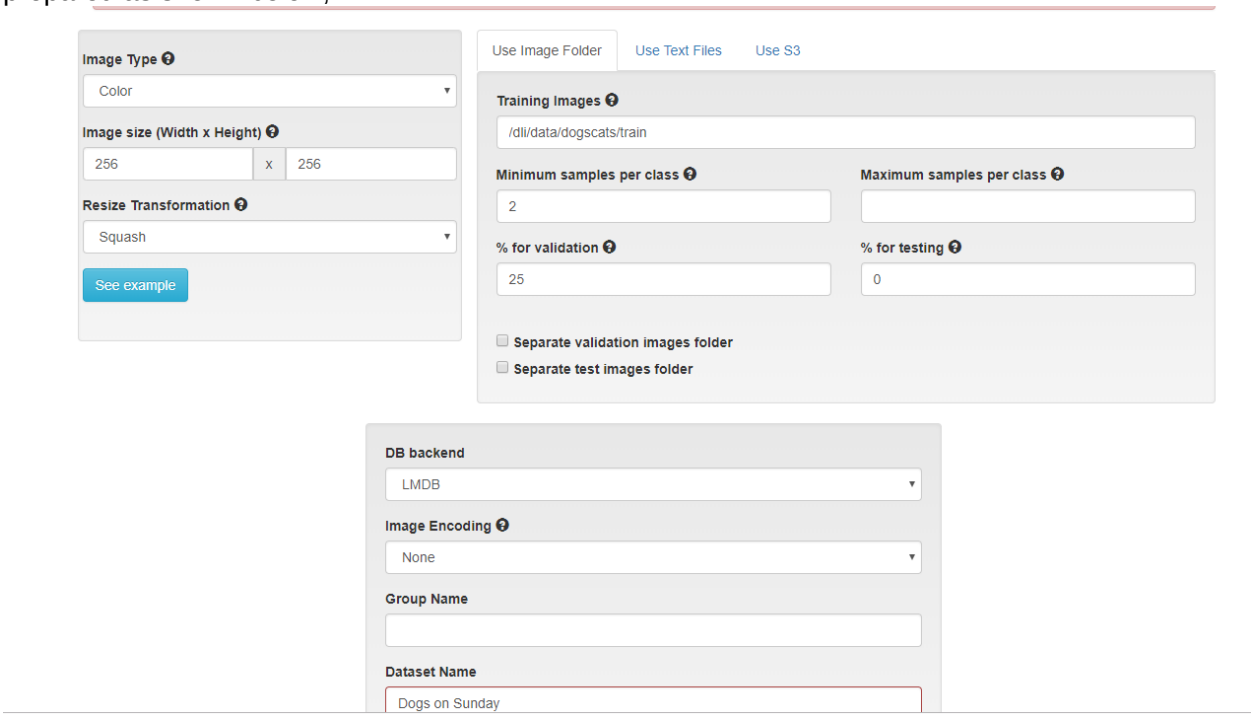
Persnoal NVIDIA Assignment: Byron Kiriibwa

Creating a model that is effective with *new data*

The goal of the first task was to train a neural network that could correctly classify dogs and cats that were not part of the training dataset. The dataset to be used for setting up the experiment was prepared as shown below;

