FINAL ASSIGNMENT – DEEP LEARNING (CNN)

Project Background & Objective

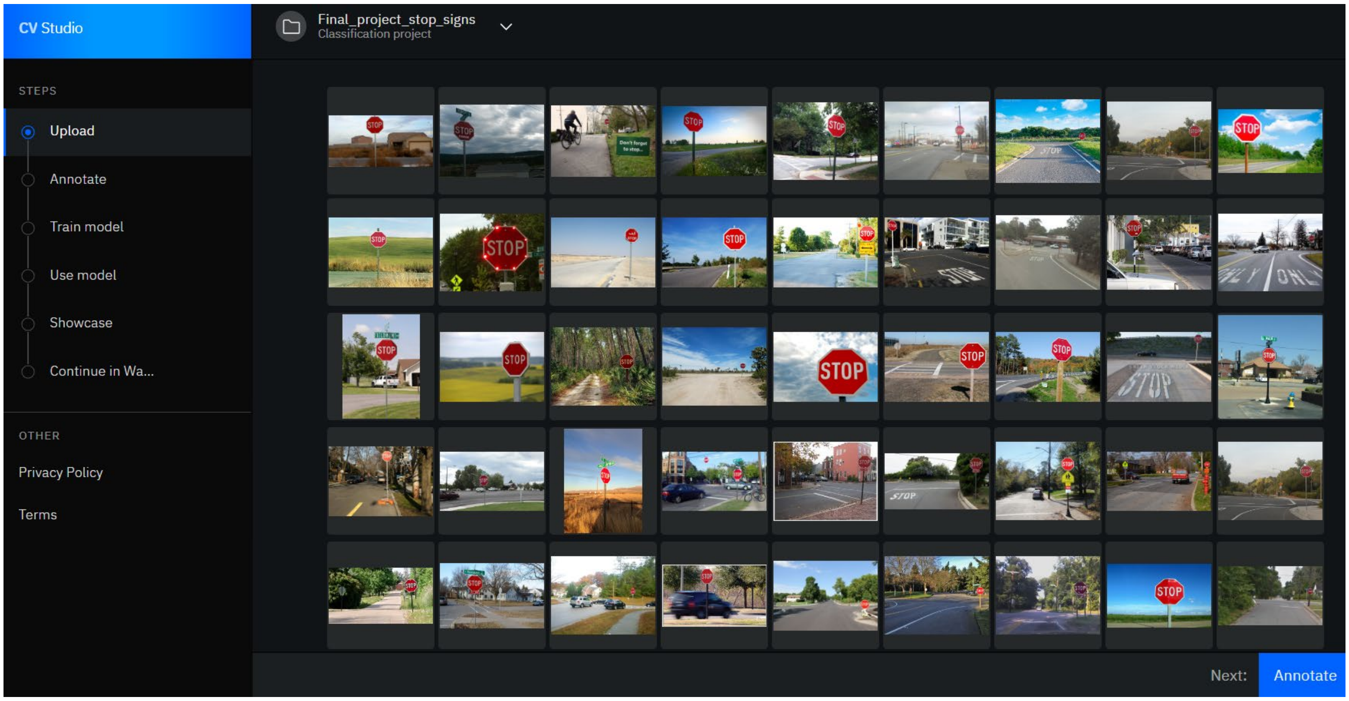
The goal of the project was to build a convolutional neural network to identify stop signs for self-driving cars.

