since ImageJ reads an error on the threshold values – this affects the ability of the application to clip grayscale values and eventually segmenting the image as it should





IMAGEJ

Image Processing and Analysis in Java

ImageJ is a **public domain image processing** program developed at the National Institutes of Health for computational instrumentation. It is a Java-based scientific image processing software that can measure properties of objects using **image segmentation techniques**, which will be the focus of the next activities.

In this activity, ImageJ was used for image segmenting and analyze regions of interests as such. User-written plugins inherent to ImageJ has made it possible for the public to solve various image processing problems not limited to three-dimensional cell imaging and even **blob analysis** (ImageJ Documentation, n.d.)

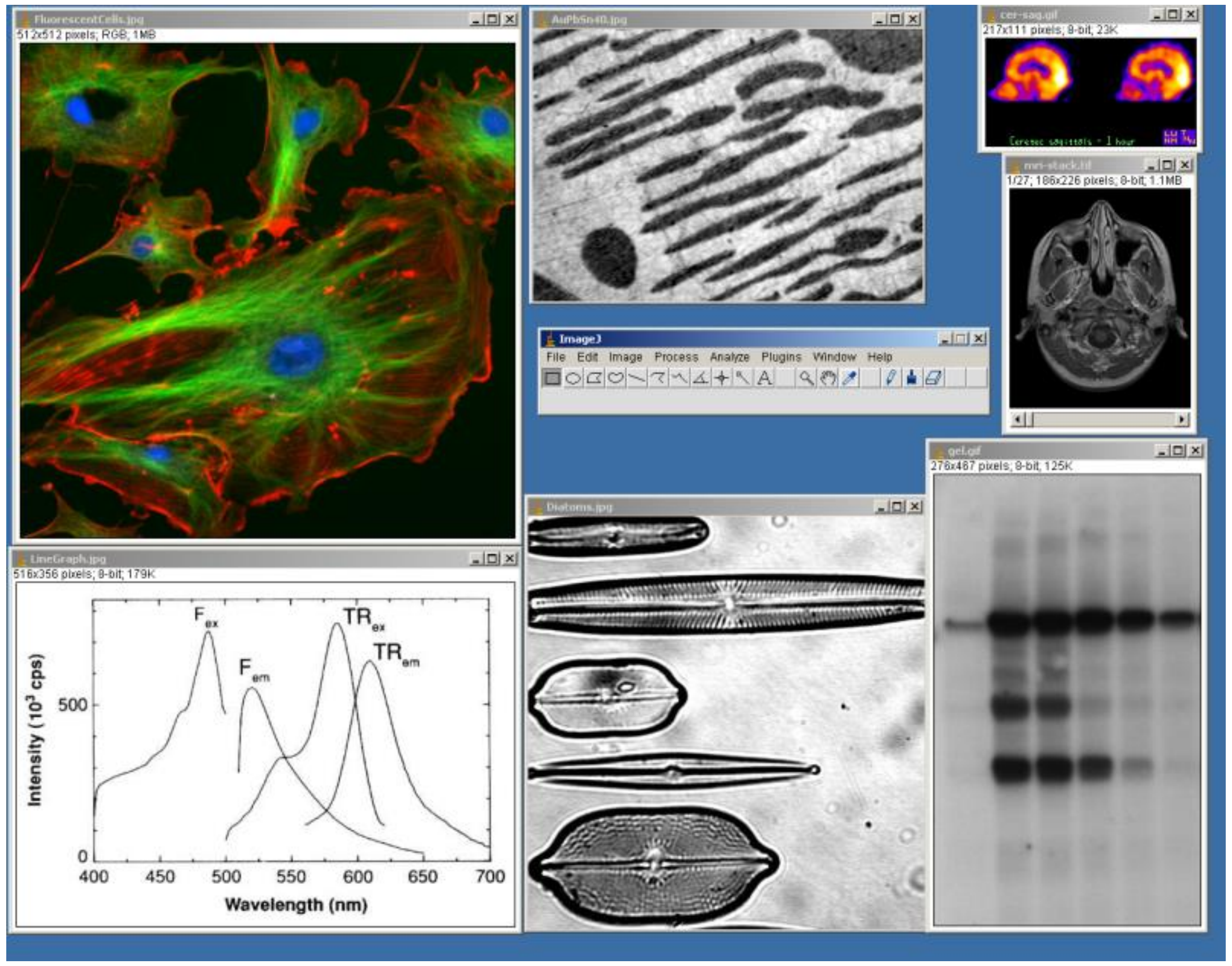


Figure 1. Screenshot of the ImageJ from Wikipedia.

