ACTIVITY 4: FEATURE EXTRACTION FROM IMAGES

Julian Christopher L. Maypa
2020-07587
App Physics 157 WFY-FX-1

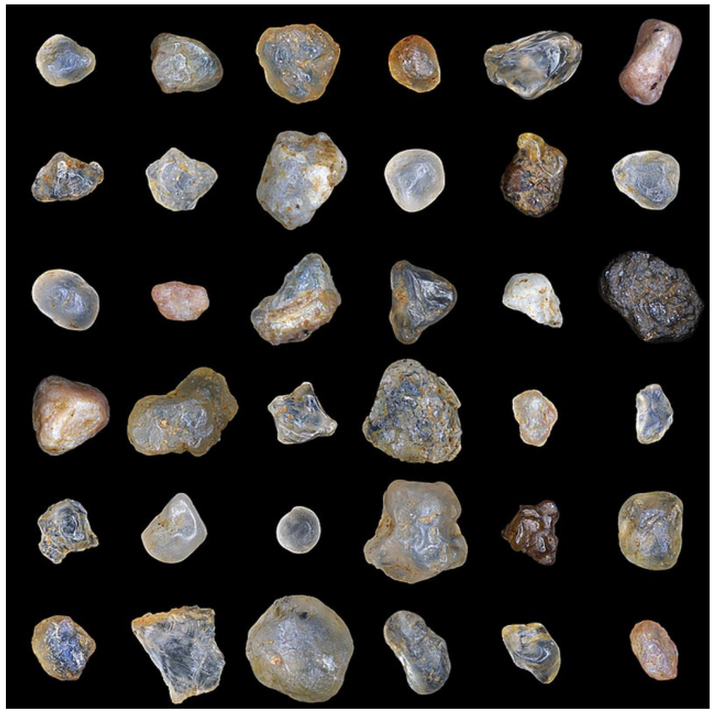
Images that I used

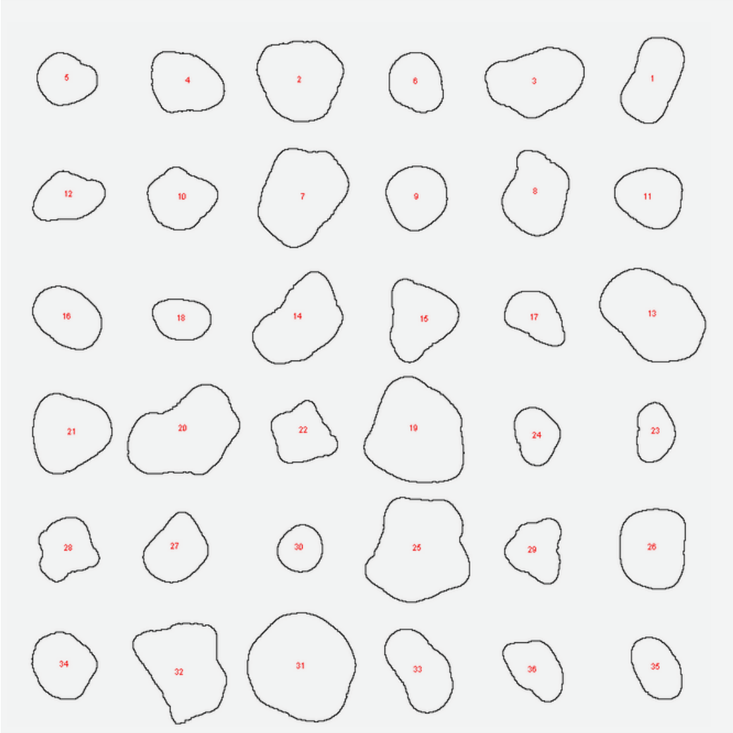
The images that I used for this activity are:

