YANGON TECHNOLOGICAL UNIVERSITY

Department of Mechatronics Engineering

Research Progress Report

Name: Aw Thura

Roll no: VI McE-11

Supervisor: Dr. Phyu Phyu Htun

Submitted on: 15th June, 2020

Introduction

According to the timeline previously proposed, the research is currently focused on training and testing the objection detection model. This report will focus on the experiments made with the customized YOLO v4 detection model and its results.

Previous Works

The research papers published on previous YOLO models are reviewed and the state of the art YOLO v4 model is selected for this research. Studying deep learning courses and tutorials for several months also aided in strengthening the background knowledge needed to understand the whole process.

Overview of Experiments

Detection models on “Car” and “Person” classes are separately trained for the first experiment. The second experiment combined these two classes and the testing results are compared with the former. Images from Google’s OPEN IMAGES DATASET is used for both experiments

Experiment I: Separate Training of Detection Models

“Car” and “Person” detection models are separately trained as single class detectors. For each training, 2000 labelled images from OPEN IMAGES DATASET are downloaded. The models are trained for 2000 batches and have final loss values of 2.6 for cars and 2.8 for persons. The results are shown in the figures below.

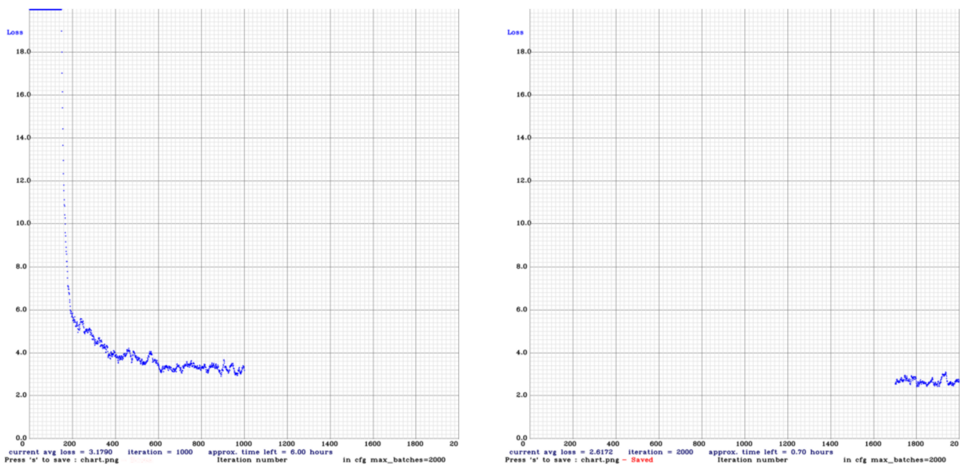

Fig 1: Loss Graphs of car detector after training 1000 & 2000 batches

Fig 2: Testing single class car detector (trained 1000 batches)

Fig 3: Testing single class car detector (trained 2000 batches)

When training the single-class car detector, the training is intentionally stopped at 1000 batches to save the weights and explore the results. Then the model training is continued with the saved weights until 2000 batches. From the graphs in Figure 1, it can be concluded that a loss value if 3.18 is achieved after training 1000 batches. This value is further reduced to 2.62 after 2000 batches. The results of this difference can be seen in Figures 2 and 3. In figure 3, the detector is sharper and able to detect the cars on the bridge and those further away from the camera.

```
[yolo] params: iou loss: ciou (4), iou_norm: 0.07, cls_norm: 1.00, scale_x_y: 1.05
nms_kind: greedynms (1), beta = 0.600000
Total BFLOPS 69.079
avg_outputs = 568027
 Allocate additional workspace_size = 52.43 MB
Loading weights from /mydrive/yolov4_car/backup/yolov4_custom_1000.weights...
 seen 64, trained: 64 K-images (1 Kilo-batches_64)
Done! Loaded 162 layers from weights-file
/mydrive/images/highway.jpg: Predicted in 21.231000 milli-seconds.
Car: 49%
Car: 78%
Car: 70%
Car: 43%
Car: 50%
Car: 50%
Unable to init server: Could not connect: Connection refused
```

