Project Title : Traffic Sign Image Classification

Capstone Project Report

Tatenda Nyamukuvhengu: u3260725

Unit Code: 4483

Unit Name: Software Technology 1

Introduction and Background

My Capstone Project title is Identifying Traffic Speed Signs.The dataset I was allocated consisted of 6164 images in PNG format. These images were between training and testing data and separated into 58 (0-57) classes. The dataset also came with two additional csv files, in the respective train and test folders for those looking to modify their own train and test data. The dataset referenced throughout this project and report is publicly available through Kaggle, at this link:

https://www.kaggle.com/datasets/ahemateja19bec1025/traffic-sign-dataset-classification

This dataset allows development of software tools for automatic visual predictive model for classifying different traffic signs based on camera images. The software development for predictive model has been done in three different stage and will be annotated as such:

Stage 1: Exploratory Data Analysis for Visual Data

Stage 2: Predictive Model Development

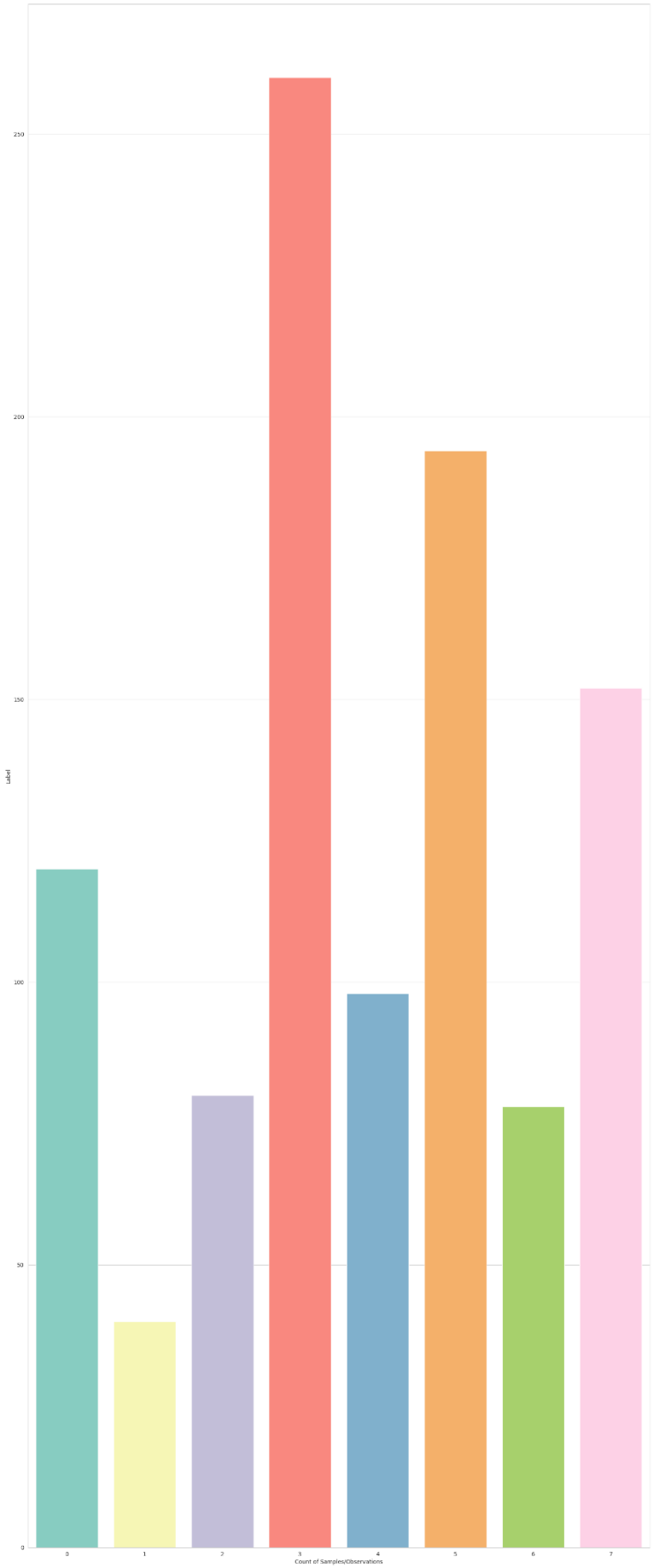
Stage 3: Deployment of Visual Predictive Model.