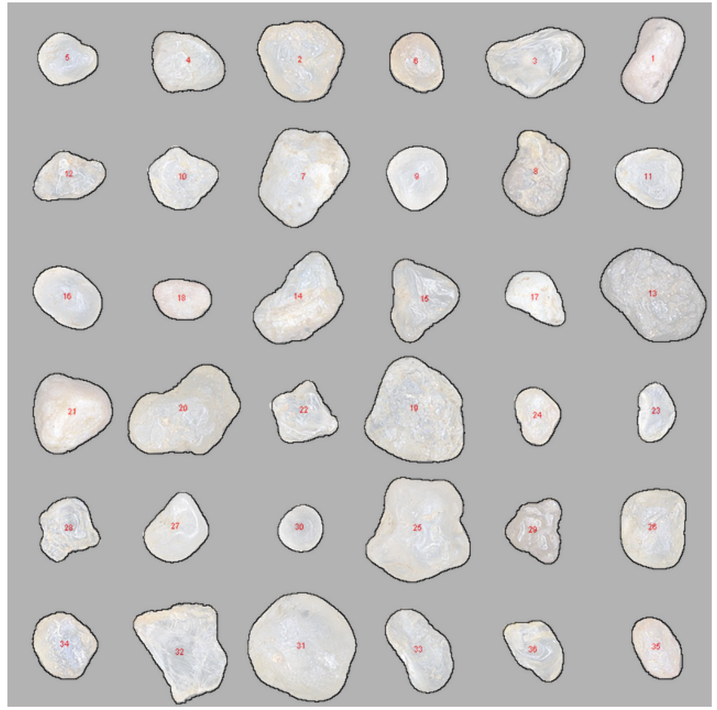
- Sand grains
- Circular objects
- Rice Grains
- Red Dice

Sand Grains

For the processing of the sand grains, I converted the picture to grayscale, applied thresholding, despeckled the image, removed outliers of radius 10, and then applied the *fill holes* function. The results are as follows:







The data that was collected from the sand grains were its area, center of mass, bounding rectangle, shape descriptors, centroid, and perimeter. Below are the first five rows of the data set.

	Area	X	Y	XM	YM	Perim.	BX	BY	Width	Height	Circ.	AR	Round	Solidity
1	4690	733.771	64.878	733.771	64.878	278.836	697	17	72	97	0.758	1.847	0.542	0.958
2	6912	332.658	64.015	332.658	64.015	319.22	283	21	99	91	0.852	1.054	0.949	0.975
3	6318	600.467	66.477	600.467	66.477	316.635	543	28	112	80	0.792	1.569	0.638	0.961
4	4510	202.395	68.235	202.395	68.235	262.35	164	33	83	69	0.823	1.302	0.768	0.972
5	2999	66.62	64.502	66.62	64.502	208.551	34	34	68	60	0.866	1.187	0.843	0.966

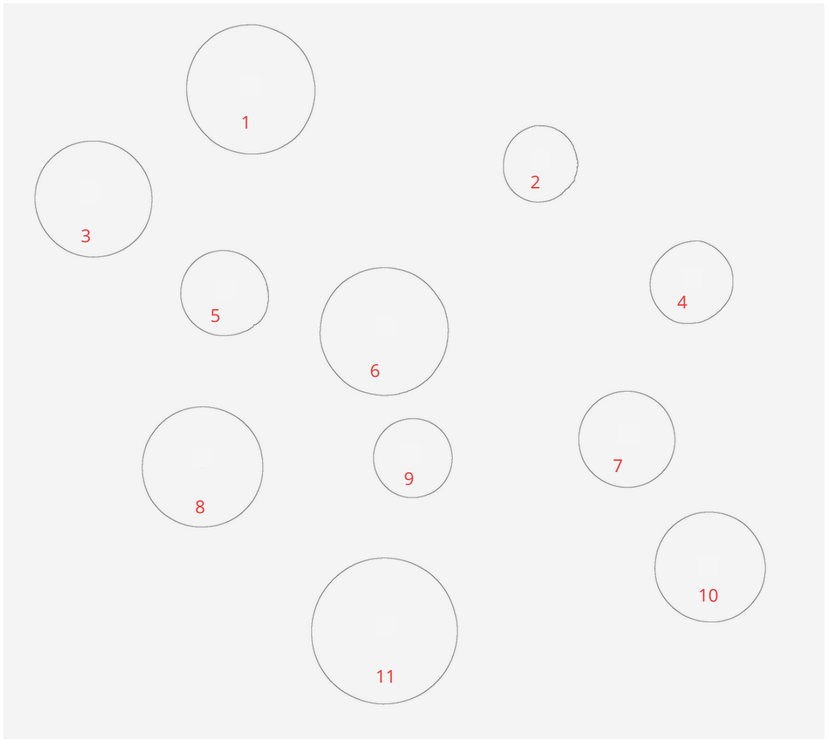
The summary of the data can be seen below.