Fig 4(a): Detection time and confidence levels (1000 batches)

```
[yolo] params: iou loss: ciou (4), iou_norm: 0.07, cls_norm: 1.00, scale_x_y: 1.05
nms_kind: greedynms (1), beta = 0.600000
Total BFLOPS 69.079
avg_outputs = 568027
 Allocate additional workspace_size = 3.61 MB
Loading weights from /mydrive/yolov4_car/backup/yolov4_custom_final.weights...
 seen 64, trained: 128 K-images (2 Kilo-batches_64)
Done! Loaded 162 layers from weights-file
/mydrive/images/highway.jpg: Predicted in 98.282000 milli-seconds.
Car: 65%
Car: 81%
Car: 97%
Car: 48%
Car: 41%
Car: 48%
Car: 51%
Car: 95%
Car: 90%
Car: 61%
Car: 89%
Car: 52%
```

Fig 4(b): Detection time and confidence levels (2000 batches)

Figure 4 compares the detection time and confidence levels of 1000 batch and 2000 batch trained weights. It can be observed that the former has a small advantage on prediction time but it is negligible as the latter has better accuracy, detection rate and confidence levels in a comparatively short amount of time.

Single class person-detector is also trained with the same settings. This time, the model is trained 2000 batches directly as it proved to give the better results. These results can be seen in the figures below.

Fig 5: Testing single class person detector with a video demo

Fig 6: Testing single class person detector on a relatively crowded image

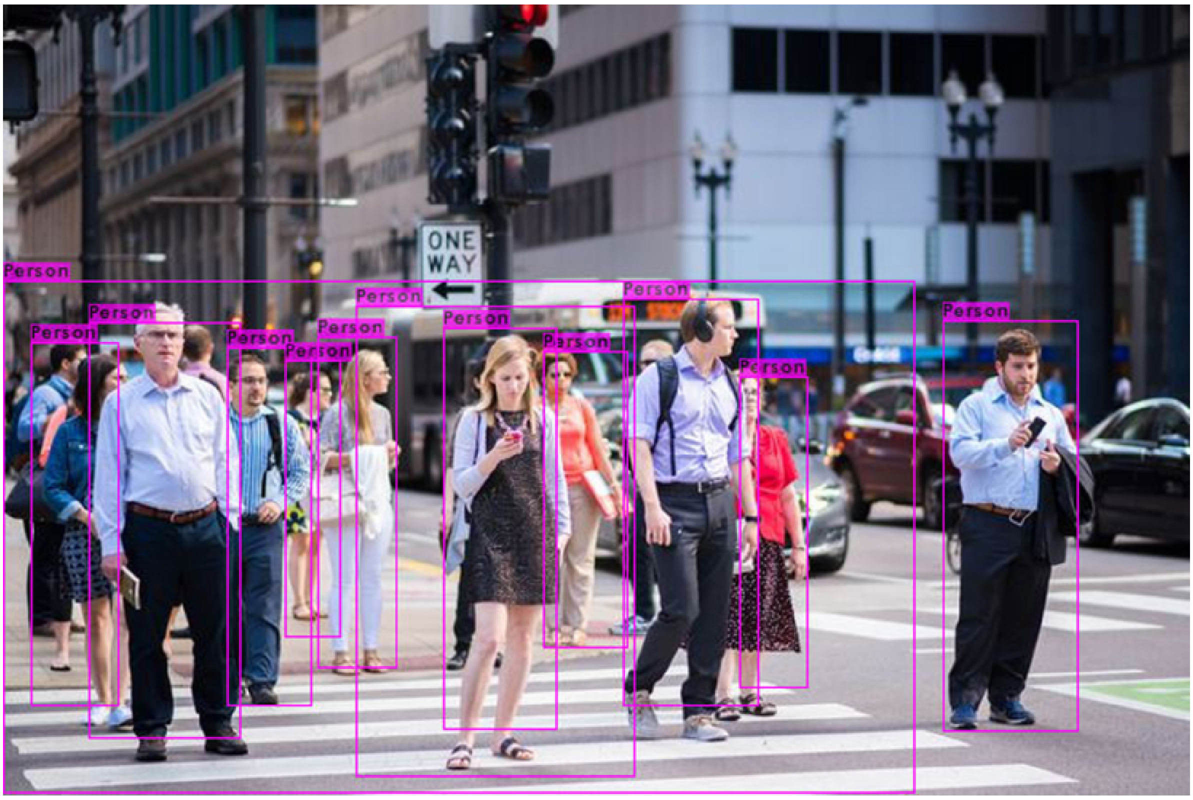

Fig 7: Testing single class person detector on a crowded crossing

The detector is run on several videos and images. In figure 6, it can be observed that the person behind the pole is left undetected and in figure 7, overlapping detections can be seen including a big bounding box covering a bunch of people.