Stage 1: Exploratory Data Analysis for Visual Data

The first component of the Capstone Project was to perform an Exploratory Data Analysis (EDA). I initially thought to perform an EDA on my entire data as given, but in this process I ended neglecting my test data, leaving my EDA as an incomplete and inaccurate reflection of my dataset. Upon realizing this, I started the EDA process again, compressing my dataset into 8 classes (0-7) as my dataset in its given form has 58 (0-57) classes which proved to be hard to work with my available resources. Once I amended the dataset, I went ahead with answering my five chosen questions for my EDA:

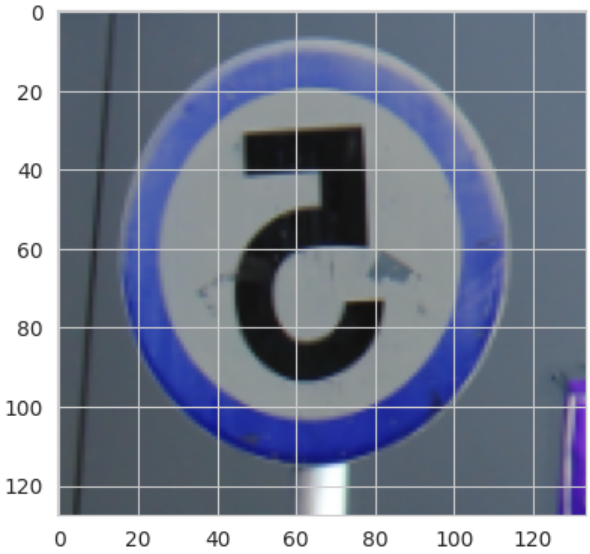
1. *What is the smallest class?*
2. *What is the biggest class?*

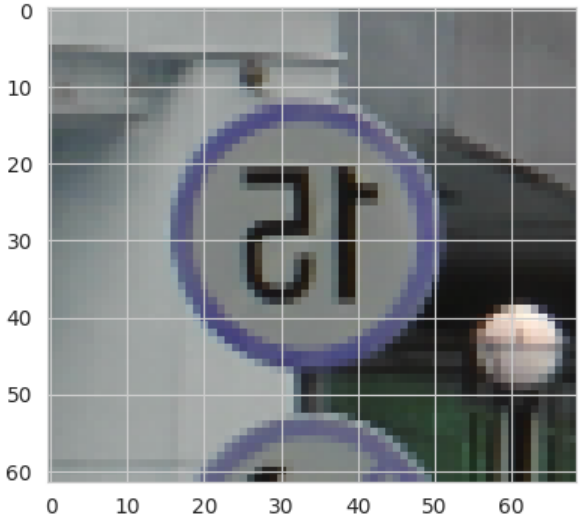
   From these two questions I wanted to understand the makeup of the dataset to better inform me in the later stages when doing. I was able to answer both these questions with the histogram I created, as seen below. I found that class 1 (“15” speed signs) was the smallest class 4 (“40” speed signs) was the largest class.