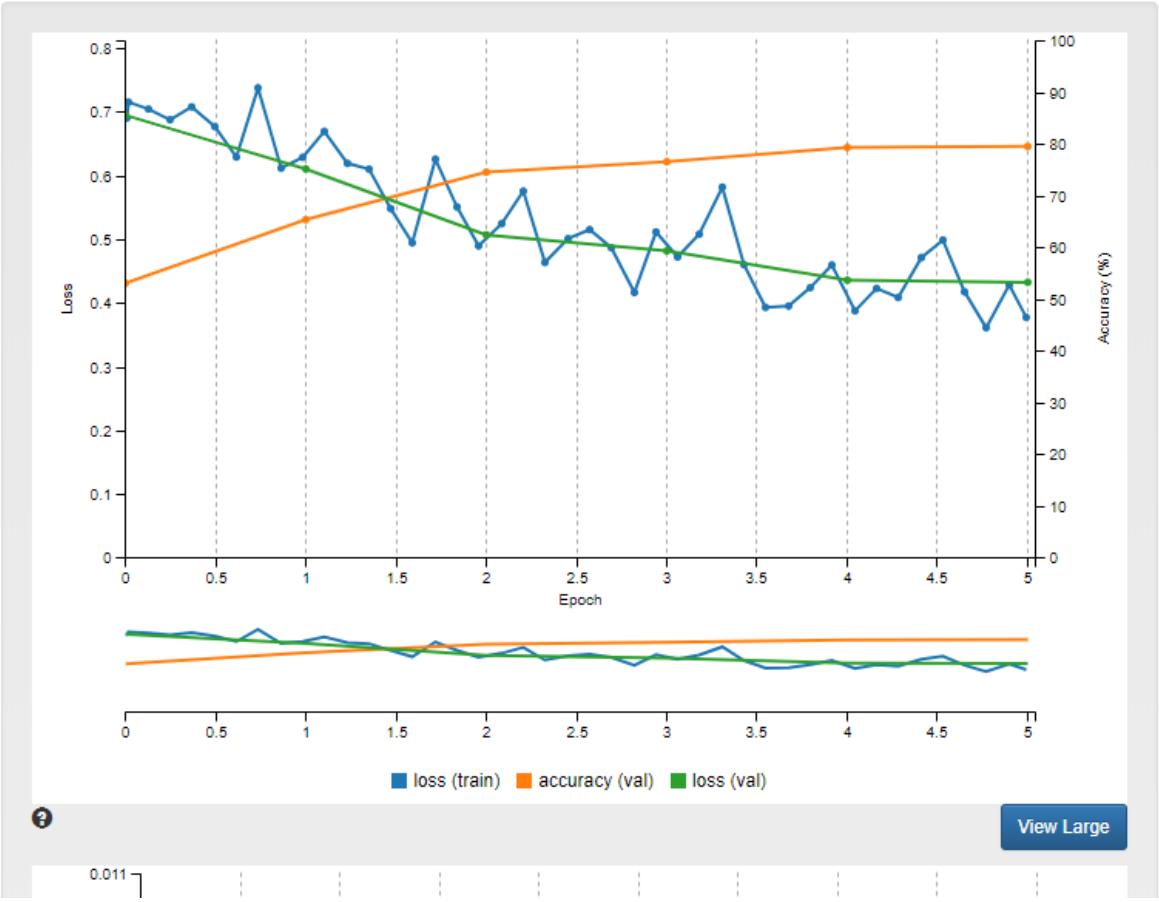


DIGITS to take the images to;

1. Standardized the images to the same size to match what the network we were going to train expects. We used AlexNet which expects 256*256 color images

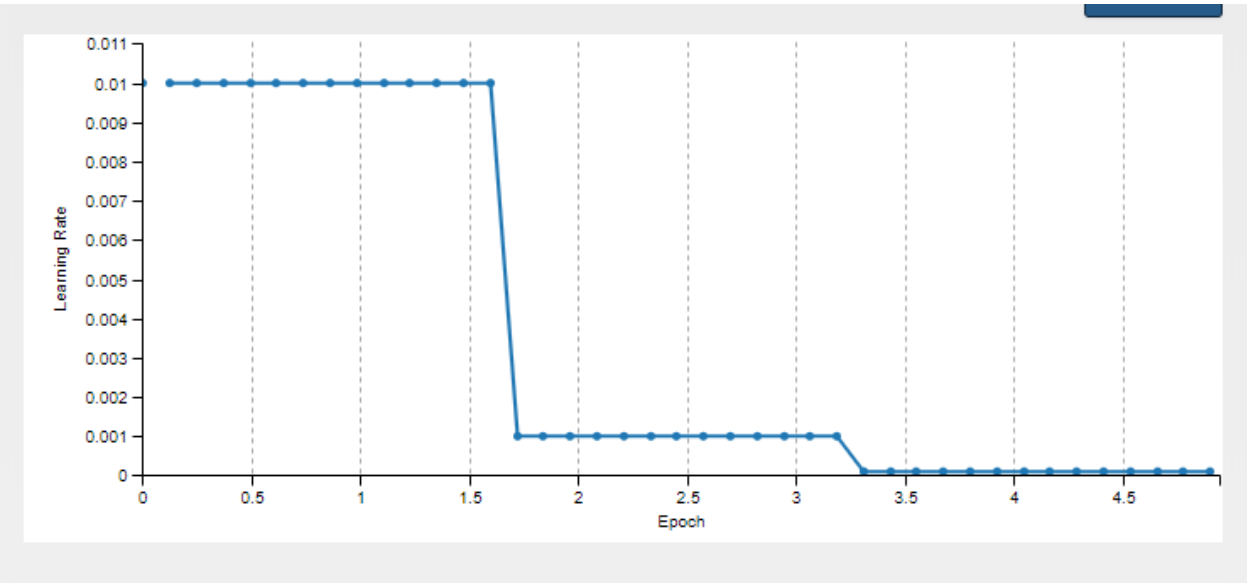
2. Split them into two datasets, where 75% of each class is used for training and 25% is set aside for validation.