Conclusion

To solve the problems of these custom-trained single class detectors, the number of training batches can be increased to further reduce the loss value and improve the accuracy. More occluded and overlapped input data can also be added so that the model can handle occlusions in the case of figure 6 and overlappings in figure 7.

Experiment II: Training of Multiclass Classifier

“Car” and “Person” classes are trained together as a multiclass classifier. 5000 labelled images, 3000 for “Car” class and 2000 for “Person” class, from OPEN IMAGES dataset is used for the training. For testing, all the images and videos are randomly downloaded from the internet. The classifier is trained with the initial setting of 4000 batches and achieved a loss value of 3.6 which can still be reduced by further training.

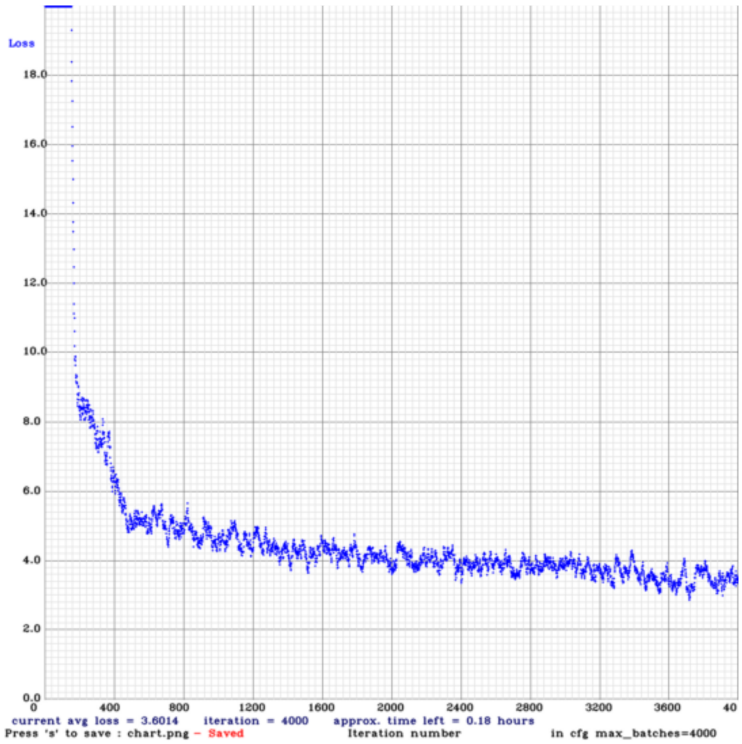

Fig 8: Loss graph of Multiclass classifier (trained 4000 batches)