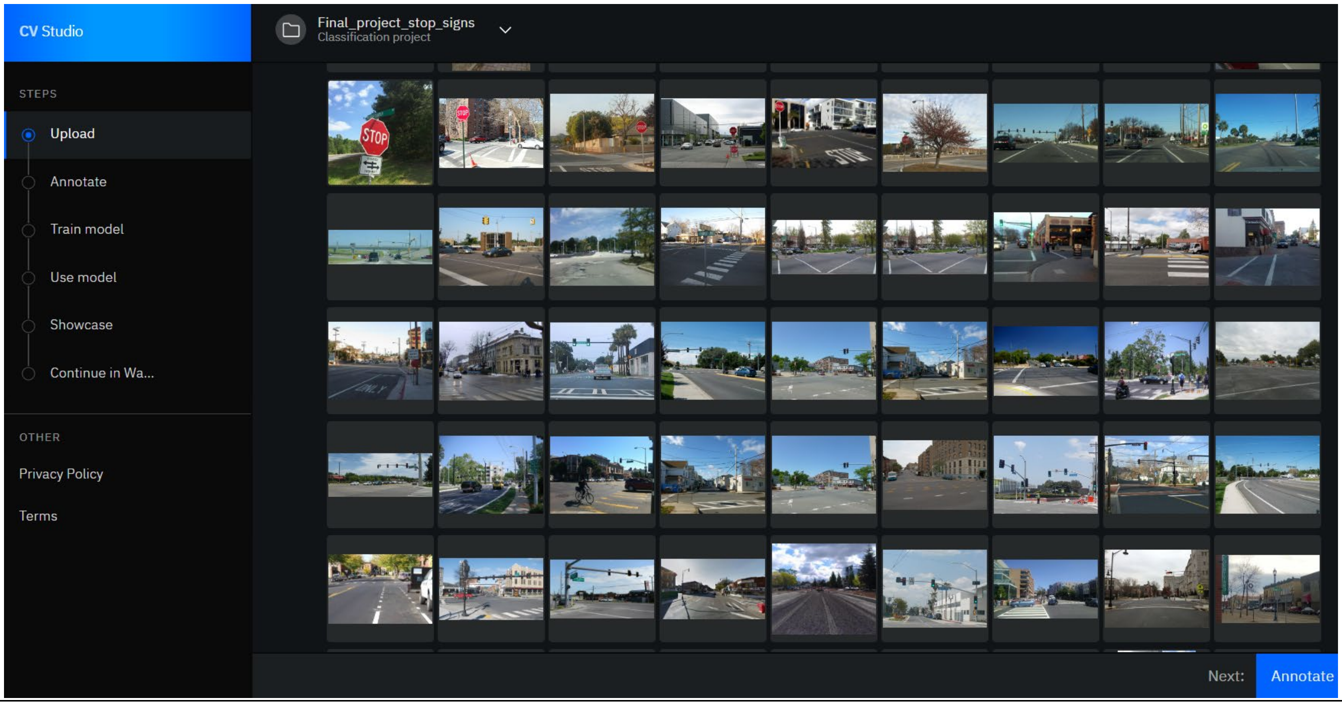
About the Dataset

The dataset consisted of several thousand examples of roads with stop signs and roads without stop signs. These were gathered from the internet and all of the stop signs were in English.