After training a network of 5 epochs, the results that I got are shown below;

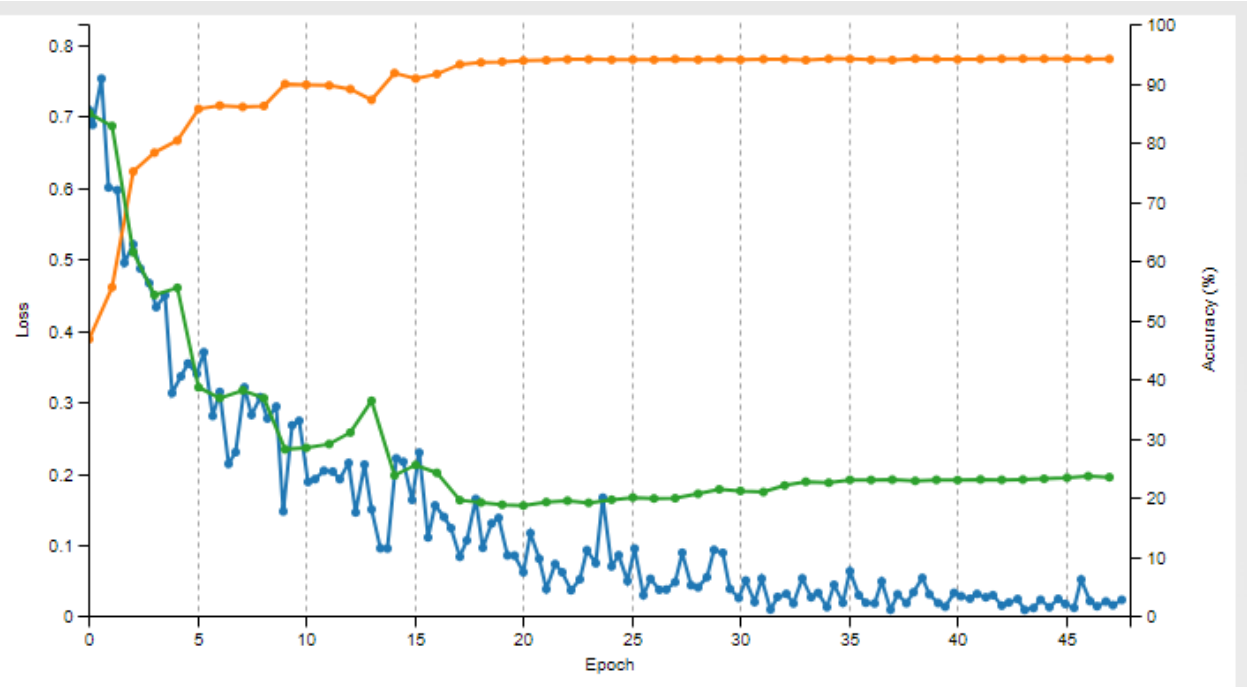


The accuracy (orange curve) kept on increasing with the number of epochs, while the training loss and validation loss also kept going down.