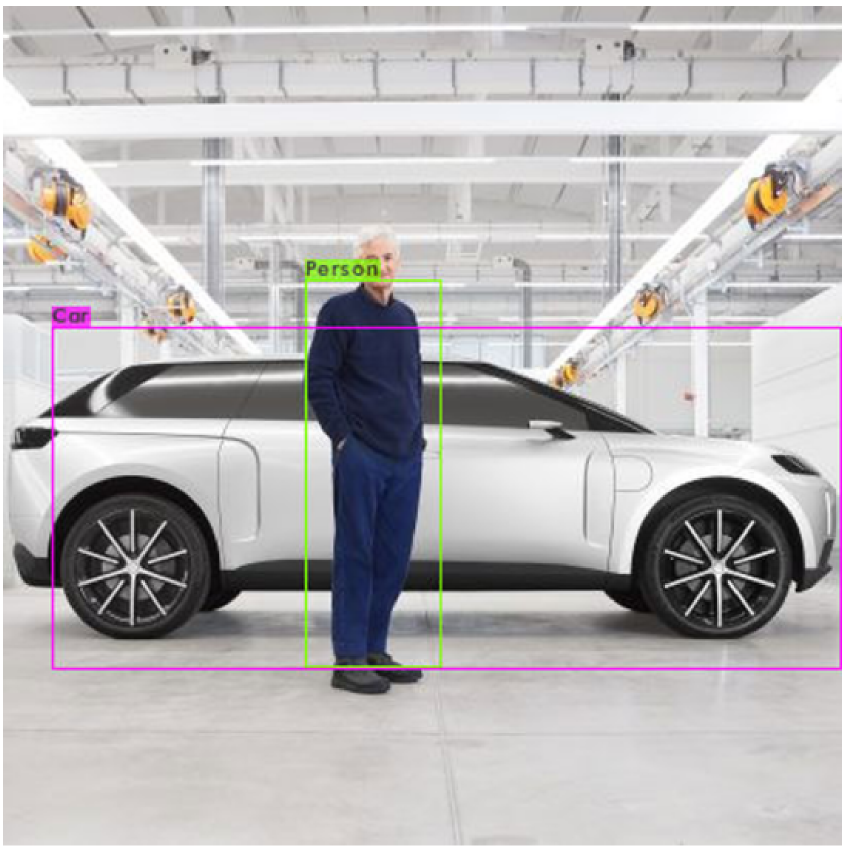

Fig 9: Testing multiclass classifier on a clean image

Fig 10: Testing multiclass classifier on a relatively crowded image

Fig 11: Testing multiclass classifier with a video demo of pedestrians

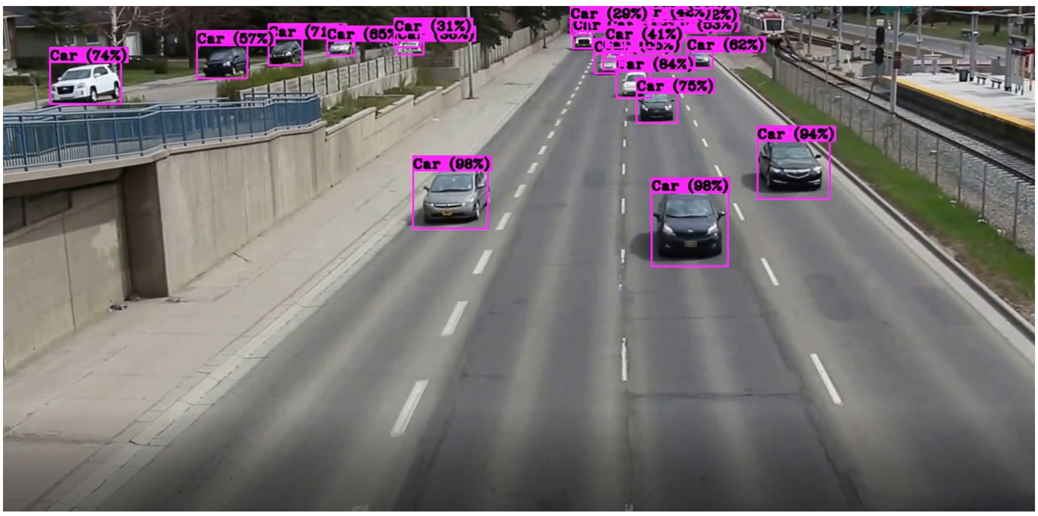

Fig 12: Testing multiclass classifier with a video demo of cars

Fig 13: Testing multiclass classifier with a video demo of a street camera

The classifier seems to be working fine on the above tests. In figure 10, it was able to detect the occluded person which the single class detector couldn't. But this classifier still has some problems. To highlight the problem, several images and demo screenshots are shown below.

Fig 14(a)(b)(c): Testing multiclass classifier with a video demo of car blackbox camera