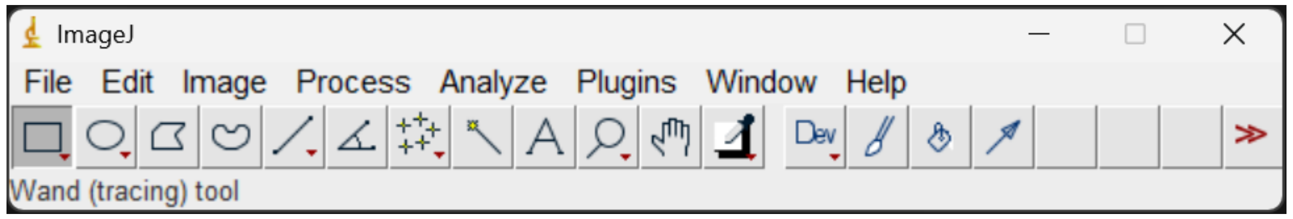


Figure 2. ImageJ interface as opened in Windows.

SAMPLE FEATURE EXTRACTION

Basic blob detection and analysis of grains of sands

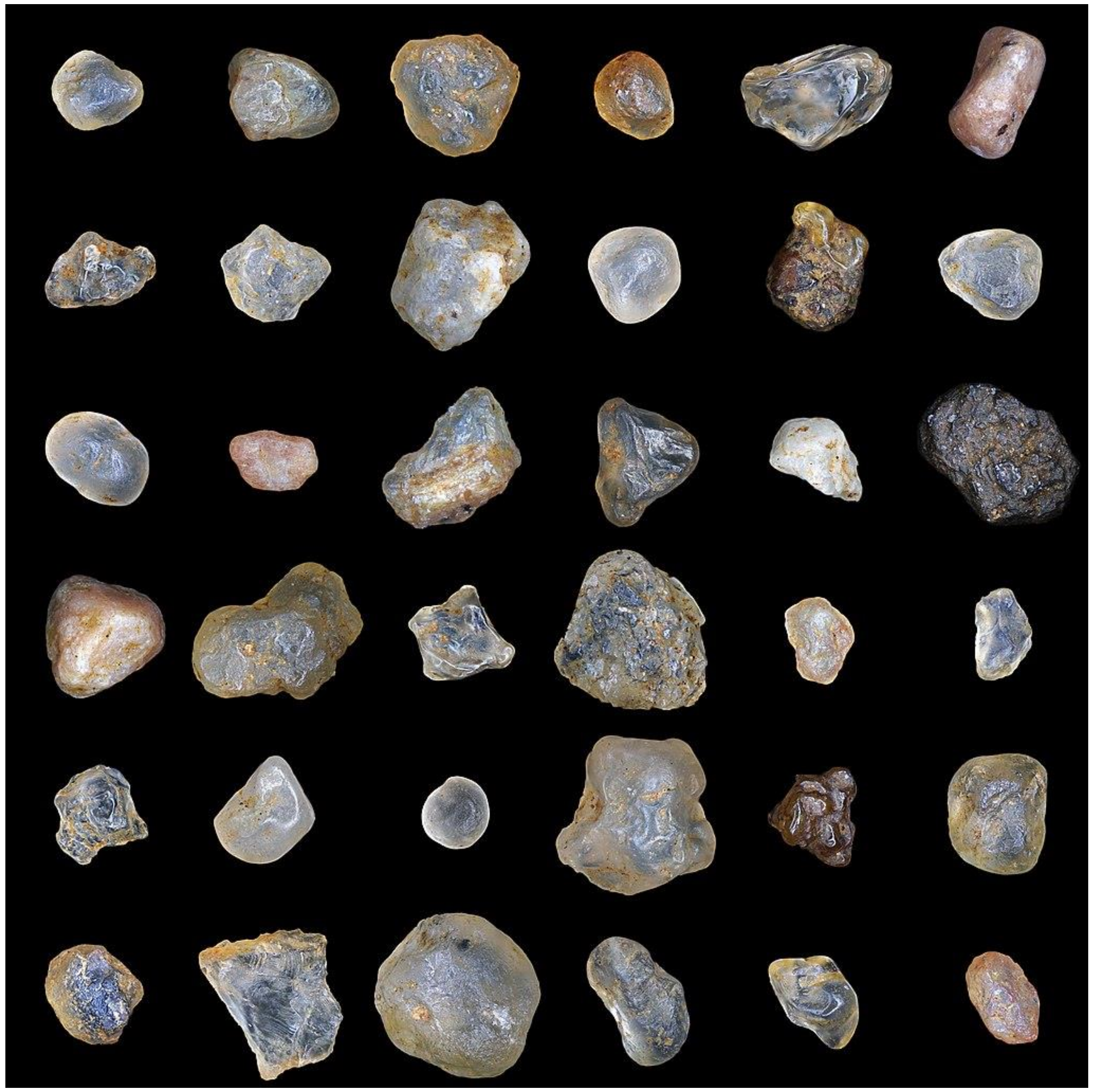
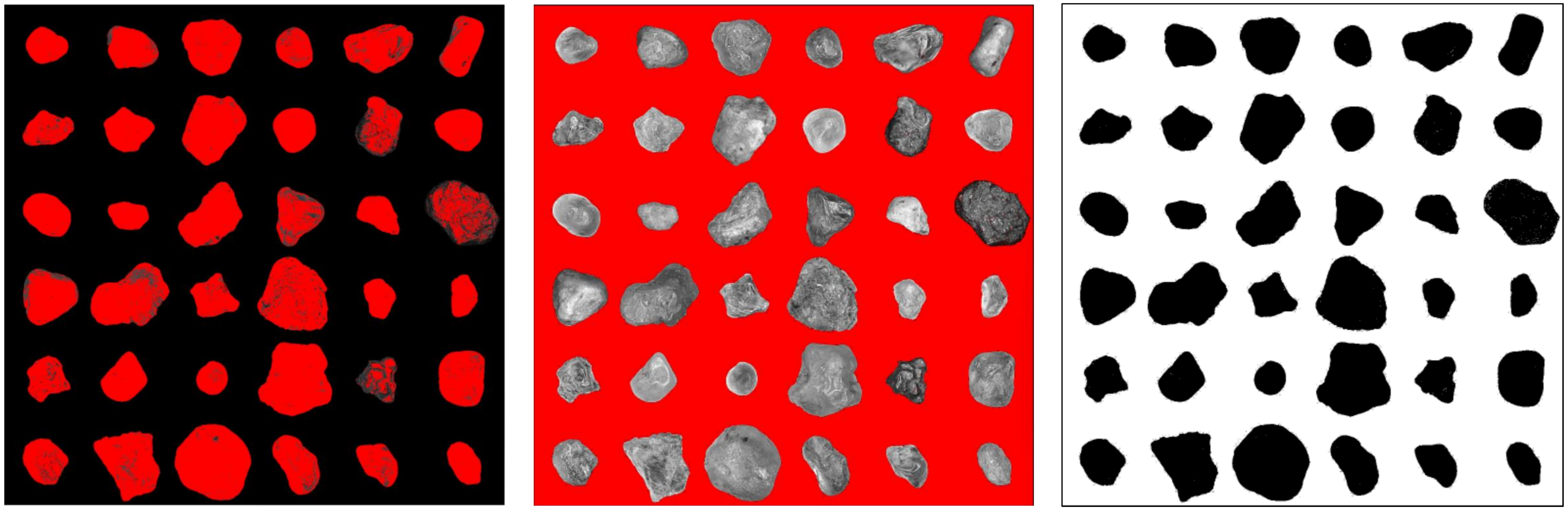