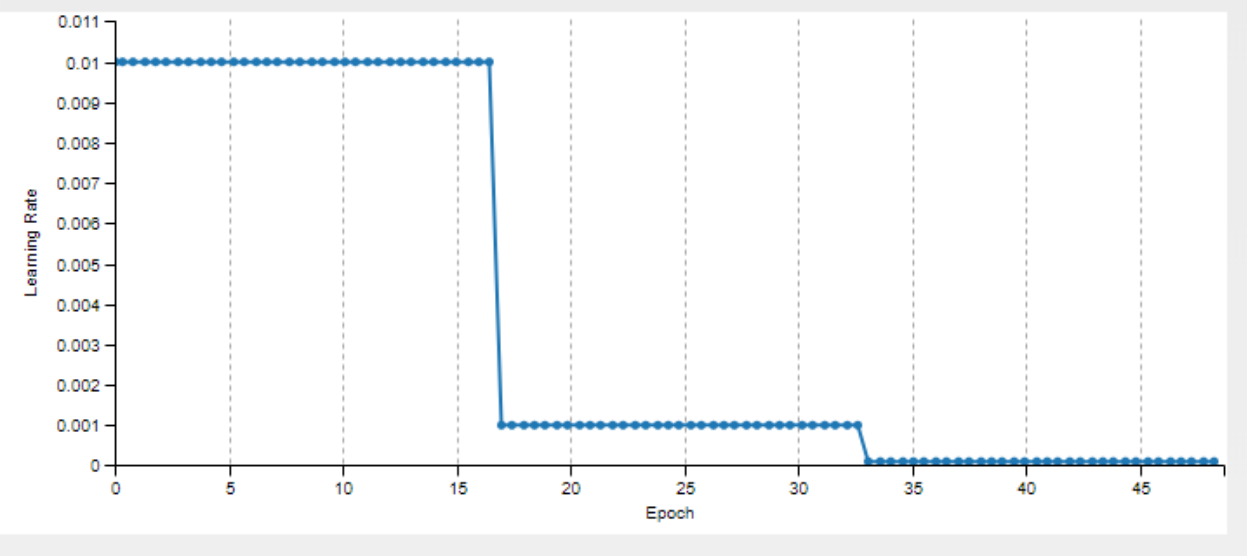
The learning rate also kept going down with the increasing number of epochs as shown below



When we increase the number of epochs to 50, the results change as shown below;



The accuracy of the network stopped increasing after the 15th epoch, the training loss stopped increasing after the 17th epoch, but the training loss kept increasing. This means that the network was overfitting on training data. The learning rate kept going down with the number of epochs.



When we tested the model on new data, it was able to classify louie as shown below;

After training the model, we deployed it in a simple application to be able to classify if we had a cat or dog. We did this by first preparing the dataset into the format that the model would accept, we prepared the testing dataset to have the same inputs and shape as we had for the training data.

We loaded the model weights and architecture to the application as shown below;

```
ARCHITECTURE = MODEL_JOB_DIR + '/' + 'deploy.prototxt'
WEIGHTS = MODEL_JOB_DIR + '/' + 'snapshot_iter_735.caffemodel'
print ("Filepath to Architecture = " + ARCHITECTURE)
print("Filepath to weights = "+ WEIGHTS)
```

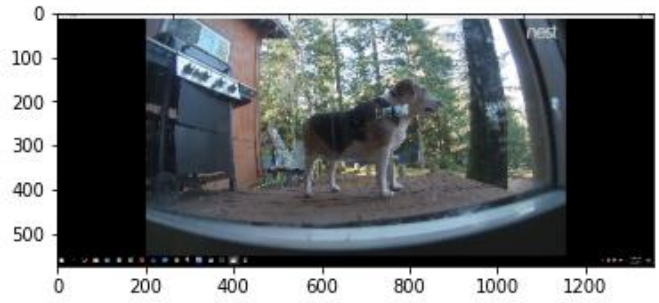
```
Filepath to Architecture = /dli/data/digits/20180301-185638-e918/deploy.prototxt
Filepath to weights = /dli/data/digits/20180301-185638-e918/snapshot_iter_735.caffemodel
```

