

Finding a safe landing site on Mercury (for a robotic spacecraft)!



Pinnacle Scholars Summer
Research Program 2022

Benjamin Knobloch, Campbell Tedtsen, Joshua Bernstein, Rebecca Guerra
Dr. Jason Rabinovitch

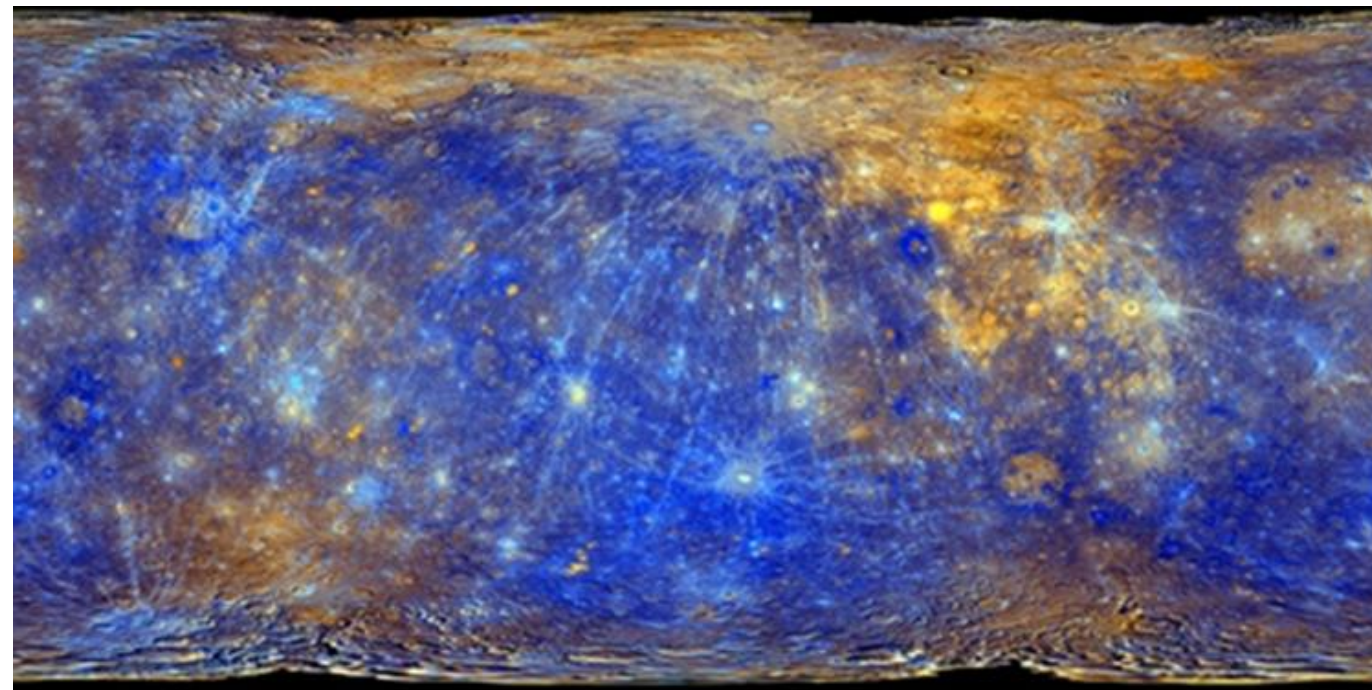
Introduction

The goal of this project was to use almost 700,000 high-and-low resolution images taken by the MESSENGER spacecraft to **identify the best landing locations on Mercury for a future rover mission.** By processing selected high-res photos and analyzing associated metadata in NASA's database, we could determine certain characteristics of well-suited images and draw conclusions about the lower-res planet-wide maps, including a rough estimate of the number of craters in each location for landing.

Study Goal

Find the ideal landing spot(s) on Mercury to land a rover. Studying the terrain and different levels of surface to provide the greatest chance of a safe landing.