Figure 3. Sand grains of varying sizes. Image sourced from Wikipedia.

In this activity, we will consider **segmenting** the image of sand grains of varying shapes and sizes and draw some properties of individual grains using ImageJ.

It must be noted at first that to make physical measurements from digital images, a scale must be set that that can be used as reference. In this image, the sand grain with **assumed known length or diameter** was used as reference for values. Since the exact definition of sand varies, a range of sand particle diameter was taken – around 0.0625 mm as per the particle size set by geologists.

SEGMENTATION BY THRESHOLDING



(a) Prior to threshold adjustment (b) Adjusted threshold near peak values (c) Threshold applied to sand image

Figure 4. Image segmentation by thresholding using ImageJ.

Prior to thresholding, the sand grains image was converted from **RGB to 8-bit grayscale image**. Converting the image to 8-bit is crucial so that we can segment our grains through thresholding. This can be seen in figures 4a and 4b where the grains are automatically segmented or separated from the background upon adjusting our threshold values. Basically, the grayscale makes the segmentation techniques much easier since a specific **gray value will act as a cut-off**, hence, threshold for our image!

BINARIZING THE IMAGE

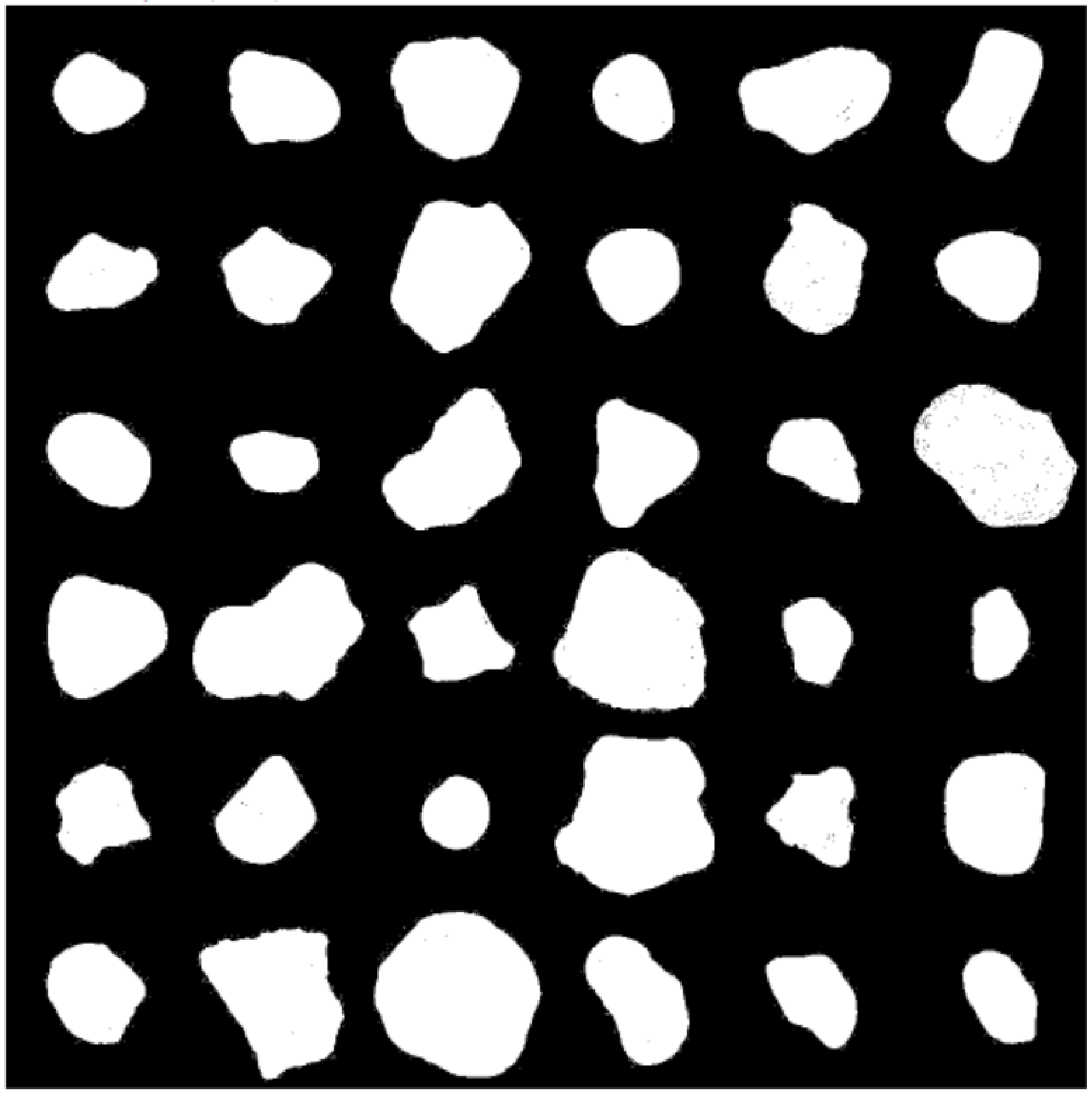
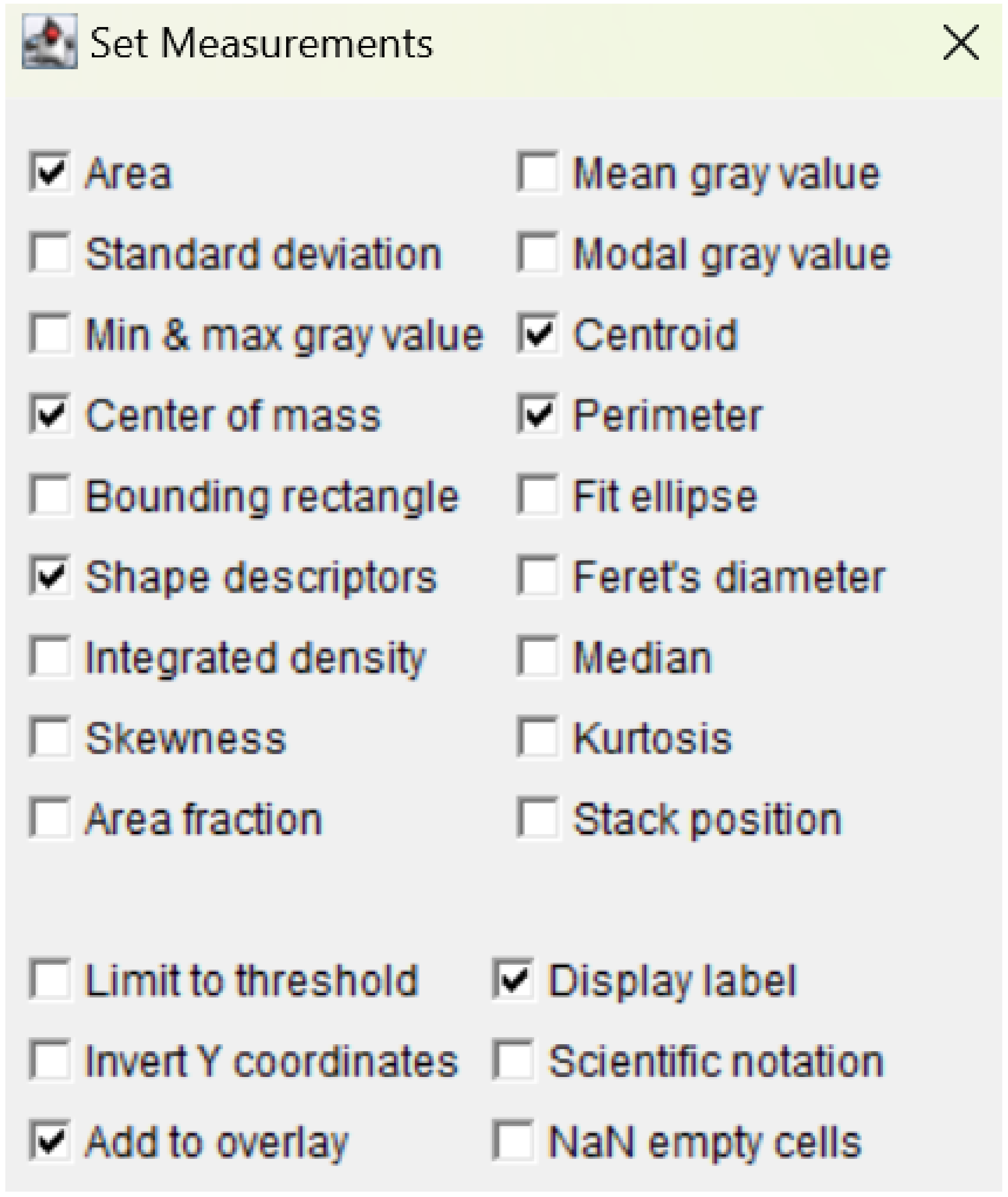