In Task 4, we learnt to engage with some strategies for improving training performance using our existing model. We run more epochs on the existing model, we learnt how to search for better hyper parameters

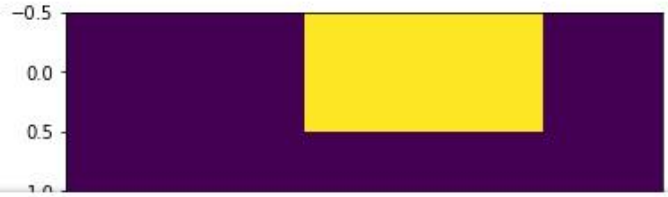
By increasing the amount of training data, tuning hyper parameters like learning rate and number of epoch, changing training time and the network architecture, we were able to improve the performance of the model to easily

Object Detection

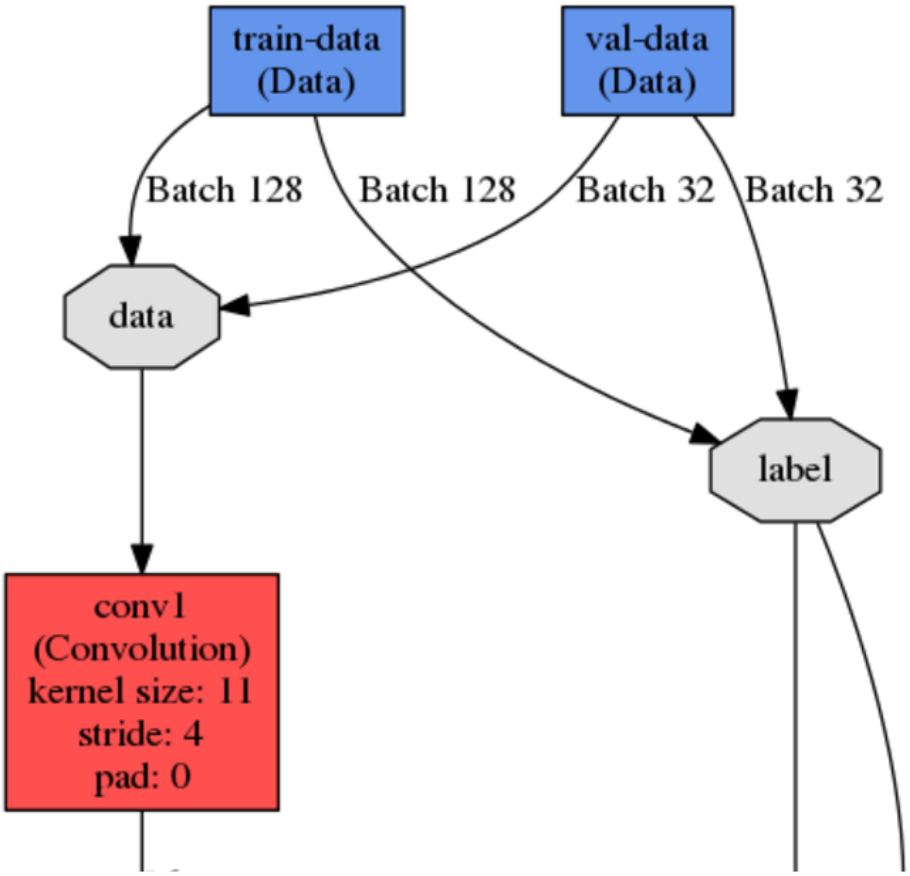
We carried out a series of experiments in an effort to detect objects using deep learning techniques. We used the sliding window approach to determine where the dog was in the image as shown below;




```
Total inference time: 0.824322223663 seconds
```



The image above shows how a classifier was able to figure out where exactly the dog was in the picture.



We changed the set up of the model by adding more layers and configuring the layers below and above the added layers to be able to take in the new inputs.

Lastly we worked on an assessment test and got Deep Learning basics certified.