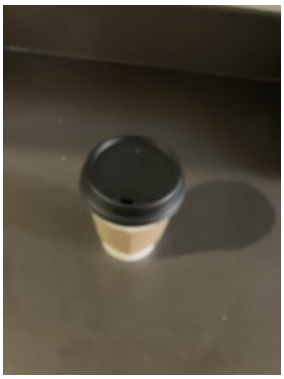
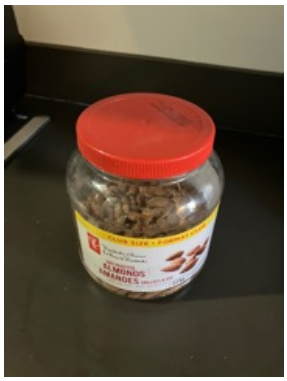
Owen Coolidge
40026478

A**1.**
My dataset consisted of around 50 images of a large drip coffeemaker and around 50 images of a pair of headphones.


As well as more than 50 images of a paper coffee cup, calculator, jar of almonds, coffee filters, container of detergent, mechanical pencil, shoe, sandal, toaster oven, roll of toilet paper, and a colourful coffee mug.







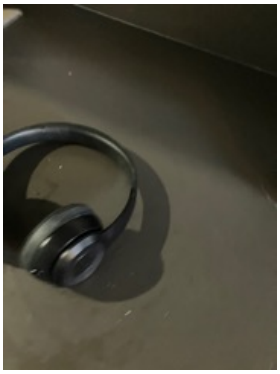
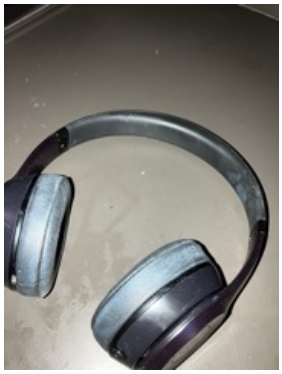
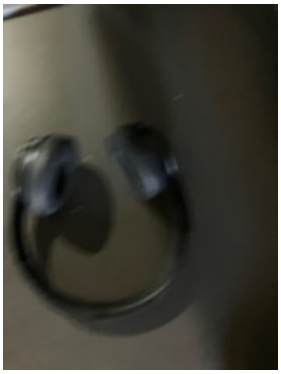
Owen Coolidge
40026478



While there are multiple images of each of these “noise” objects, that are far fewer of them than of the important objects, the coffee maker and headphones. Images of these objects had many variations in angle, lighting, zoom level, and blurriness level. As well as combinations of these variations.






An important thing to note is that I was unable to provide a neutral background for my objects. This means that different objects that look nothing like each other will still share features originating from the background. However, this problem is partially neutralized due to different angles. Furthermore, this inclusion of this noise in the background could contribute to the robustness of the model. As long as the noise is not always the same; because if it were the exact same background behind each object, then the model would believe that the background is part of the object and would not be able to recognize it in different settings.