Figure 5. Binarized image of sand grains after thresholding.

After thresholding, we binarize the image. In principle, **binarizing** is the process of taking a grayscale image and converting it to **black and white** as it is. This reduces the information contained in the image since the 256 shades or values of gray has been reduced to two – black and white! This prepares ImageJ to measure some properties of the grains since the blobs or grains are now explicitly distinct from the background.

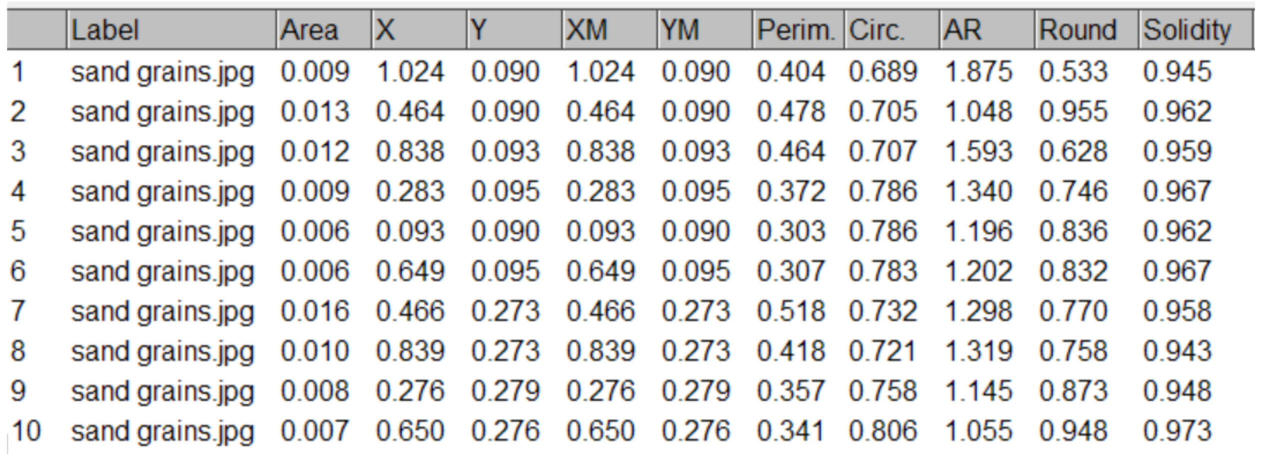
SAND GRAIN MEASUREMENTS



Set Measurements

☑ Area	☐ Mean gray value
☐ Standard deviation	☐ Modal gray value
☐ Min & max gray value	☑ Centroid
☑ Center of mass	☑ Perimeter
☐ Bounding rectangle	☐ Fit ellipse
☑ Shape descriptors	☐ Feret's diameter
☐ Integrated density	☐ Median
☐ Skewness	☐ Kurtosis
☐ Area fraction	☐ Stack position
☐ Limit to threshold	☑ Display label
☐ Invert Y coordinates	☐ Scientific notation
☑ Add to overlay	☐ NaN empty cells

Figure 6. Parameters to be measured from the grains.