Exploratory Data Analysis, Pre-Processing & Model Training

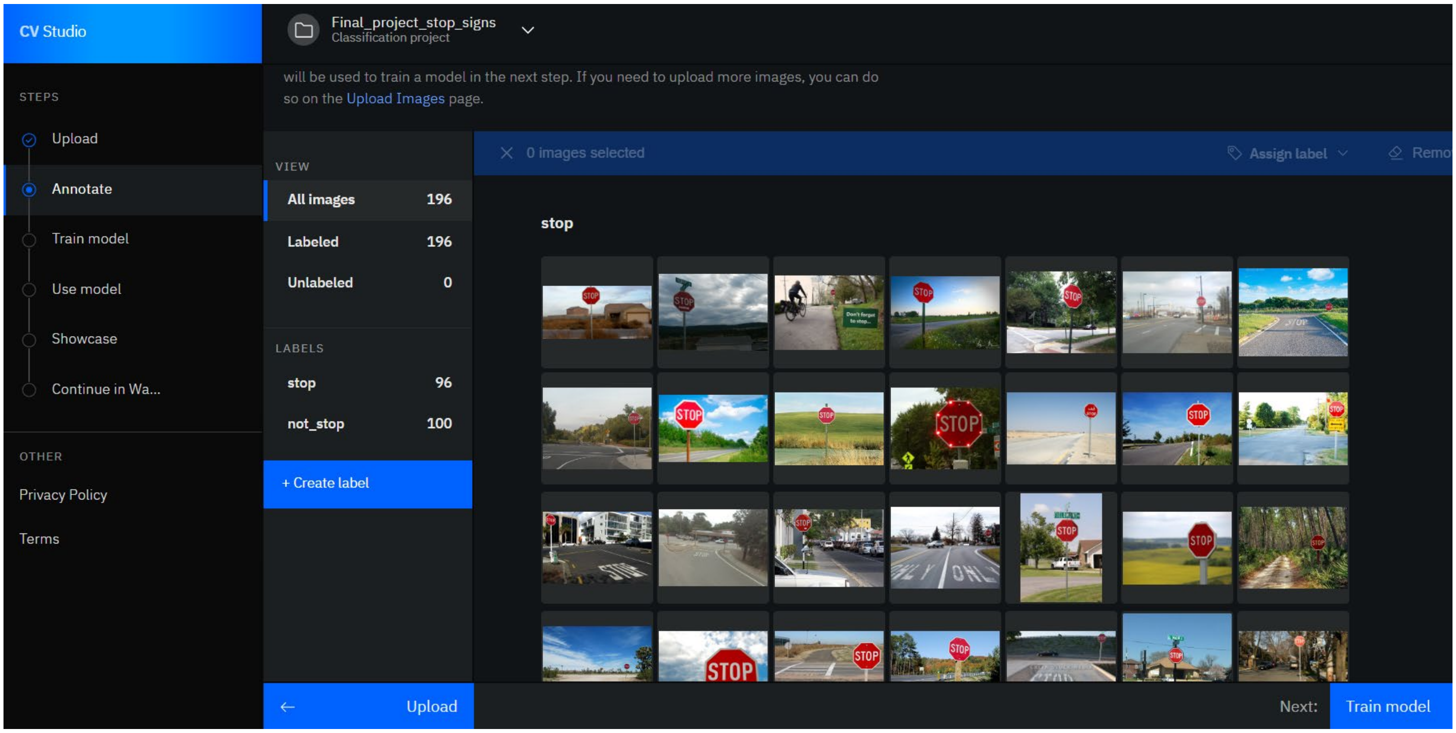
As the data was a basket of images there was not much analysis or pre-processing to do. In some instances, the photos needed to be cropped and rotated. Some images were rotated to make the model more robust although in the case of stop signs specifically, the cameras on the car should be reading them from the right side up (i.e. on its wheels).





Exploratory Data Analysis, Pre-Processing & Model Training

Photos were then labelled as “Stop” or “Not-Stop”