Slice	Count	Total Area	Average Size	%Area	Perim.	Circ.	Solidity
sand.jpg	36	185305	5147.361	28.954	273.109	0.826	0.963

The data above were calculated based on pixels because there was no scale for reference. But the data shows that ImageJ is capable of making measurements solely based on the picture. Although it is also worth noting these measurements are not 100% accurate as I had to clean some noise after thresholding. But visually inspecting the overlap between ImageJ's outline with the edges of the sand grains in the previous slide, it can be seen that the outlines are a good estimation of the grains' shape. To see how reliable ImageJ's measurements are, I measured the perimeter and area of different circular objects.

Different Sized Circular Objects

I tried testing the accuracy of ImageJ by letting it measure different sized circular objects with known radii. I processed the image by first converting it to an HSB stack (to better see the dark green circles), used the brightness layer, applied thresholding, used fill holes, removed outliers with a pixel radius of 20, and used despeckle. The following are my outputs.



I used the same parameters from from when I analyzed the sand grains. Below are the first five rows of the results.

	Area	X	Y	XM	YM	Perim.	BX	BY	Width	Height	Circ.	AR	Round	Solidity
1	5.77	5.192	1.809	5.192	1.809	8.972	3.844	0.45	2.694	2.713	0.901	1.02	0.98	0.996
2	1.96	11.263	3.373	11.263	3.373	5.275	10.494	2.569	1.562	1.606	0.885	1.044	0.958	0.99
3	4.703	1.891	4.11	1.891	4.11	8.096	0.663	2.894	2.456	2.438	0.902	1.017	0.983	0.995
4	2.341	14.441	5.863	14.441	5.863	5.696	13.569	4.988	1.738	1.731	0.907	1.059	0.944	0.993
5	2.595	4.64	6.093	4.64	6.093	6.06	3.719	5.188	1.844	1.788	0.888	1.042	0.959	0.993

The summary of the data can be seen below.

Slice	Count	Total Area	Average Size	%Area	Perim.	Circ.	Solidity
Brightness	11	45.033	4.094	16.848	7.396	0.899	0.994

Visually inspecting the outlines in the previous slide, it seems that ImageJ was able to properly trace the outlines of the objects. With this, I will validate the accuracy of the results by comparing the measured area and perimeter of ImageJ with the actual measurements of the coins. I will use the 25 centavo (new), 5 peso (new), and 20 peso coins for validation.

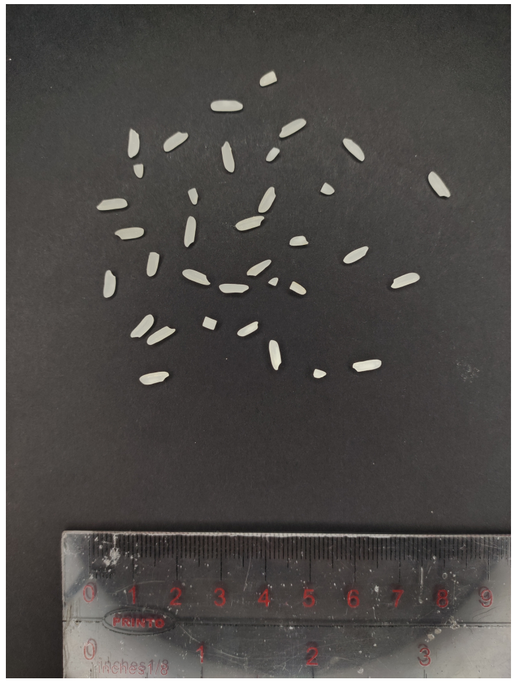
The respective diameters of the coins are 2.000 cm, 2.5000 cm, and 3.000 cm [1]. I used the diameter released by Bangko Central ng Pilipinas because the coins are rigid so the wear on the coins will have minimal effect on its radius. With this, their respective perimeters are approximately 6.283 cm, 7.854 cm, and 9.425 cm. Thus their respective percent errors are equal to 0.8%, 6.7%, and 7.7%.