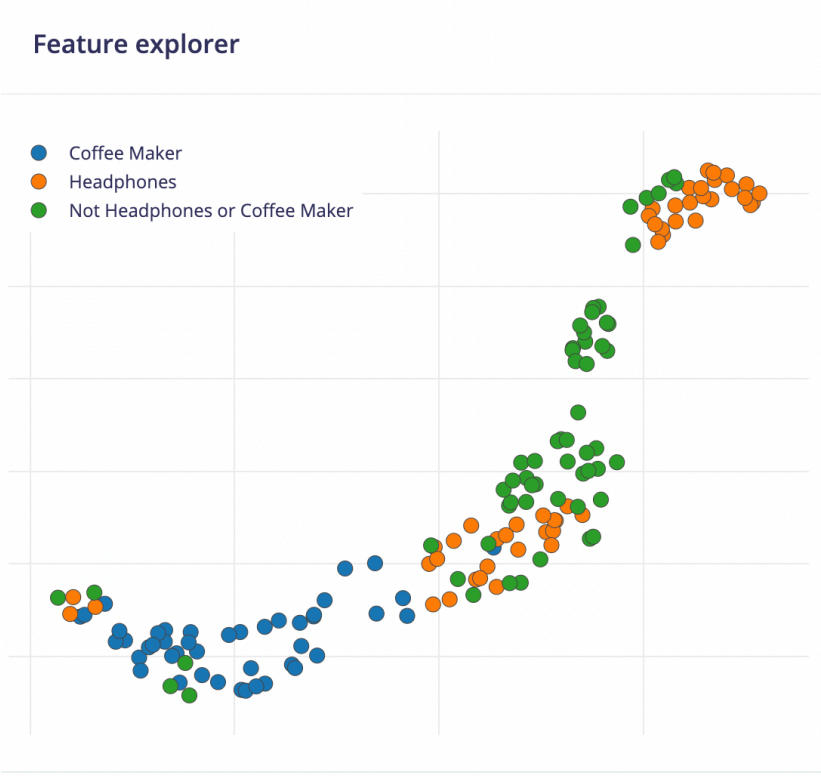
Owen Coolidge
40026478

2.
By creating an image classification model, we learned about the mechanics of machine learning. The possibilities and limitations of machine learning are more obvious to me know. For example, I know understand how large the dataset must be for a model to be able to operate successfully and classify a wide range of objects in even a relatively predictable environment, let alone any environment. My biggest takeaway from the task was that no model can ever be correct 100 percent of the time.

3.
First, we uploaded our images in three batches: object 1, object 2, and other. These labels were necessary so that the graph of features could be easily read after each image was processed. We then used the data explorer, which applied a pre-trained image classification model to our dataset. The resulting graph displayed which images of different classes looked too much like each other. These outliers were removed to improve the accuracy of the model we then created later.
Next, we created an 'impulse' which is a classification program derived from the data we uploaded. It uses small versions of out images to increase efficiency. And most importantly, it uses Transfer learning. Which is a method of using certain insights from other accurate models to increase the accuracy of our model and reduce its raining time. In this instance, the 'other' model contributes basic features already found in most photographs. These building blocks make it so that our model doesn't have to learn everything from scratch. Once these settings were in place, we trained the model.
Next, we deployed the model so that we could perform live classification.

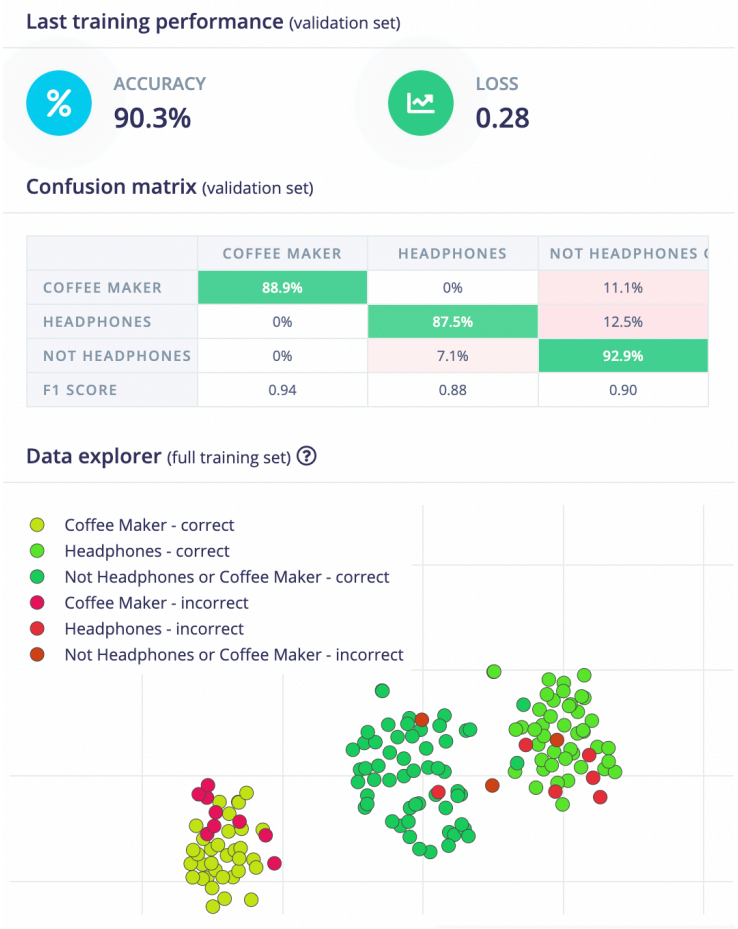
4.
My dataset had a 100% accuracy. Which means that my precision and recall were also 100%.