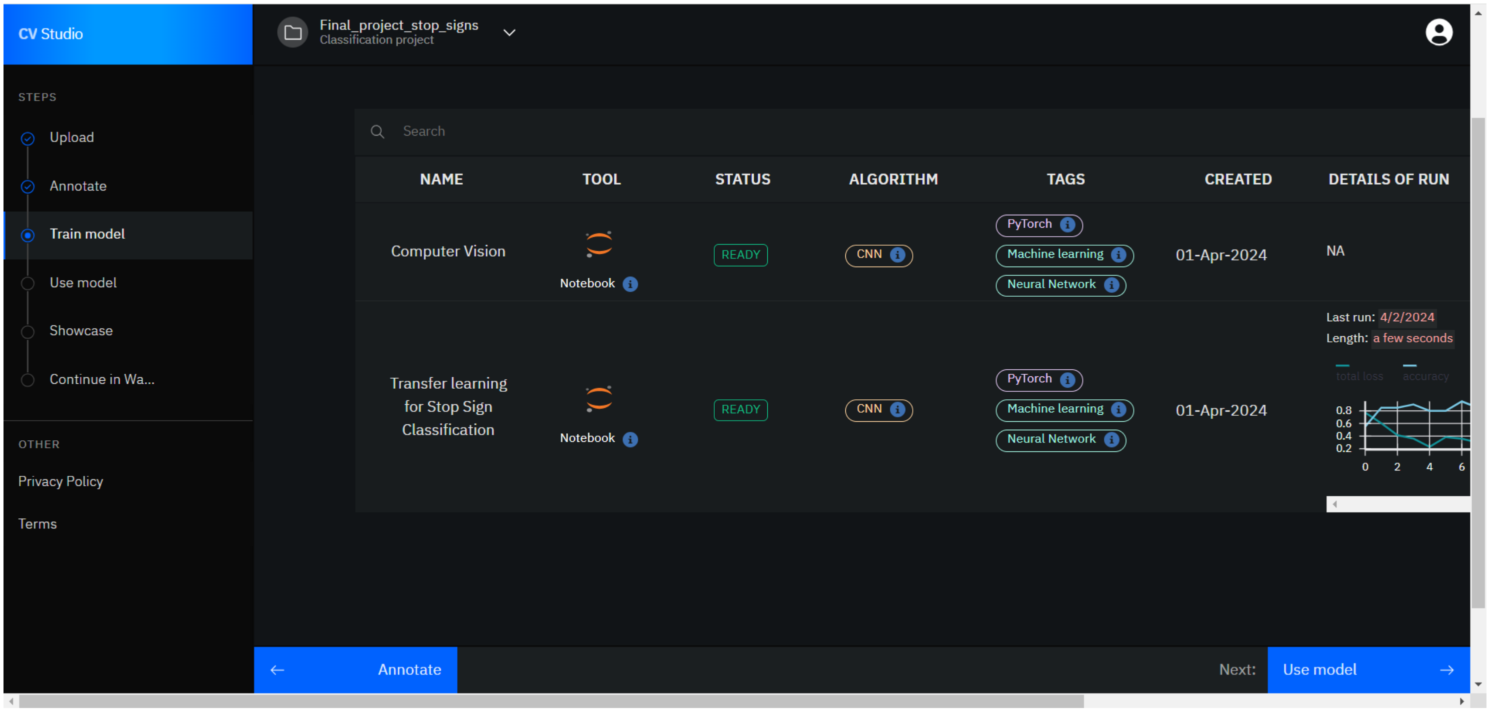
will be used to train a model in the next step. If you need to upload more images, you can do so on the Upload Images page.

VIEW	
All images	196
Labeled	196
Unlabeled	0

LABELS	
stop	96
not_stop	100

stop

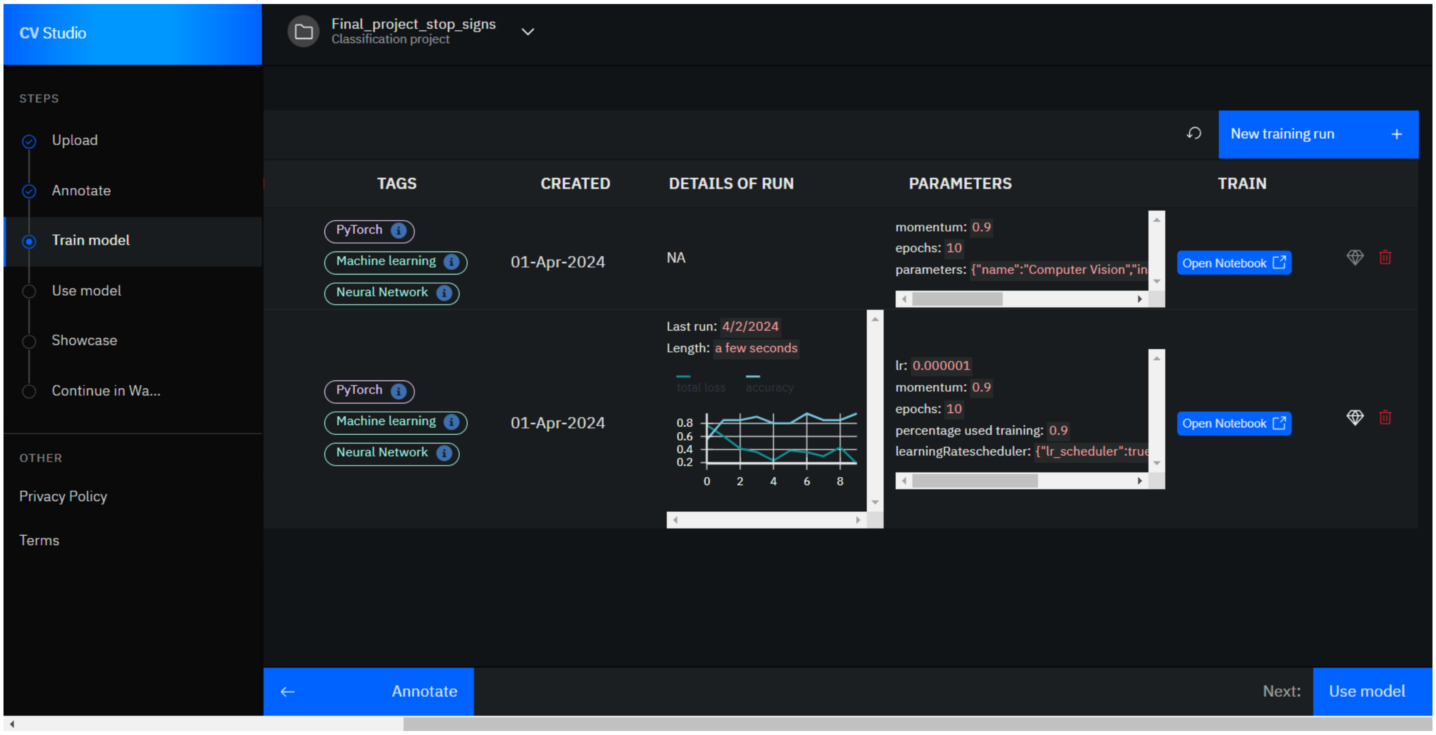
The model was then trained on a convolutional neural network in PyTorch The model utilized transfer learning whereby the high-level data set of stop signs was overlaid a deeper data set which had been pre-trained with more data and computational power on a broader image data set. This substantially reduced training time to about 20 minutes.