	Label	Area	X	Y	XM	YM	Perim.	Circ.	AR	Round	Solidity
1	sand grains.jpg	0.009	1.024	0.090	1.024	0.090	0.404	0.689	1.875	0.533	0.945
2	sand grains.jpg	0.013	0.464	0.090	0.464	0.090	0.478	0.705	1.048	0.955	0.962
3	sand grains.jpg	0.012	0.838	0.093	0.838	0.093	0.464	0.707	1.593	0.628	0.959
4	sand grains.jpg	0.009	0.283	0.095	0.283	0.095	0.372	0.786	1.340	0.746	0.967
5	sand grains.jpg	0.006	0.093	0.090	0.093	0.090	0.303	0.786	1.196	0.836	0.962
6	sand grains.jpg	0.006	0.649	0.095	0.649	0.095	0.307	0.783	1.202	0.832	0.967
7	sand grains.jpg	0.016	0.466	0.273	0.466	0.273	0.518	0.732	1.298	0.770	0.958
8	sand grains.jpg	0.010	0.839	0.273	0.839	0.273	0.418	0.721	1.319	0.758	0.943
9	sand grains.jpg	0.008	0.276	0.279	0.276	0.279	0.357	0.758	1.145	0.873	0.948
10	sand grains.jpg	0.007	0.650	0.276	0.650	0.276	0.341	0.806	1.055	0.948	0.973

Figure 7. Table of values for measurements generated.

From the set measurement options in ImageJ, the **parameters of interest** were chosen that will be extracted from the binarized image of our sand grains. This is why we have set a scale earlier since most of the measurements that will be returned were based on the set scaled diameter of the grain – 0.0625 mm as estimate or average.

SAND GRAIN MEASUREMENTS

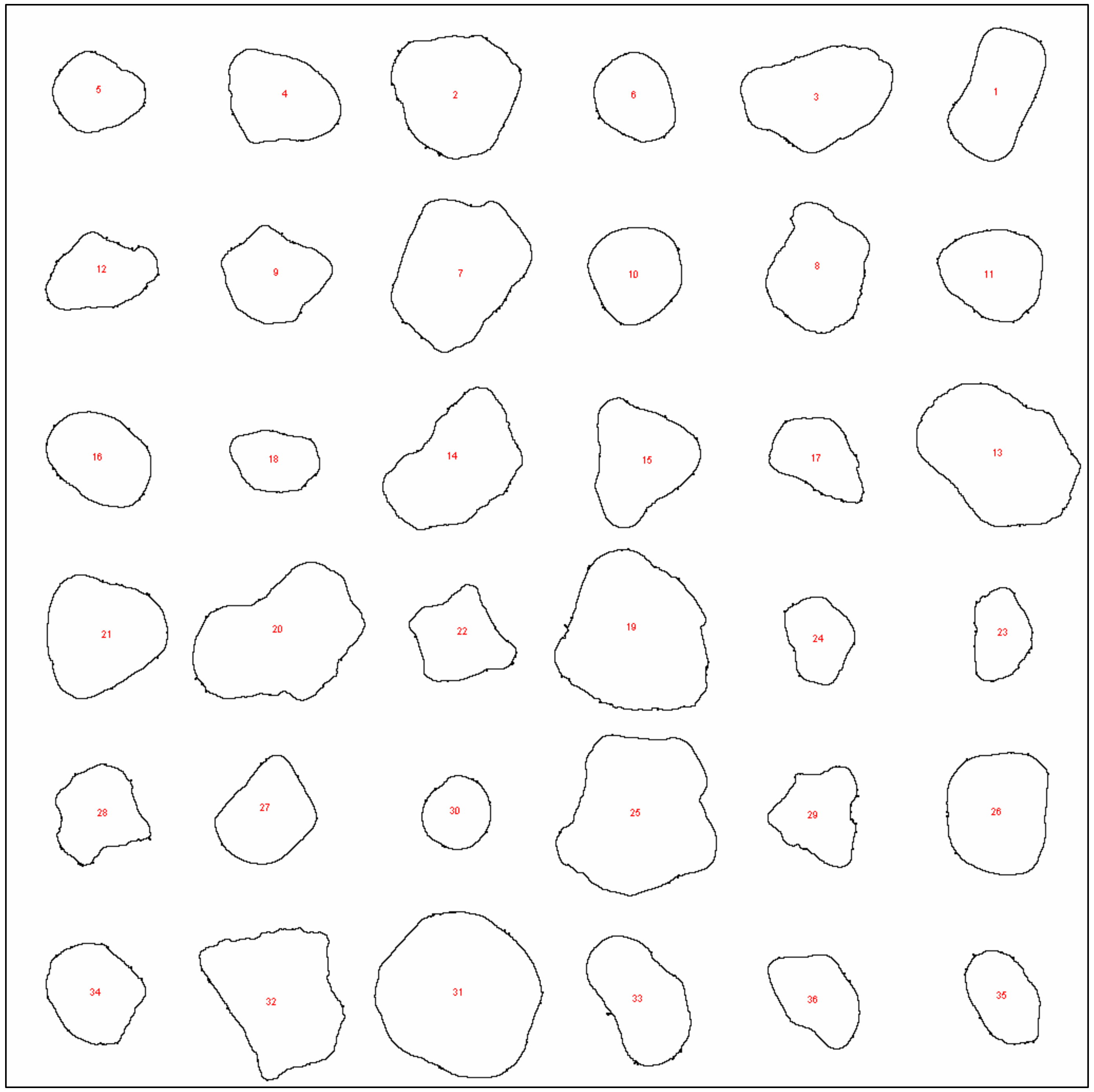
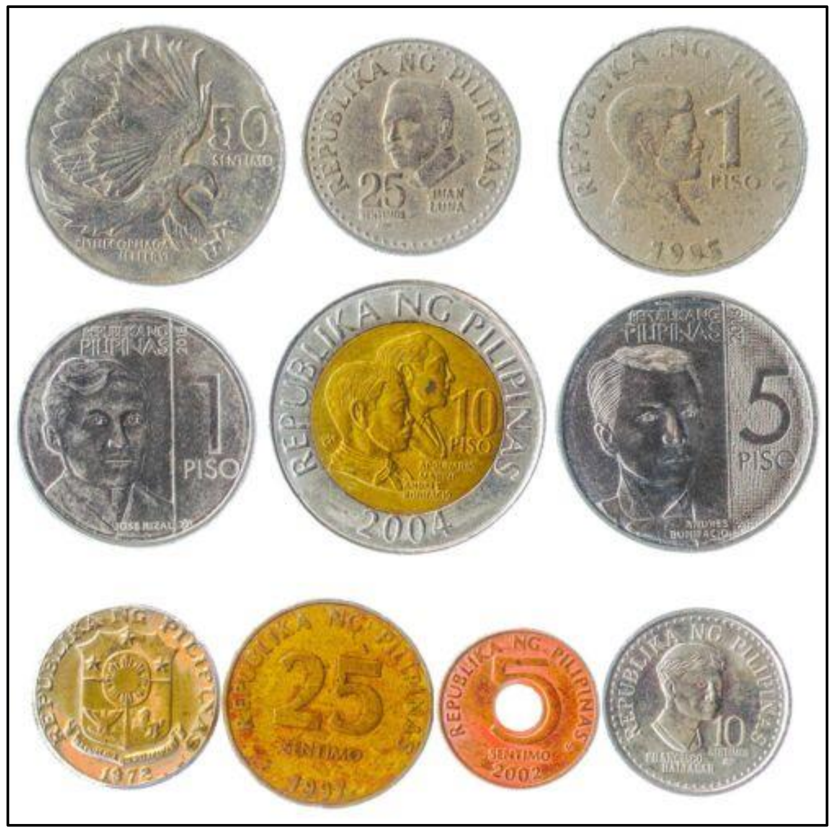