5.
This graph shows the correlation of each image before transfer learning has been applied. When I looked at the images the dots represent, I found that lighting had a big influence on where the image ended up on the graph. Images taken with a flash, regardless of the object being represented, ended up in the top right corner, and images taken in low light ended up in the bottom left corner.



Owen Coolidge
40026478

This confusion matrix and dot matrix show which images the model had trouble recognizing as the class that they are a part of. It is possible that the accuracy could be improved if the model had more training time. It is also possible that some images are just too similar to others of a different class to a degree that not enough unique features exist in the image. Overall though, this dot matrix shows much more integrity in the dataset than the previous one from before transfer learning had been applied.



Owen Coolidge
40026478

Despite the imperfect confusion matrix, the model testing results are perfect (depending on how you measure it). This matrix and dot matrix show that every test image (the 20% of image data that was withheld from training) has been accurately classified by the model. This could be because we were lucky and only the distinct images happened to be used as test images. This is unlikely however. Overall, because my input images had considerable background noise which was shared between classes of images, I am very surprised by these results.
However, the live classification is far less impressive. What I found in testing is that human heads are highly misclassified as headphones for example. And there are probably an infinite amount of other objects that would be misclassified as headphones as well.

6.
If the goal of my model is to be able to recognize my specific coffee maker and my pair of headphones in any scenario, then I think this could be mostly achieved with many more differing images of these objects in front of different backgrounds. As well as adding all of these backgrounds in isolation to the “other” set of images. Since I’m only interested in these two objects, I only need to know if one or both is in the frame. The third category is pretty much an ”everything else” category.

I think a “better” version of this model is one which recognizes the object no matter what else is in the same frame. And also recognizes the common state that neither are in the frame. This means that while the dataset for the objects of interest needs to be greatly expanded, the “other” dataset needs to be expanded far more. Depending on how many scenarios I plan to deploy my classifier in, I could need millions of photos. It might be easier to think about the model as a “not headphones or coffee maker” classifier instead, because this class of images would need to require the most attention.

Owen Coolidge
40026478

C.
Object detection could be modified into beautiful object detection. What I mean by this is instead of training the model to classify objects themselves, it could be trained to classify beautifully composed scenes. Once completed, it could be deployed on a high quality camera the user of which would never have to actually click a the shutter. If the frame is determined by the algorithm as being beautiful above a certain threshold, it would automatically take a picture. You could go a step farther and deploy the camera on a robot that has three axis rotation and hook the robot up to the algorithm as well. If the frame is not right, the robot would pivot in a direction that it thinks is more likely to yield a better composition.
The training data would consist of only two categories. One consisting of photos taken by famous artist and another one consisting of images that are not well composed. The “good” images would just have to be good in a composition sense, computers don’t understand conceptual art so only certain kinds of “good” photos would be included. These would be photos that follow common photography conventions such as the rule of thirds, foreground-background isolation, symmetrical balance, golden ratio. Likewise, the “bad” photos would have to be photos of similar subjects but with off-kilter framings. As well as many horrible photos of nothing.

Here are some examples of good composition.

Owen Coolidge
40026478

Here are three images with bad composition.

For this purpose, it is less important to teach the model about objects in the world, and more important to teach it about the overall relationship between things in the world. It still doesn't really matter that it does not know what it is looking at, it just needs to have a general idea of what is a compelling frame.