NAME	TOOL	STATUS	ALGORITHM	TAGS	CREATED	DETAILS OF RUN
Computer Vision	Notebook	READY	CNN	PyTorch Machine learning Neural Network	01-Apr-2024	NA
Transfer learning for Stop Sign Classification	Notebook	READY	CNN	PyTorch Machine learning Neural Network	01-Apr-2024	Last run: 4/2/2024 Length: a few seconds

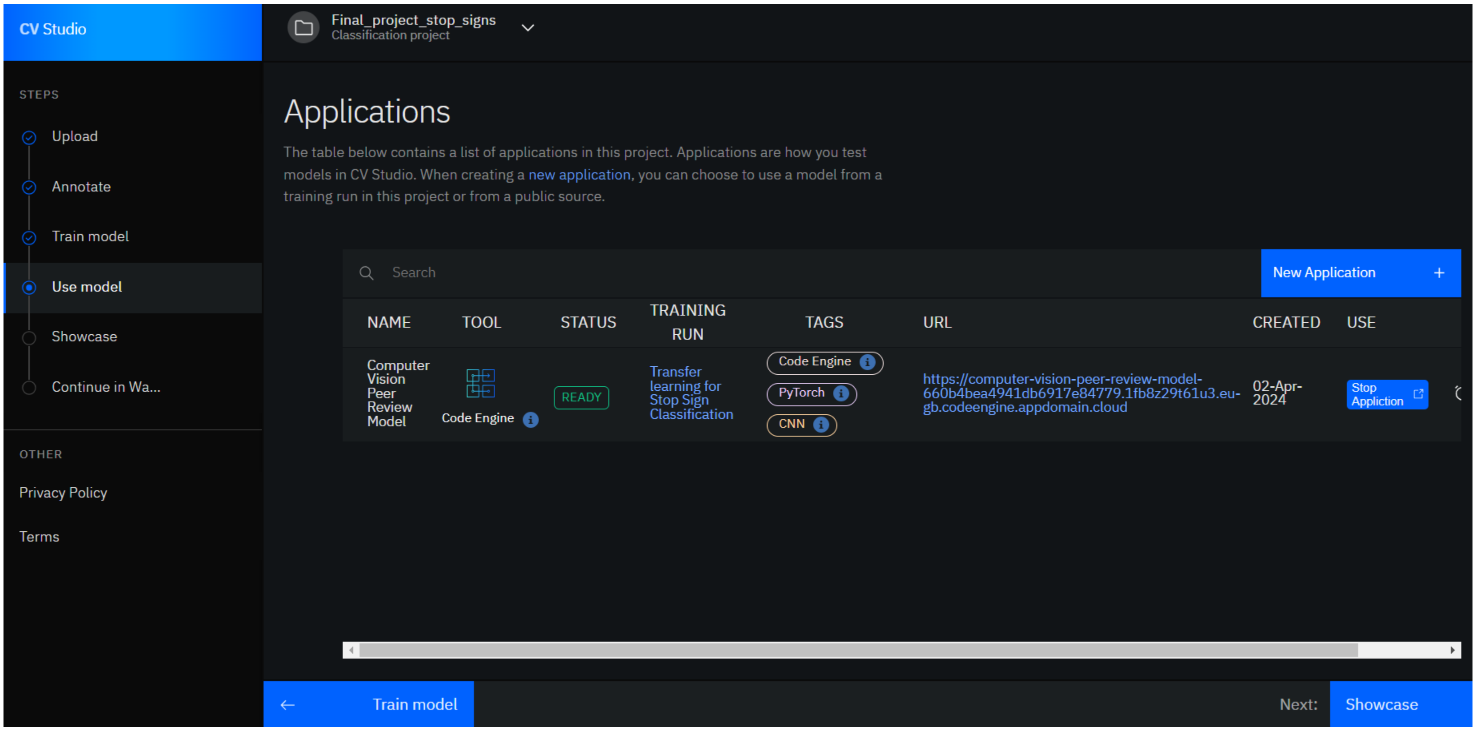
Exploratory Data Analysis, Pre-Processing & Model Training