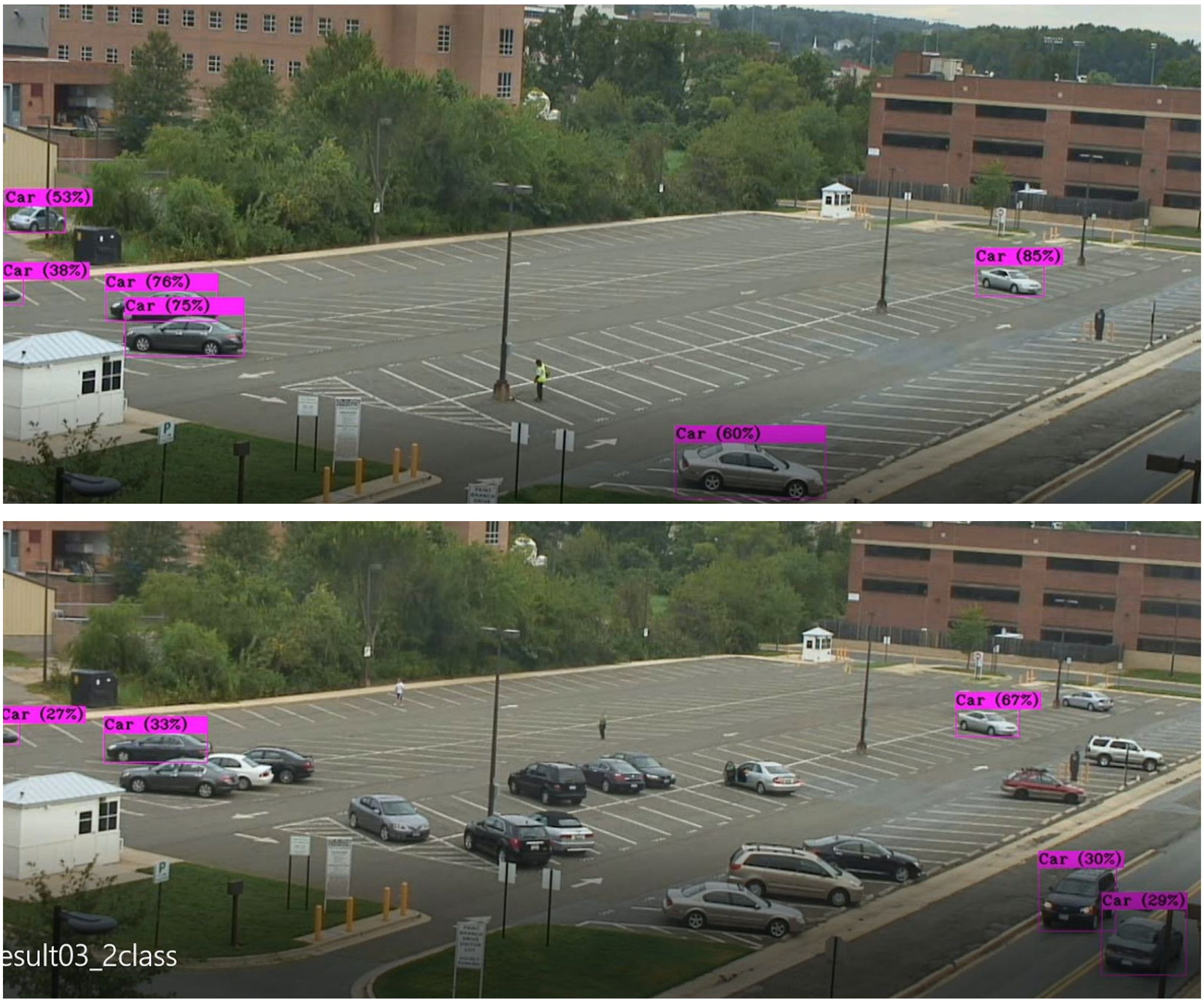

Fig 15(a)(b): Testing multiclass classifier with video demos of campus surveillance camera

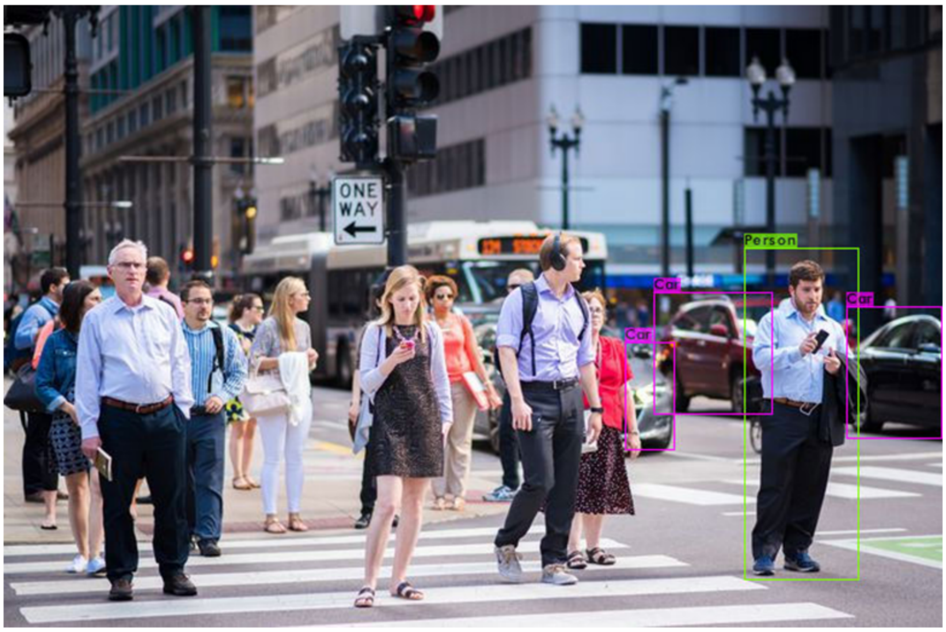

Fig 16: Multiclass classifier struggles to detect people

In figure 14, it is observed that the classifier missed most of the pedestrians walking on the street while having no problem detecting cars. In figure 15(a), it missed the man standing near the lamp post and in figure 15(b), it failed to detect a bunch of cars most likely because they are parked in a slanting way. When checking figure 16, it became obvious that the classifier performs better on “Car” class and worse on “Person” class. Figure 16 clarifies this problem.