250
200
150
100
50
0
Label
0
1
2
3
4
5
6
7
Count of Samples/Observations

3. *How does the image look flipped?*

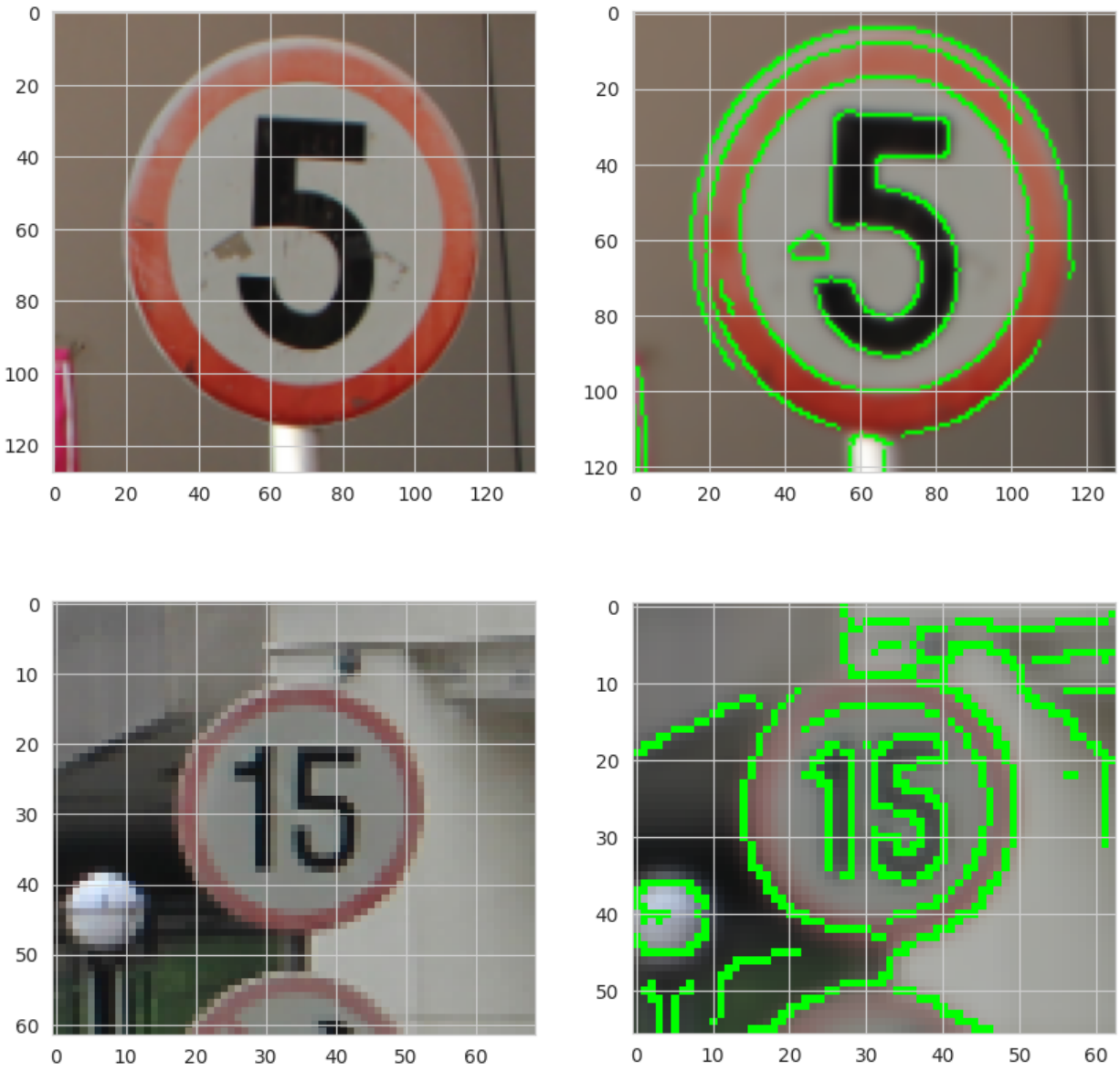
   I wanted to incorporate directional image manipulation to get a better understanding of the data and how a model would be able to augment images to better read them. I was able to achieve an answer through the EDA, as seen below.





4. How do the images look using canny edge detection?

   As most traffic signs bear many similarities, it can be hard for models to effectively identify them, so I wanted to investigate which features are



5. What does the image look like in grayscale
    - This will be helpful for the model to understand the images without any RGB color channels and I wanted to see what the differences are in the EDA.