I used a limited number of epochs (10) with a relatively small learning rate.



TAGS	CREATED	DETAILS OF RUN	PARAMETERS	TRAIN
PyTorch Machine learning Neural Network	01-Apr-2024	NA	momentum: 0.9 epochs: 10 parameters: {"name":"Computer Vision","in	Open Notebook
PyTorch Machine learning Neural Network	01-Apr-2024	Last run: 4/2/2024 Length: a few seconds	lr: 0.000001 momentum: 0.9 epochs: 10 percentage used training: 0.9 learningRatescheduler: {"lr_scheduler":true	Open Notebook

The model was then trained:



Applications

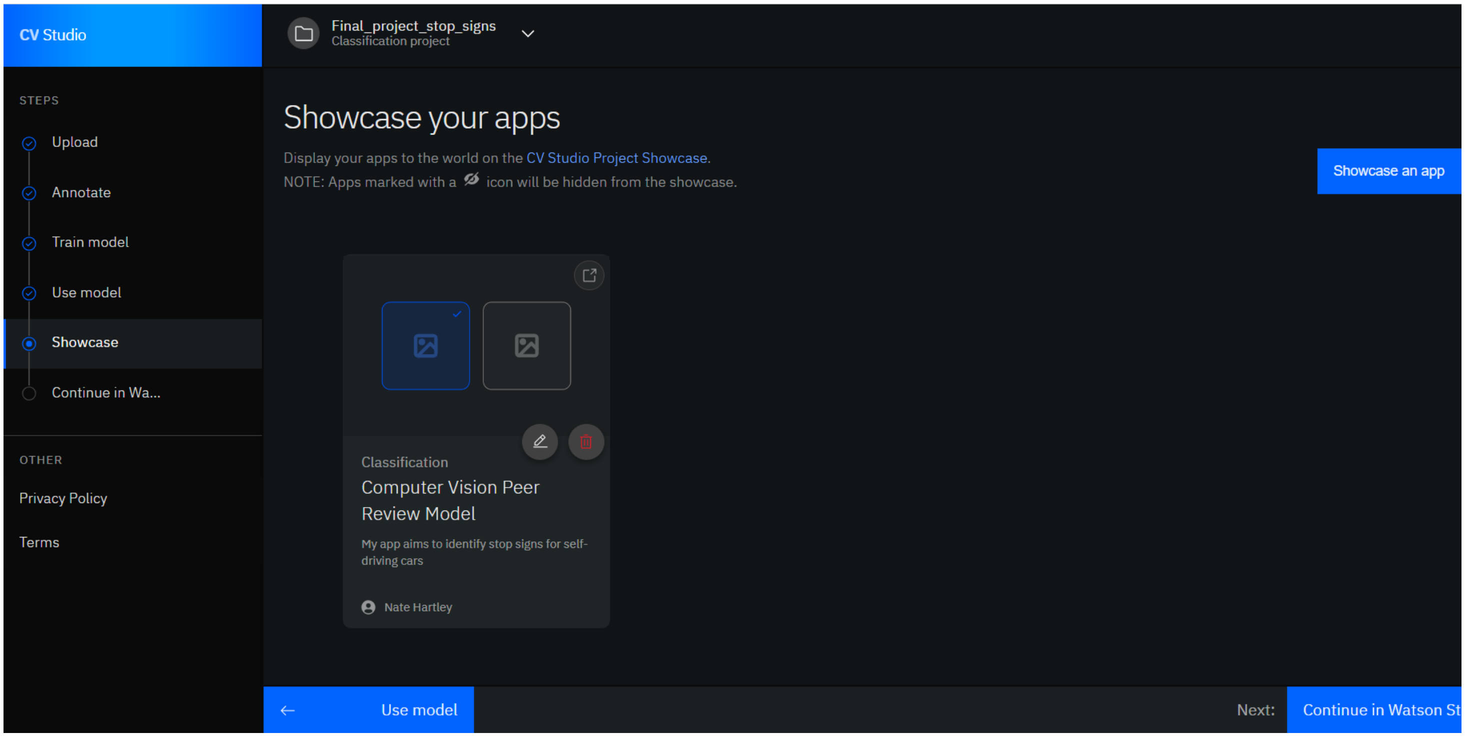
The table below contains a list of applications in this project. Applications are how you test models in CV Studio. When creating a new application, you can choose to use a model from a training run in this project or from a public source.

