Real time object detection project report

Heng-Ying Chou

Department of Mechanical Engineering, University of Washington, Seattle, WA 98195, United States

1 Introduction

In this project, we developed a camera system with real-time object detection for Sound Transit. We used this system to help Sound Transit to collect a parking space information. This camera system is composed by a dome camera and a development kit, Nvidia Jetson nano kit. We implemented object detection model in the Nvidia Jetson nano kit and collected occupied information of a parking space. In this report, we will go through the design about each components, including hardware and software parts. We will also discus about object detection results and data analysis. Finally, we will show the accuracy of the system and see if the system meets the requirements of the design.

2 Design procedure

2.1 Jetson nano kit

The Nvidia Jetson nano kit is an AI development kit with GPU computation. It supports multiple AI packages, including, pytorch, and tensorflow. It can be used for convolution neural networks for training models or computer vision for object detection. In our project, we use this development kit for detecting vehicles in the parking space with a object detection model. The 2G Jetson nano kit is in figure 1.

Figure 1: Jeton nano kit 2G

2.2 Camera selection

The first camera we selected for this project was Logitech C920 webcam. This camera is recommended by Nvidia because it is compatible with Jetson nano kit and Jetson inference which is an artificial intelligence package. However, there are few disadvantages of this camera. For one thing, C920 is not a waterproof camera. If we wanted to collect information in a parking space, heavy rain in Seattle could break C920 easily. In addition to waterproof camera, C920 does not have night mode function. If we wanted to collect the information for whole day, C920 could not collect anything at night. Hereby, we were looking for an alternative camera.

Figure 2: Logitech C920 webcam

Figure 3: Dome camera

Although C920 has good compatibility with Jetson nano kit, we still can find another camera which is compatible with Jetson nano kit. Fortunately, we found that a Dome camera which can satisfy our requirements. For one thing, Dome camera does not need any firmware to flash in and can be connected with Jetson nano kit with USB. This means the Dome camera is compatible with Jetson nano kit. In addition to compatibility, dome camera is a waterproof camera with night mode. We can use this camera for collecting the whole day data not matter it is a rainy day or night time. Hereby, the final camera we used was Dome camera.

2.3 Object detection model selection

The first object detection model we used in our capstone project was SSD-Mobilenet-V2. It contains 6 convolution layers for object detection, which is more suitable for portable device, such as smart phone, to process. This model is also included in the Jetson inference. We can use this model directly from this package. The figure 4 and 5 are object detection results by SSD-Mobilenet-V2:

Figure 4: SSD-Mobilenet-V2 object detection

Figure 5: SSD-Mobilenet-V2 object detection

We can notice that there are two vehicles identified in figure 4 but many objects in the figure had not been identified. In figure 5, it is apparent that SSD-Mobilenet-V2 did no identified most objects and gave an incorrect result. Although SSD-Mobilenet-V2 can identify some objects in a pictures, it does not necessarily satisfy our requirements.
Another object detection model we considered was YOLOv3. It contains 106 convolutional layers, which is a relative large model. Since Jetson nano kit has a dual core CPU in ARM cortex-A57, we thought that it could provide enough computational power for YOLOv3 model. Thus, YOLOv3 becomes one of our

potential models. The following figure is object detection result:

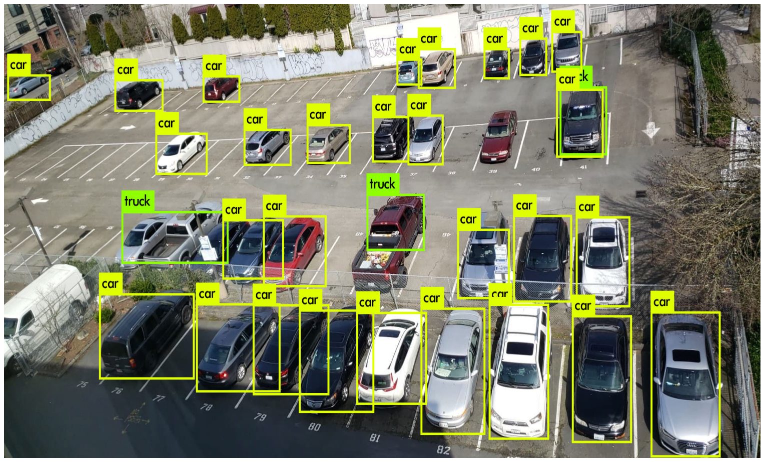

Figure 6: YoloV3 object detection

From figure 6, we can notice that YOLOv3 detected most objects. Although there are some objects that were not detected by YOLOv3, the result is still better than the result from SSD-Mobilenet-V2. Hereby, we finally choose YOLOV3 model as our object detection model.