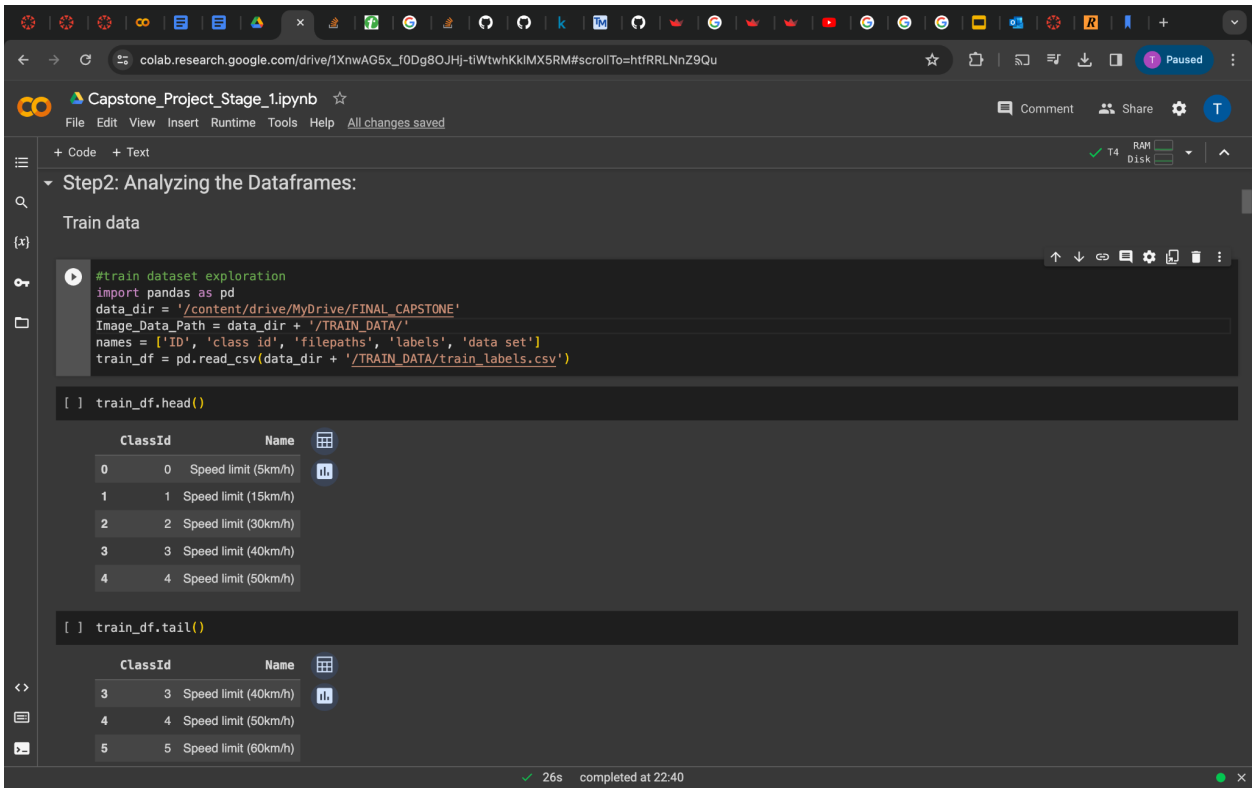
Original Image
"5" speed limit sign

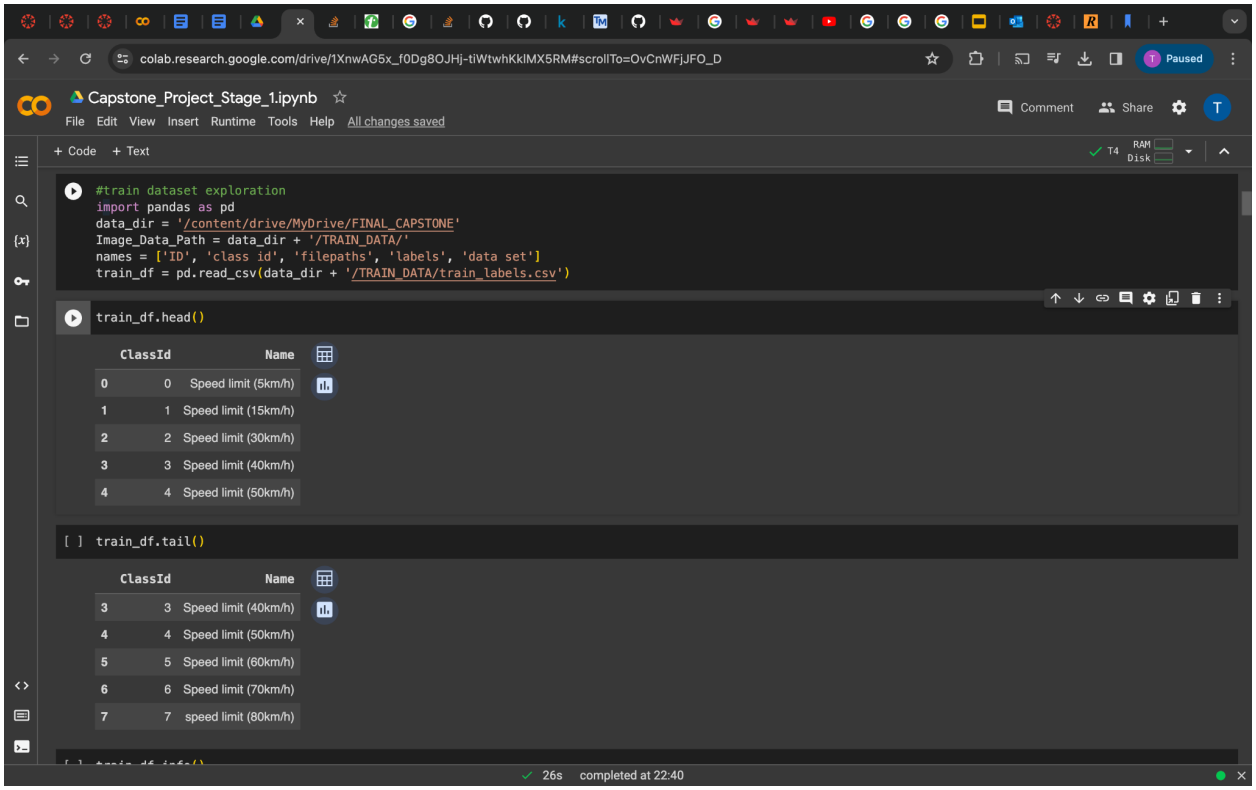
Gray scale image
"5" speed limit sign

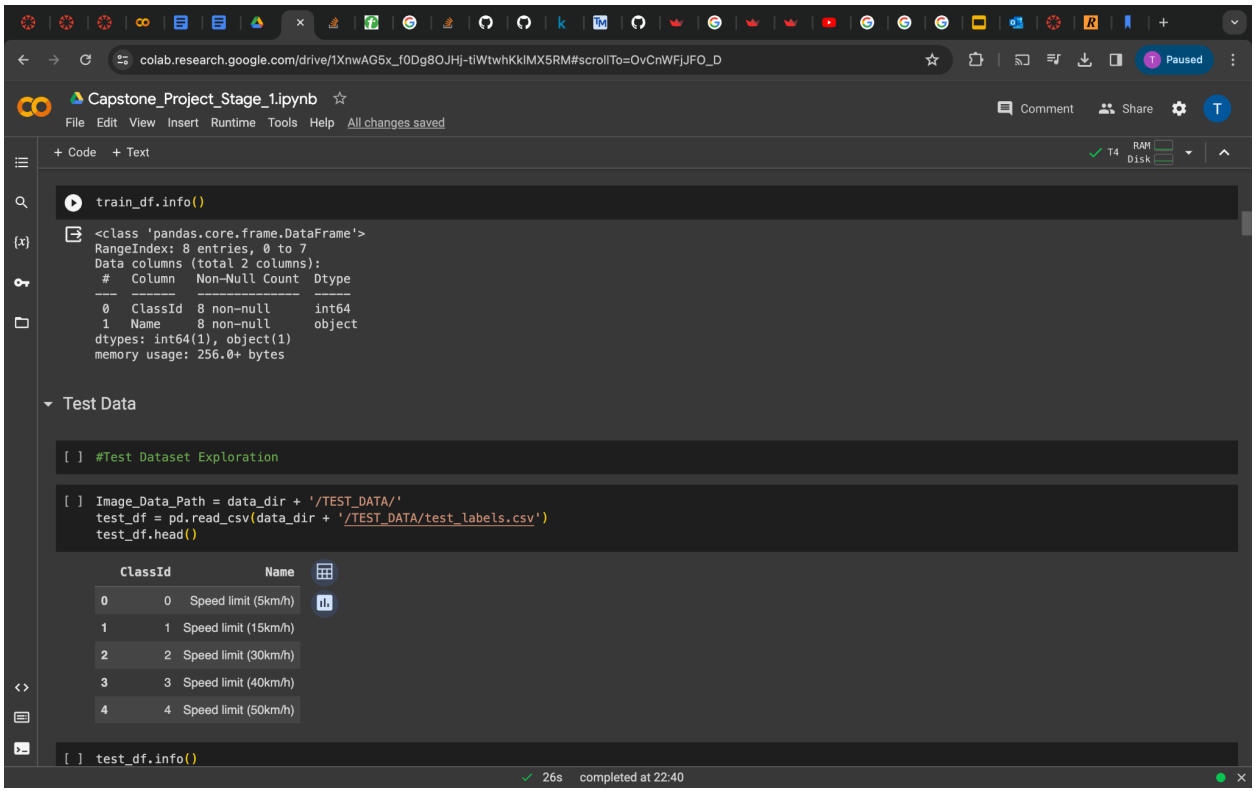
Original image
"15" speed limit sign

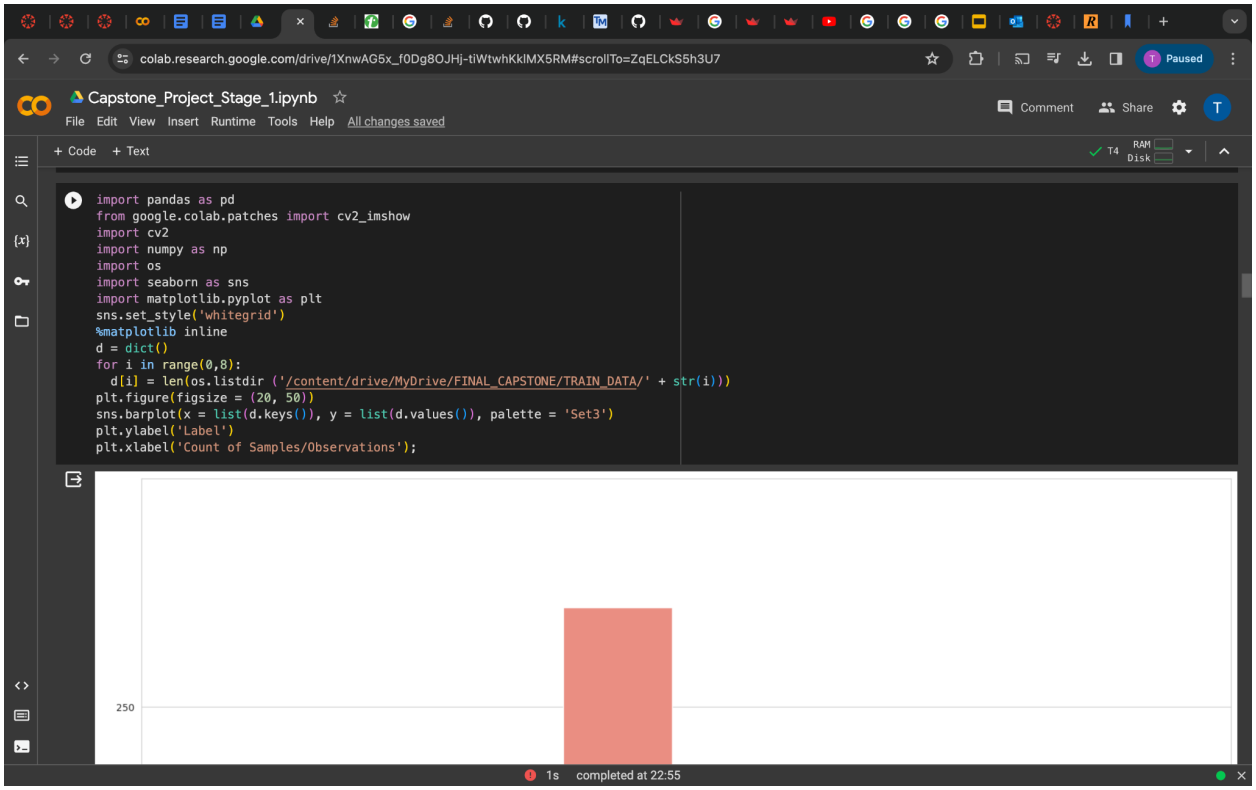
Original image
"15" speed limit sign

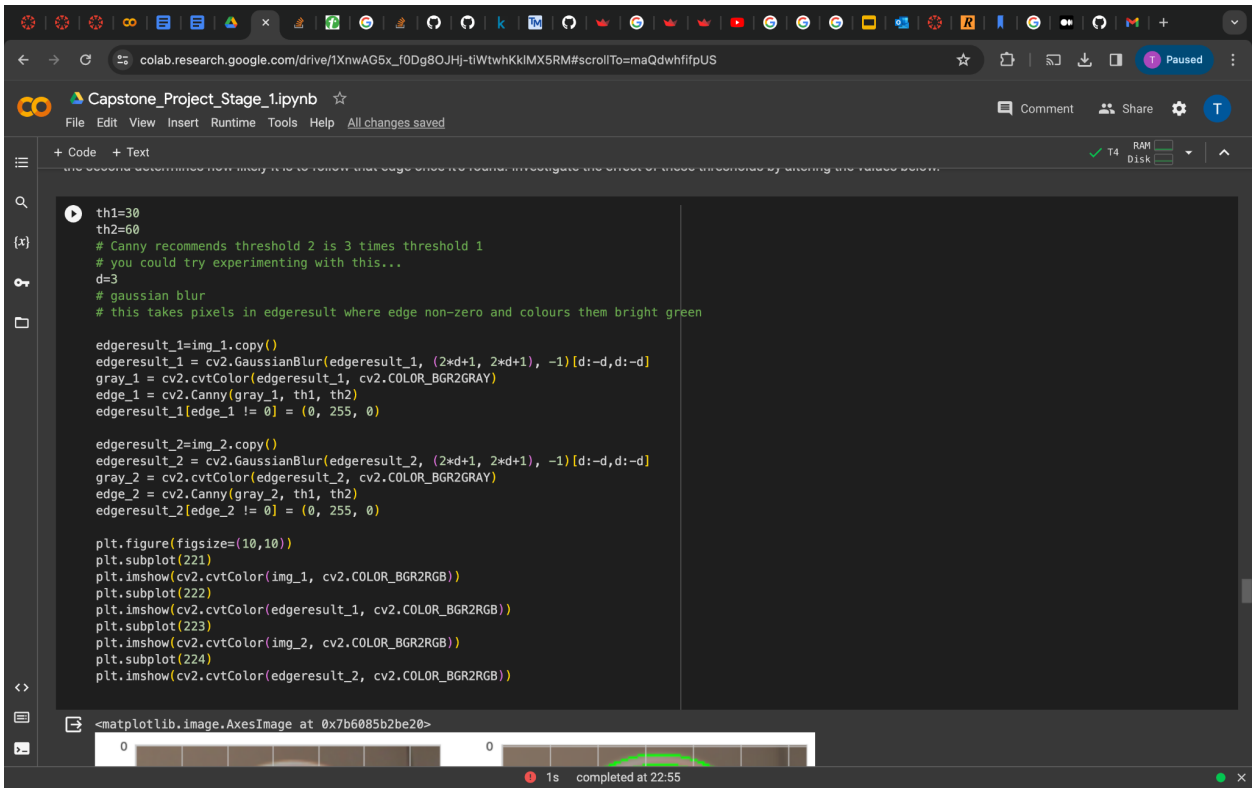
Below are supplementary frames from my EDA.




colab.research.google.com/drive/1XnwAG5x_f0Dg8OJHj-tiWtwhKkIMX5RM#scrollTo=OvCnWFjJFO_D
Paused
Capstone_Project_Stage_1.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
T4
RAM
Disk
#train dataset exploration