NAME	TOOL	STATUS	TRAINING RUN	TAGS	URL	CREATED	USE
Computer Vision Peer Review Model	Code Engine	READY	Transfer learning for Stop Sign Classification	Code Engine PyTorch CNN	https://computer-vision-peer-review-model-660b4bea4941db6917e84779.1fb8z29t61u3.eu-gb.codeengine.appdomain.cloud	02-Apr-2024	Stop Appliction

Exploratory Data Analysis, Pre-Processing & Model Training

The final app can be viewed here: https://computer-vision-peer-review-model-660b4bea4941db6917e84779.1fb8z29t61u3.eu-gb.codeengine.appdomain.cloud/

The source python file can be found here: https://github.com/NH-Davis/IBM-Deep-Learning-Final-Assignment



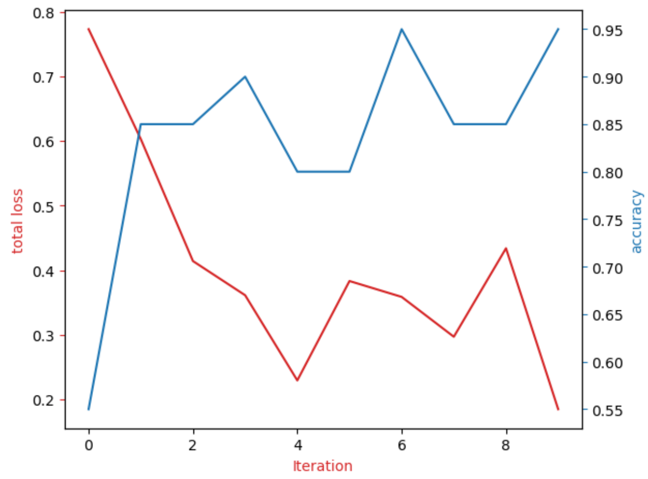
Showcase your apps

Display your apps to the world on the CV Studio Project Showcase.
NOTE: Apps marked with a icon will be hidden from the showcase.

Classification
Computer Vision Peer Review Model
My app aims to identify stop signs for self-driving cars
Nate Hartley

Model Evaluation

The model performed as expected, not entirely perfect but high accuracy – up to 95% after 8 iterations



Suggested Next Steps

Although the model displayed a high degree of accuracy I think it would be important to push this even higher – to 99% or even more given the human safety implications of misidentifying a traffic sign. A larger data set and more layers in the model could make this possible.