

Daily Log

Detail for each day about what you researched, coded, debug, designed, created, etc. Informal style is OK.

Tuesday September 3

Figured out how to pixelate images by looking within a square of pixels of a certain size, and then if a certain proportion of them are not white, make everything black, else, make everything white. Determined that if a region is less than 70% white, it is a good idea to make everything black. Wrote method that determined what degree image should be pixelated given width of object, pixels in image, and width of pixelated pixel. This is useful for maintaining objects' relative size to one another (a globe might look the same size as a marble based on the scale of some pictures, so a notion of scale needs to be defined).

Thursday September 5

Used pillow rotate methods to generate rotated forms of images to be pixelated. Had to troubleshoot default background color of rotated images being yellow, as needed to be white. Found some weird solution in stack overflow using converting an image to a different format and then converting it to RGB. Created program that uses previous methods to generate a matrix of integers that represent black and white coloring of pixels in pixelated image. Created class used for images that stores and approximate width in feet of object in the image (for scaling purposes). Started work on simplifying methods to make testing images easier.

Timeline

Date	Goal	Met
August 30th	Wanted to learn how to use pillow and image processing so I could work on pixelating images the following week	I did learn enough to do so
September 6th	This week I wanted to be able to generate a black and white pixelated form of an image given an image of an object with a white background	Was able to get accomplished, also was able to convert to matrices
September 13th	I hope to be able to create several rotations of several images and convert them into arrays using only one method. I also hope to be able to insert an array of integers inside of another array of integers so I can begin the process of using the matrices to represent ways to stack the objects	
September 20th	I hope to be able to specify a column in a matrix representing a packing container and have my code place an object in the optimal position and rotation in this column	

Reflection

On Tuesday, I worked on testing out images and pixelating them, which I was able to do. The outputted images look fairly similar to the originals, which is good, as a concern of mine was that the conversion would not be a great representation of the original objects. Results are shown below, with different degrees of pixelation shown:



On Thursday, I was able to fairly quickly modify my code on Tuesday to generate images of different rotations and pixelate them, since pillow has the built in command `.rotate()` which rotates images. An issue I had to deal with was that the default background of a rotated image is yellow, and I wanted it to be white. I found some weird solution on Stack Overflow that has been working by changing the image format.

I have not seen my timeline I created last year since June, but I think the process of pixelating

images and converting them into arrays is going quicker than I initially anticipated. This is good, because I expect I will need to work on and modify my algorithm more than I initially anticipated in order to be efficient and have efficient run time, so I will likely spend several months testing it out and modifying it. Something that might be an obstacle is that if there is any pure white in an object it will be considered as part of the background, creating a hole in the object not initially meant to be there. I can modify my code to account for this in the future.

I'm pretty happy with how things are going so far, I believe I will start being able to work on the actual packing algorithm soon since I am confident in representing objects in a form that can be packed.