import pandas as pd
data_dir = '/content/drive/MyDrive/FINAL_CAPSTONE'
Image_Data_Path = data_dir + '/TRAIN_DATA/'
names = ['ID', 'class id', 'filepaths', 'labels', 'data set']
train_df = pd.read_csv(data_dir + '/TRAIN_DATA/train_labels.csv')
train_df.head()
ClassId Name
0 0 Speed limit (5km/h)
1 1 Speed limit (15km/h)
2 2 Speed limit (30km/h)
3 3 Speed limit (40km/h)
4 4 Speed limit (50km/h)
[ ] train_df.tail()
ClassId Name
3 3 Speed limit (40km/h)
4 4 Speed limit (50km/h)
5 5 Speed limit (60km/h)
6 6 Speed limit (70km/h)
7 7 speed limit (80km/h)
26s completed at 22:40


colab.research.google.com/drive/1XnwAG5x_f0Dg8OJHj-tiWtwhKkIMX5RM#scrollTo=OvCnWFjJFO_D
Paused
Capstone_Project_Stage_1.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
T4
RAM
Disk
train_df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8 entries, 0 to 7
Data columns (total 2 columns):
Column Non-Null Count Dtype
0 ClassId 8 non-null int64
1 Name 8 non-null object
dtypes: int64(1), object(1)
memory usage: 256.0+ bytes
Test Data
[ ] #Test Dataset Exploration
[ ] Image_Data_Path = data_dir + '/TEST_DATA/'
test_df = pd.read_csv(data_dir + '/TEST_DATA/test_labels.csv')
test_df.head()
ClassId Name
0 0 Speed limit (5km/h)
1 1 Speed limit (15km/h)
2 2 Speed limit (30km/h)
3 3 Speed limit (40km/h)
4 4 Speed limit (50km/h)
[ ] test_df.info()
26s completed at 22:40





Observations

I found that in the EDA stage, I was able to successfully execute all the required parts except for exploring the dataset through columns due to multiple syntax and file path errors which I was

not able to correct. Aside from that, I found the EDA to be quite insightful. I observed canny edge detection to be the most effective way to identify the nuances between the class, especially since the two classes I was using in the EDA were quite similar both having “5” in them.