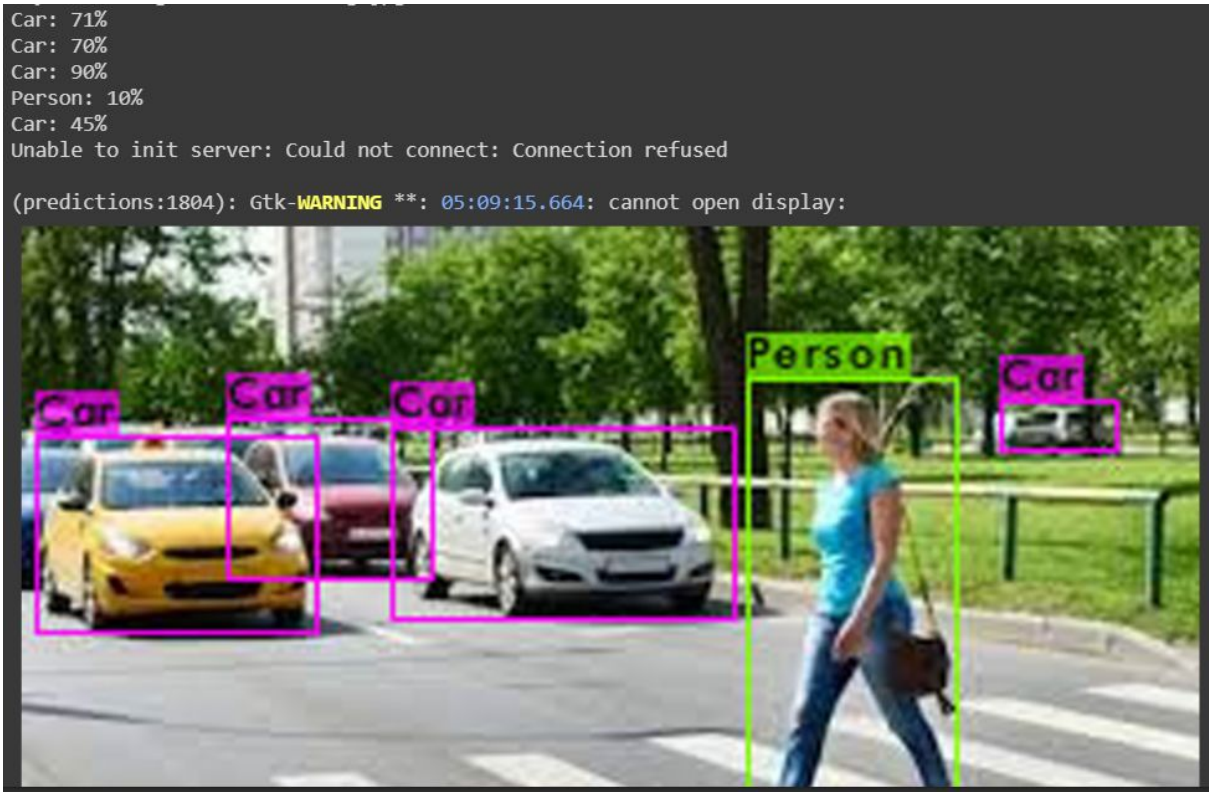

Fig 17: Confidence levels of multiclass classifier tested on Pedestrian Crossing

Checking the confidence levels in figure 17, it can be observed that this classifier can detect cars with ease but still struggles to detect person class. This is most likely caused by the slight imbalance of the input data among two classes, thus, car class will always perform better than person class at any loss level.

Conclusion

There are several ways to solve this problem. Training data can be simply balanced and the classifier can be trained again to neutralize this. The training can also be extended until an optimal loss value is obtained (usually below 2). Adding more specific data on pedestrians and parked cars can also help improve the performance.

Future Works

In this report, current progress on training and testing of the detection model is presented and it is likely to finish at the end of June 2020. In the meantime, several bug fixes and improvements on the classifier will be done. When the detection model is complete, the research will move on towards developing the tracking model.