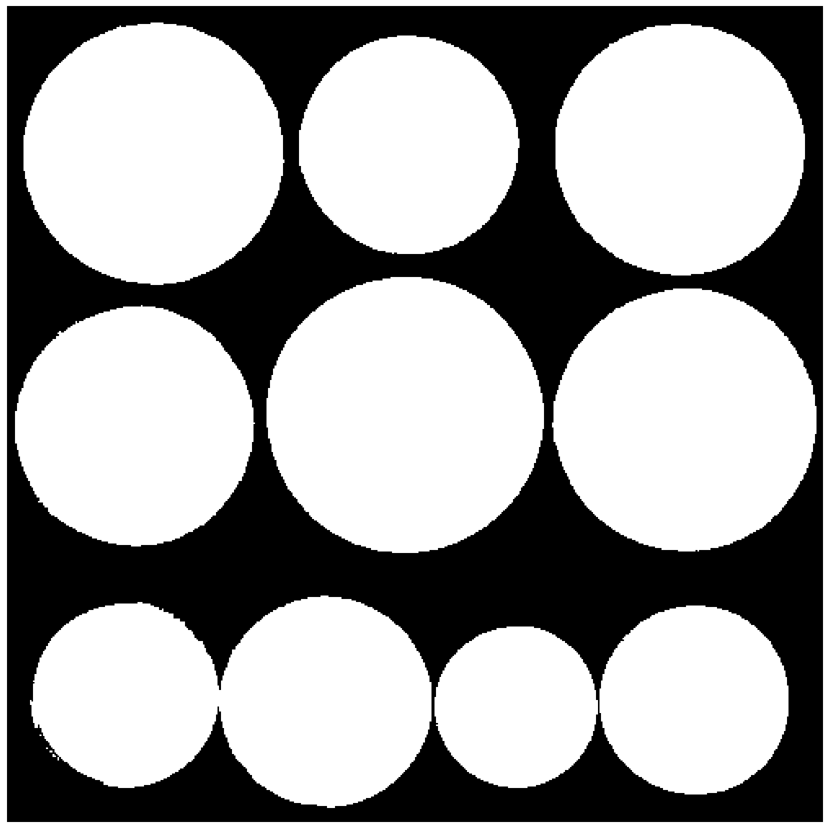
Figure 8. Labelled blobs or sand grains after measurement in ImageJ.