For the area, the coins have respective areas of 3.142 cm^2, 4.909 cm^2, and 7.069 cm^2. Thus their respective percent errors are equal to 2.1%, 2.8%, and 4.2%. With this, all of the percent errors are within an acceptable range. Thus ImageJ can reliably make measurements from pictures.

Some possible sources of error are the: creation of the scale, the warping of the camera, and the removal of noise. For the creation of the scale, the ruler in the image is not perfectly straight. Thus creating a scale from it will introduce error. For the warping of the camera, the sides of the coins that are far from the center can be seen, thus increasing the perimeter and area. And finally, the removal of noise affected the outlines of the coins. As I did thresholding, there were noise in the binarized image. So I had to remove them, which consequently changed the shapes of the outlines. But overall, the errors are below 10%, so the measurements are still somewhat reliable.

Rice Grains

Since I have successfully shown that ImageJ's measurements are reliable, I can now use it on different pictures. I recall that the dimensions of rice grains were measured in an experiment in Physics 106.1. So I tried using ImageJ to get the area and the perimeter of the rice grains. For the image processing, I first used an automatic threshold to binarize the image, removed outliers with a radius of 10 pixels, and then despeckled the image. The following are my results:



I used the same parameters from from when I analyzed the sand grains. Below are the first five rows of the results.

	Area	Min	Max	X	Y	XM	YM	Perim.	Circ.	AR	Round	Solidity
1	0.083	255	255	4.622	0.482	4.622	0.482	1.174	0.76	1.666	0.6	0.977
2	0.152	255	255	3.687	1.08	3.687	1.08	1.729	0.637	2.876	0.348	0.984
3	0.129	255	255	5.163	1.568	5.163	1.568	1.637	0.607	2.817	0.355	0.973
4	0.127	255	255	1.615	1.926	1.615	1.926	1.567	0.65	2.448	0.408	0.974
5	0.132	255	255	2.54	1.869	2.54	1.869	1.582	0.664	2.38	0.42	0.969

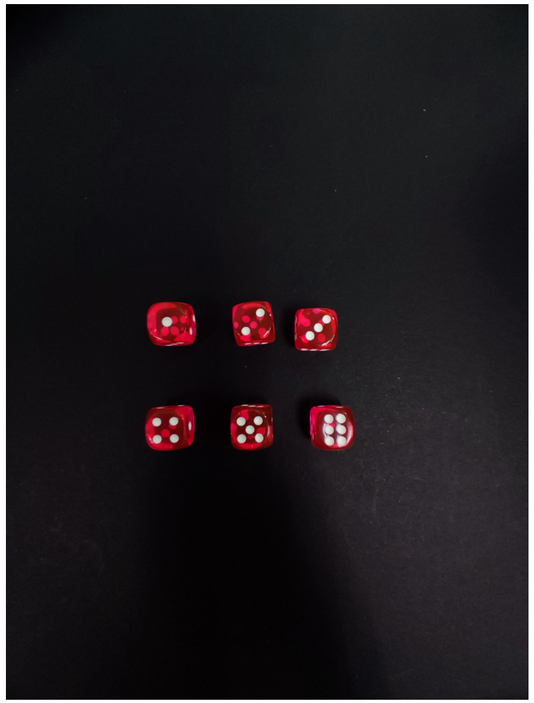
The summary of the data can be seen below.