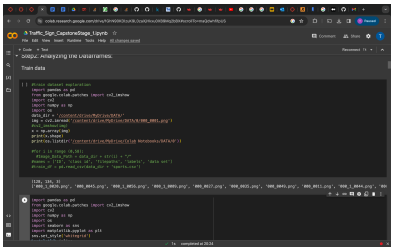
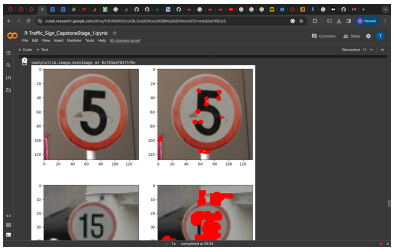
Stage 2: Predictive Model Development

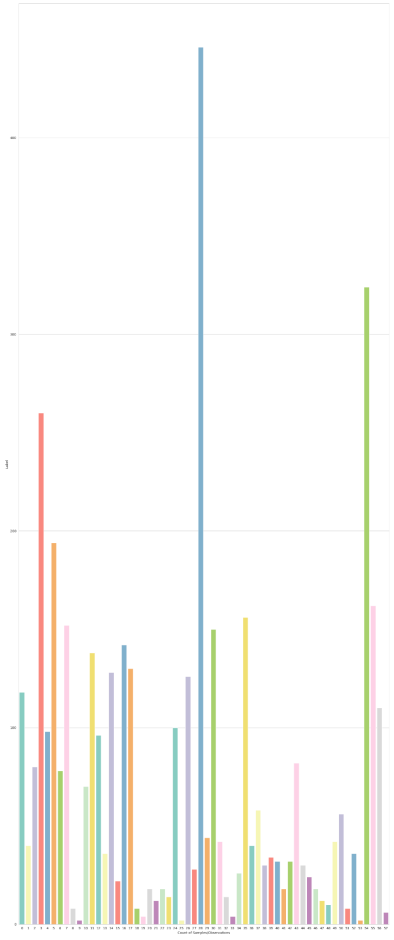
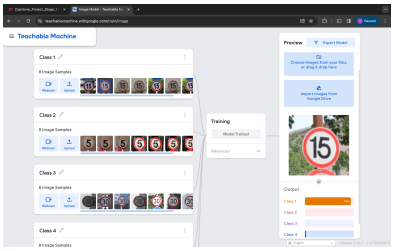
For the second component of the project, I used Teachable Machine to create the model. I used all 8 classes from my compressed dataset and shortened each class to its 40 feature photos. Teachable Machine was a great tool to use, as it streamlined the training process, allowing me to focus primarily on testing. For testing I tested each class and some results from the testing can be seen in the images below. I found I achieved accuracy rates of 80% - 98% between my 8 classes, through testing up to five times each to avoid overfittting. The model code can be accessed through Github via this link: https://github.com/googlecreativelab/teachablemachine-community/blob/master/snippets/markdown/image/tensorflow/keras.md

Conclusion and Reflection

The Capstone Project allowed me to take a momentary delve into the world of machine learning and how it relates to other aspects of coding such as the very libraries I had been using within my classes. Overall it was a learning experience, especially on the uses of Machine Learning models and what the process of creating and implementing them looks like.

Lab Journal:

Week	Entry	Excerpts/ Evidence
10	I did this lab at home, but I spent a lot of time in this week going over content to prepare and begin the EDA. as I began the first stage. I found the lectures to be a great help as I was previously facing a lot of confusion over where to start, but I was able to make a start on what essentially ended up as a draft EDA, seen in the next column.	
11	In this lab, we didn't go over the lab questions as they had already covered this week's portion of work in the previous work. This gave us the opportunity to work on the Capstone Project. I spent that time rewatching the week 9 and week 10 lectures to get a better understanding of the task requirements with the EDA. I continued following along with the lectures and the template	

12	The presentations started during this lab and as I watched the first set of people present (as mine was in the coming week), I found it was good to reaffirm that I am on the right track with my progress. I also spent some time working on my EDA. I experienced some trouble coding my histogram at first but I managed to get help from a peer.	
13	This was the week I presented my progress. At the time I was attempting to use my full dataset, which created a problem when moving onto my PDA. In initial form the dataset included 58 classes, so after the presentation I spent time starting again with a compressed version of my dataset - using 8 classes instead of 58.	

Challenge Questions:

Week 7: https://gist.github.com/tnyamuku/794ddcbacd8c3ceaecc9c8f1265df771

Bibliography

[1]“Traffic Sign Dataset - Classification,” *www.kaggle.com*. https://www.kaggle.com/datasets/ahemateja19bec1025/traffic-sign-dataset-classification (accessed Oct. 24, 2023).