Background



- From the MESSENGER Project the terrain of Mercury was made into a colored mosaic which helped to depict the craters and how deep they were. These smaller, more specific area, colored images are obtained with a high resolution of 1 km/pixel.
- The purple being the deep craters and red as the higher elevations; these two elevations having a 10 km separation.
- The light exposed areas can be greater than 400 Kelvin (approximately 126 degrees Celsius), while craters that are always in the shadows can get very low, less than 100 Kelvin (approximately -173 degrees Celsius).
- Mercury has intercrater plains, which are filled with many large craters, and smooth plains, which have smaller craters

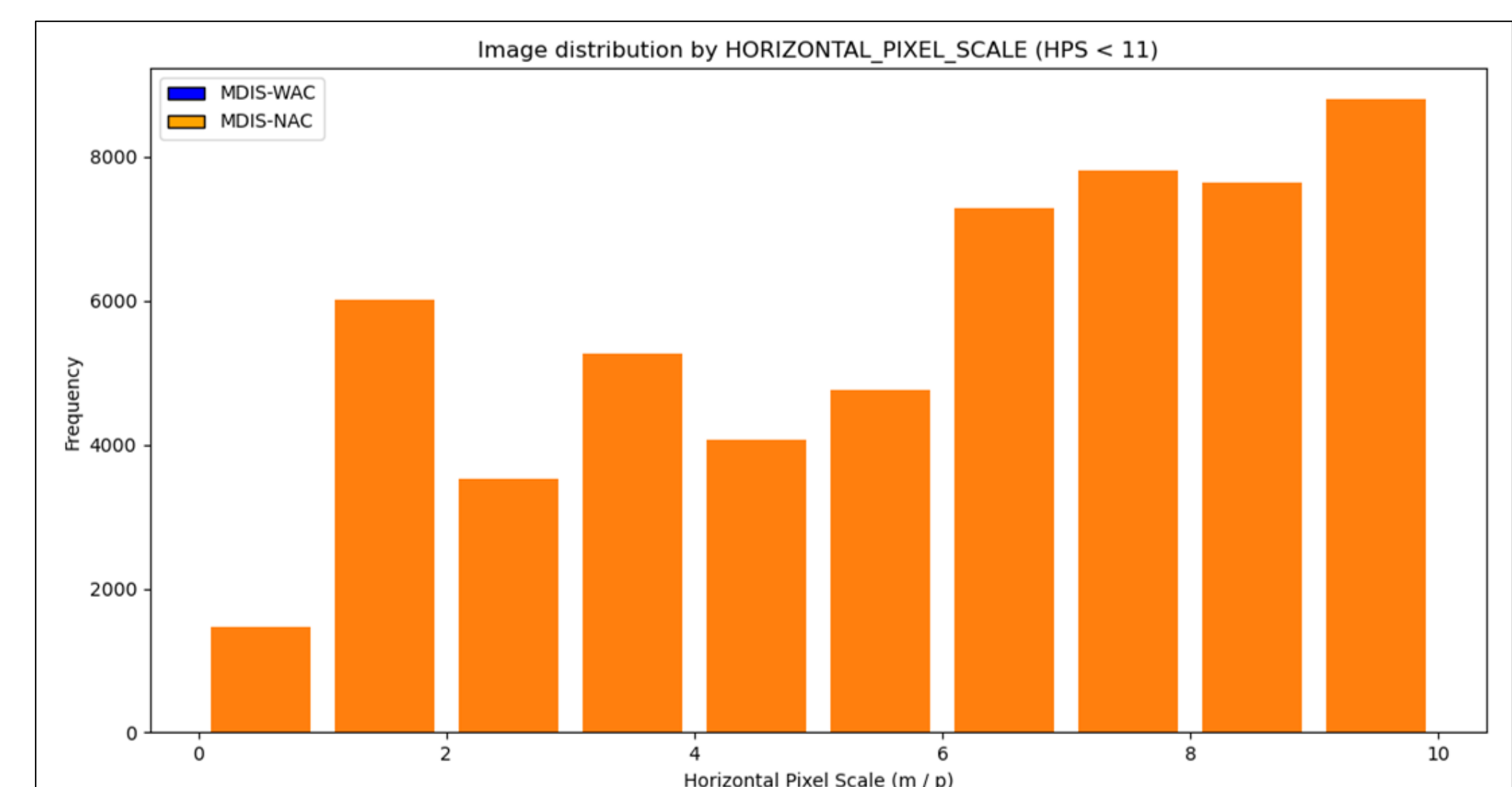
Key Qualities

Using the MSGRMDS Index TAB metadata we created an excel file to find **important characteristics of valuable photos:**

- **Data Quality ID** - A data quality index is used to encode figures-of-merit into a parameter that is included in the label of each EDR
- **Exposure Duration** - Exposure duration of too long results in too many saturated pixels for a usable image
- **Smear Magnitude** - used to check if the images are too blurry to count craters
- **Incidence Angle** - Affects the appearance of light in an image because of the way the light falls on the surface
- **Horizontal Pixel Scale** - The number of surface meters of mercury per pixel of the image, used to find high-resolution images where the craters are easily visible and not too small
- **Longitude / Latitude** - Location of a point on a map, used to check the image of the intercrater plains or the smooth plains
- **IF / RA images** - For every image in the database, there was an IF and an RA (radiance) version, with the IF version being slightly adjusted to be of a higher quality

Method

This histogram represents the horizontal pixel scale distribution from 0 through 10 meters /pixel.