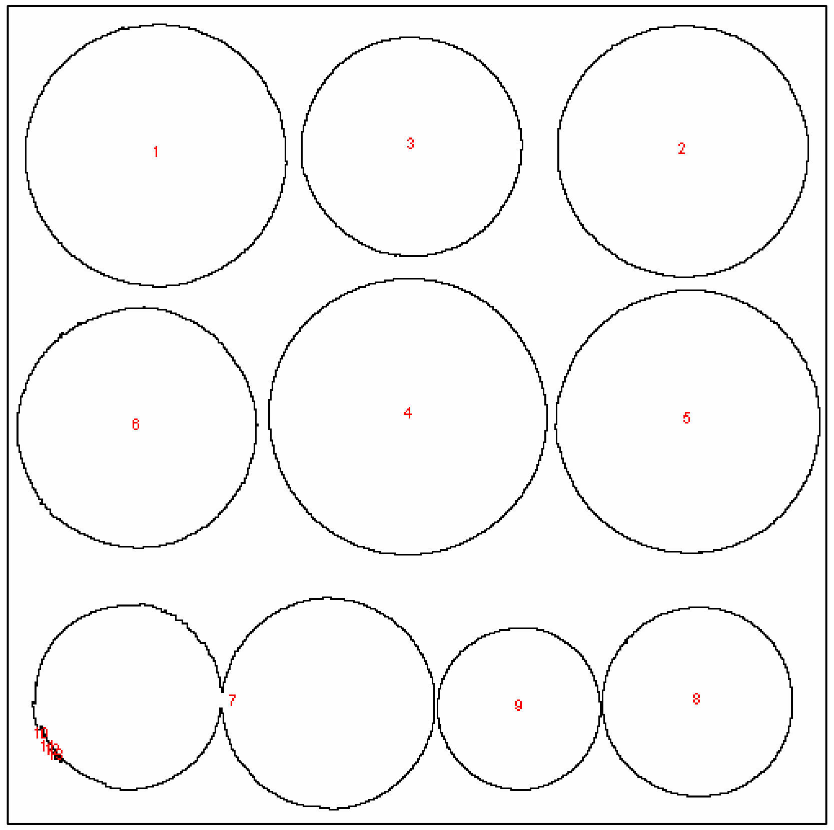
One of the interesting results observed is that ImageJ was able to quantify the number of sand grains in the image. This can be useful when one wants to count a somehow uncountable setup of objects on large scale – such as rice grains.

Other **extracted features**, from Figure 7, were the aspect ratio (AR), Roundness or inverse of aspect ratio, and solidity of grains, which is the ratio of the area and convex area.

ANALYSIS ON PHILIPPINE PESO COINS


(a) Original image from this site


(b) Binarized image after thresholding



(c) Identified coins through ImageJ

Figure 9. Images of interest for the Philippine peso coins analysis.

We also considered trying out other images – in this case is an image of the Philippine coins series – using ImageJ. Executing the same processes carried out in the sand grains, we obtain the result as shown in figure 9c. In this latter part of the activity, we validate if the ImageJ measurements were consistent with the actual measurement – diameter and area of coins.

ANALYSIS ON PHILIPPINE PESO COINS

Coin number label	Coin denomination	ImageJ perimeter (mm)	ImageJ diameter (mm)	Actual diameter (mm)	Percent error
2	Php 1	80.067	26.641	24.000	10.158
4	Php 10	88.327	28.115	26.500	5.744
5	Php 5	84.235	26.812	25.000	6.758

Table 1. Summary of some coin measurements derived upon segmentation using ImageJ.

Table 1 shows some of the coin measurements derived upon segmenting the coin image in ImageJ. The discrepancy in values between the experimental and actual diameters were minimal, which can be attributed to how ImageJ segments and measures the coins after thresholding. It can also be observed from figure 9c that coin number 7 were labelled on two coins. This is because the spacing between those two coins – the 25 cents and 10 cents – were minimal that ImageJ treated them as one blob or coin. This is where morphological operations come into place, which is yet outside the scope of this activity! Nonetheless, the measurements ImageJ reported, and the actual values were still worth comparing to as some measurements can be carried out using applications like this. Interesting!

REFLECTION

I find the activity generally fun since it is somehow a breather from the recent coding activities done in the course. I find ImageJ convenient and straightforward to use since the interface was generally intuitive. Overall, I would give myself a score of **110/100** !

REFERENCES | GITHUB

1. M. Soriano, Applied Physics 157 – Feature Extraction, 2023.
2. Documentation_ (nih.gov)
3. ImageJ User Guide - IJ 1.46r | Analyze Menu (nih.gov)
4. Coins of the Philippine peso - Wikipedia