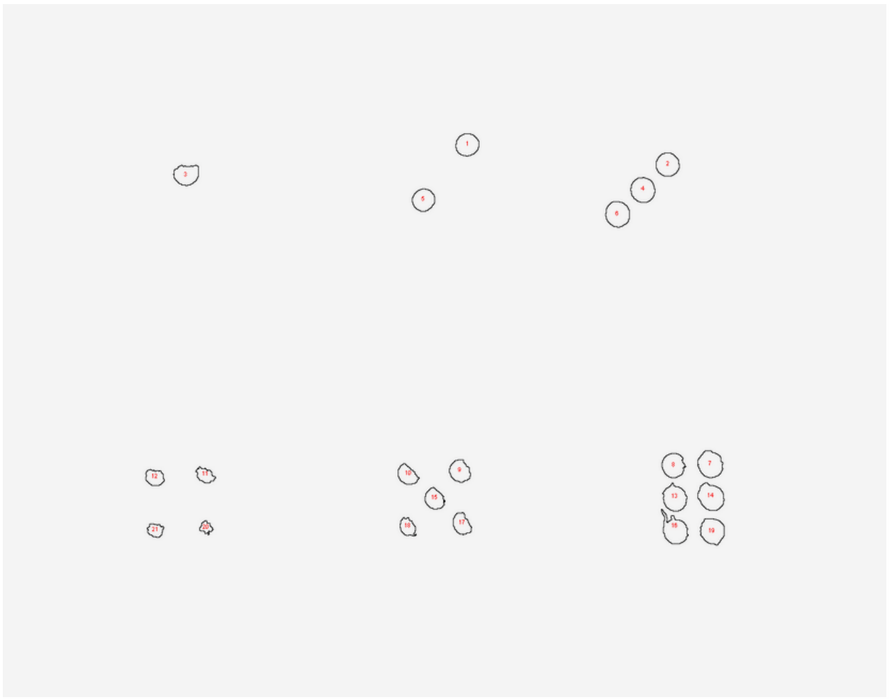
Slice	Count	Total Area	Average Size	%Area	Perim.	Circ.	Solidity
rice.jpg	35	3.682	0.105	5.213	1.402	0.665	0.973

Looking at the overlap between the outline and the grains of rice, it can be seen that ImageJ has successfully traced the grains and measured their areas and perimeters. But there are still sources of error. These are the same as the sources that I mentioned when I measured the areas and perimeters of different circular objects: the ruler is not perfectly straight, the warping of the camera, and the removal of the noise. The ruler being not perfectly straight introduces an error to the scale. The warping due to the camera's lens captures the sides of rice grain, thus making it appear larger. And finally, the thresholding of the image results in noise in the binarized image and removing the noise can affect the size and shape of the grains.

Counting Dice

An experiment in Physics 191 made us sum all the faces of a dice. And I realized that with ImageJ, I can automate this process. I processed the image of a dice by first separating the different RGB channels and only using the green channel (it showed the highest contrast contrast between the points of the die and the die itself), used thresholding, applied despeckle, and analyzed the results by only considering particles with a dimension greater than 200 by 200 pixels. My results are as follows:





For this part, I did not need to measure anything because all I needed was the count of the dots on the die. Below is the summary of the results.

Slice	Count	Total Area	Average Size	%Area
Green	21	18565	884.048	1.233

For this part of the activity, the dice I used were reflective. And while using the green channel, the glare appeared as bright spots along with the dots on the dice. So what I did was that I decreased the threshold such that the glare would be significantly smaller than the radius of the dots. I was able to successfully do this and ImageJ was able to sum up all the face-up sides of the die.

It can also be observed that the size of the outlines of the dots do not reflect the actual size of the dots of the die. This is irrelevant because I was only interested in the number of dots, and not the actual measurements of the dice.

Reflection

For this part of the activity, the only hassle was taking the photos and getting the correct lighting. Other than that, I had no trouble. ImageJ was easy to learn and I really had fun with this activity. This is probably the easiest activity so far.

I'd like to thank my instructors, Sir Rene Principe Jr. and Sir Kenneth Leo, for guiding me throughout the activity. I would also like to thank my professor, Ma'am Jing, for guiding me in my coding while my classmates and I worked in R202. I would also like to acknowledge my classmates: Abdel, Johnenn, Jonabel, Richmond, Lovely, Hans, Genesis, Jeruine, Rusher, and Ron for helping me complete this activity.

Self Grade

Technical Correctness	I understood the lesson and met all the objectives.	35
Quality of Presentation	The images I added to this report are of good quality and all the graphs are properly labelled.	35
Self Reflection	I got the expected results, and acknowledged the contributions of my peers while doing this activity. I also properly cited online references.	30
Initiative	Apart from doing the required tasks, I also applied what I learned to sample images that I took. I also helped my classmates with their images by lending them my dice sets.	10
	Total	110

References

[1] Bangko Sentral ng Pilipinas. (2022). The Philippine New Generation Currency Coin Series. Retrieved April 28, 2023, from https://www.bsp.gov.ph/Coins%20and%20Notes/Coins/NGCCS/NGCCoins.pdf