SMART TECH CA1 – IMAGE CLASSIFICATION

Michael Flynn & Kyle McQuillan

For this project we were tasked with creating an image classification model based on two datasets Cifar10 and Cifar100. The purpose of this model was to predict different hazards one might come across on the road such as vehicles, pedestrians, animals, and other hazards. The goal was to train the model based off certain classes from Cifar10 & Cifar100 to create an accurate enough model that could when tested with an image to be able to accurately identify what the image was.

The first stage of this project would prove to be the most difficult and time consuming as it required us to take the data from both Cifar datasets and combine, then once combined we had to filter certain classes out of each dataset to train our model. The reason why we found this time consuming was we could not agree on the best approach in download the data and spent a lot of our time going back and forth from unpickling the data or importing the data sets. In the end we chose the later option as it meant we did not have to spend time worrying about the labels and matching them with the data.

The first function shown below was used to filter the datasets based on selected classes and done so by returning the filtered images and labels. Once we had our filtered dataset the next was to combine the two datasets together to begin our training, this was done using the built in concatenation function and merged the training labels, training images, test images and test labels of the datasets

A screen shot of a computer program

Description automatically generated

Once the data had been combined, we then wanted to make sure that the classes we selected were working. To do this we ran a for loop that selected five random images from the list of unique classes.

A group of images of people

Description automatically generated

Once the two datasets had been combined into one, we created our first model which we simply called alpha model, this model was based off of one we had worked on in class as we felt like this was a good starting point, below is a screenshot of what it was like

A screen shot of a computer program

Description automatically generated

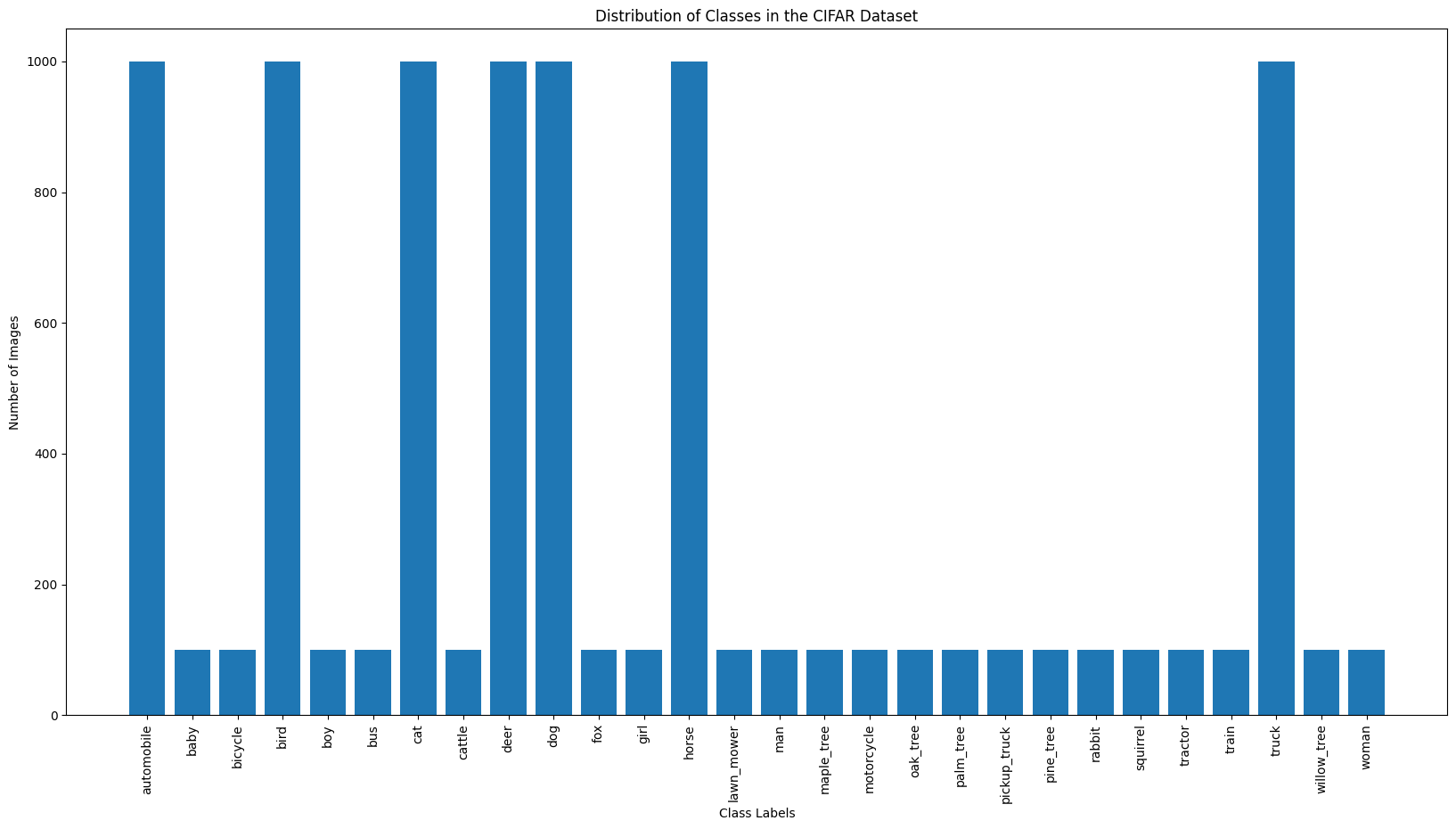
As this is the first model we wanted to get a general idea of what the data would look like without any augmentation. We used preprocessing to grayscale the image. The accuracy came out around 55%. We ran this model for 10 epochs and looking at the screenshots below the data is underfitted

A graph of a training loss

Description automatically generatedA graph with a line and a red line

Description automatically generated

Looking at the distribution table shown below we can see that the there are certain classes that have very low data when compared when to the other classes (cifar10 has 4500 more then cifar100), so try and improve the accuracy without making changes to the model we then decided to augment some of the data in the hopes of improving accuracy this way



Once we augmented the data to added new batches we then ran the model again and the results were