Between our team we each took a range of pixel scales to create scatter plot maps of the location of each image to narrow down the number of images to search through. Each of us went through our assigned set of images and started looking at various qualities that would determine if that image was usable for counting craters.

For example, we found that quality images had a *Data Quality ID* of all "off" bits.

For horizontal pixel scale, we limited usable images to less than 5 meters per pixel. Qualities such as these helped us to greatly narrow down the number of images to count craters on.

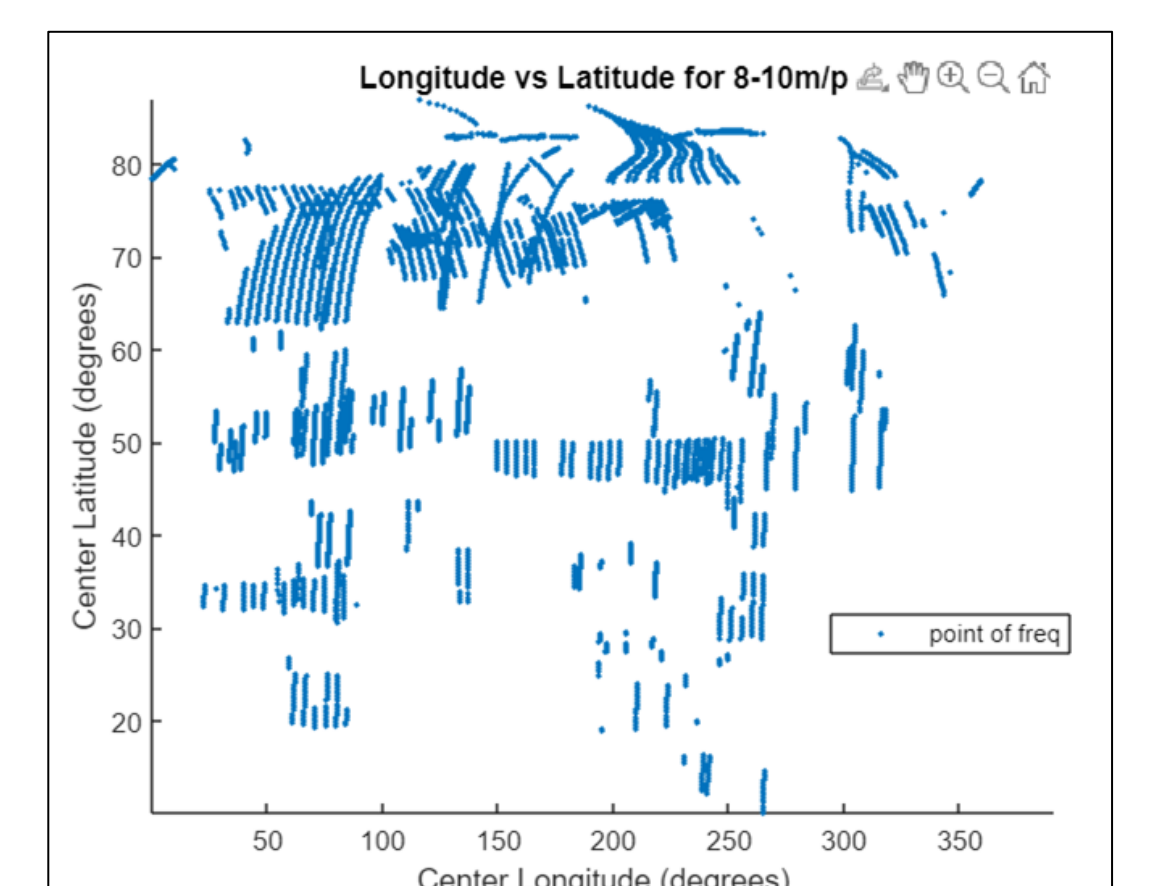
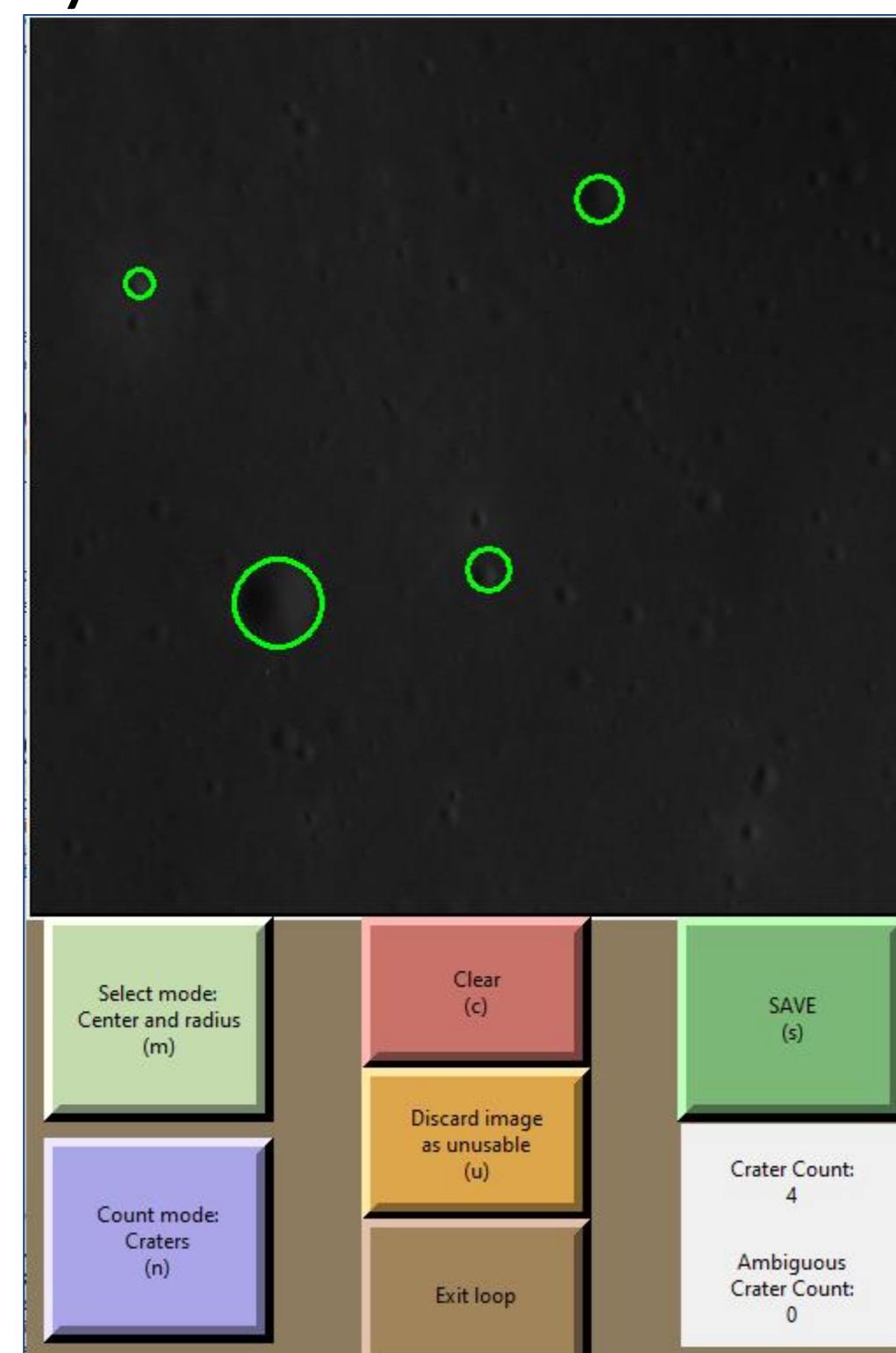


Image Processing

We tried a couple different processes when it came to actually counting the craters of the images. One process was called jmars, which had a crater counting process already setup. However, we had trouble with uploading our own custom map, so we instead used a program developed by our own team members called circledrop.py.



Circledrop.py is fed a list of Mercury images, and displays them to the user one at a time. With each image, the user is asked to select any and all crater formations they may see. If they see a formation that could be a crater, but they aren't sure, they can mark it as an "ambiguous crater". The program automatically keeps count of the number of craters, and saves their location on each image once the user is finished. Using the haversine formula (far too long to write), the approximate real-world locations of these craters can be acquired, along with their radii.

Conclusion and Ongoing Work

Looking at these various qualities, we were able to narrow down the initial number of images to 3,682. From these images we were able to count craters for a few example images. This work will be continued further by other students in order to exemplify the difference, or lack of a difference, in the number of craters in the intercrater versus smooth plains of Mercury.