2.4 Software programming

Since we wanted to receive the picture from the dome camera, we needed to use OpenCV to receive picture from the dome camera. However, the OpenCV version in YOLOv3 is OpenCV2 which is an old version OpenCV. This verson is not supported by many devices currently. Under this situation, using OpenCV with C programming has some difficulties. The way we overcame this was using python-cv2 package which supports OpenCV2 for YOLOv3. We can use this package to get picture from camera and process the object detection in YOLOv3. Hereby, we decided to use python as our programming platform.

2.5 Hardware implementation

Figure 7: Hardware implementations

In the figure 7, the picture with label 1 is the doom camera with night mode. When the night mode is on, we can take picture at night time. The picture with label 2 is our whole hardware system. We put Jetson nano kit in the white waterproof box and connect it with camera by USB cable. The picture with label 3 is our camera system and targeted park space. It is obvious that the camera is placed in the side view of parking space.

3 Result

3.1 Object detection results

The following picture is the result of implementing our solution in the day time:

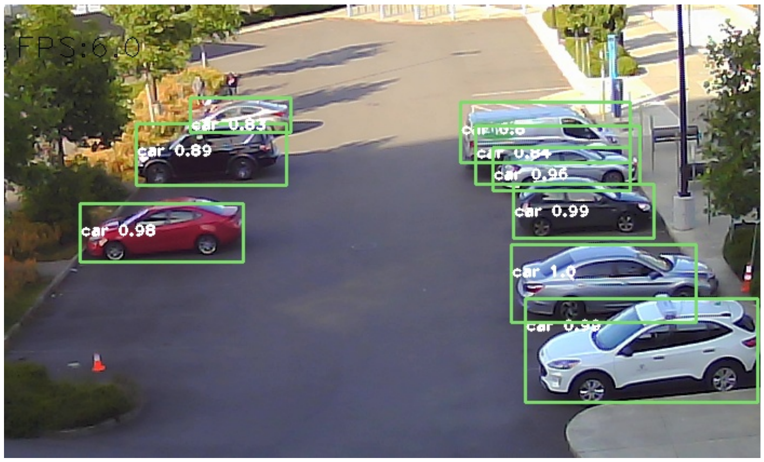

Figure 8: Our solution in day time

In figure 8, we can see that YOLOv3 works very well in object detection. Every vehicles in the park spaces are detected and most accuracy rates are above 80%. In the day time, this system satisfies our requirement. Since we want to collect the information , we also need to check the object detection in the night time. The following picture is our solution in night time:

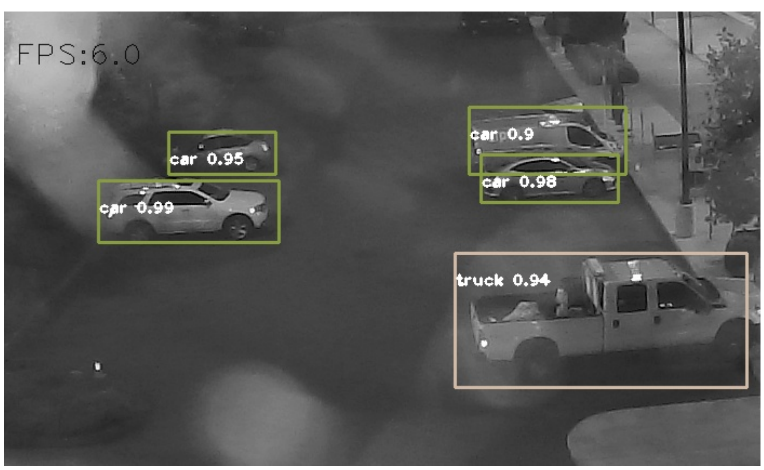

Figure 9: Our solution in night time

In figure 9, we can see that our solution find all object in the picture. Even though the camera uses night mode for getting gray scale picture, the object detection is not influenced it.We placed the camera system in the side of parking space which causes overlapping of vehicles. This can increase the object detection difficulty and influence the accuracy of our system. The following picture is an example of object detection with strong overlapping of vehicles:

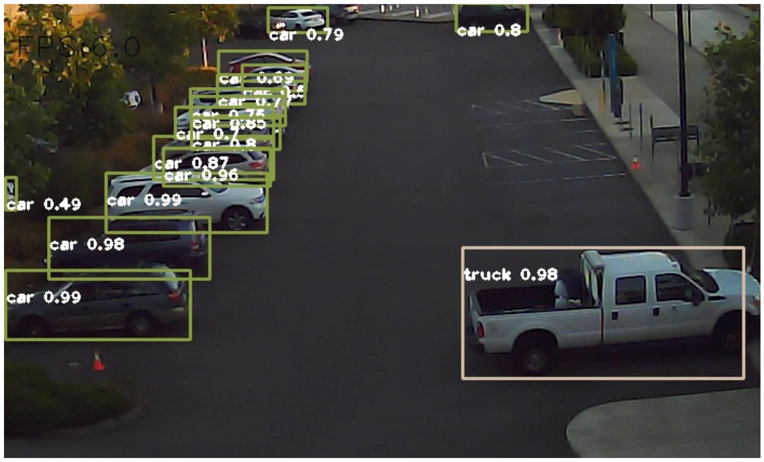

Figure 10: Our solution of strong overlapping of vehicles

In the figure 10, the strong overlapped vehicles influence the result of object detection which causes potential inaccuracy. This can cause one vehicle be detected as two vehicle or a vehicle cannot be detected because of frontier vehicle. Hereby, we need to define the definition of accuracy rate so that we can deal with these incorrect detection.

3.2 Confusion matrix and accuracy

Due to the overlapped vehicles, we need to employ a method for computing accuracy of our system. The method we employed is called confusion matrix which describes the correlation between True class and Predicted class. The following figure is the idea and explanation of confusion matrix:

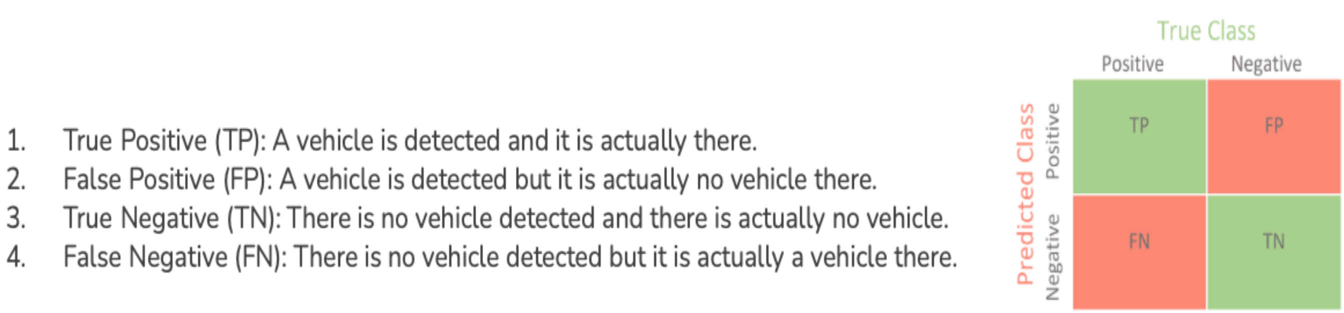

Figure 11: Confusion matrix and explanation

By this four correlations between True class and predicted class, we can define four rate to describe our result: Parking Occupancy Rate, Accuracy Rate, False Negative Rate, and True Negative Rate. The explanation of these four rates are in the following:

- Parking Occupancy Rate(POR): # all cars in the parking lot / # parking spaces
- Accuracy Rate: (#True Positive + #True Negative) / #Total
- False Negative Rate(FNR): # False Negative / # Total Positive

- True Negative Rate(TNR): #True Negative / # Total Negative

We can compare these four rates to know the performance of our system. The comparison results are shown in the following sections.

3.2.1 Parking Occupancy Rate versus Accuracy

The following picture shows parking occupancy rate versus accuracy rate.

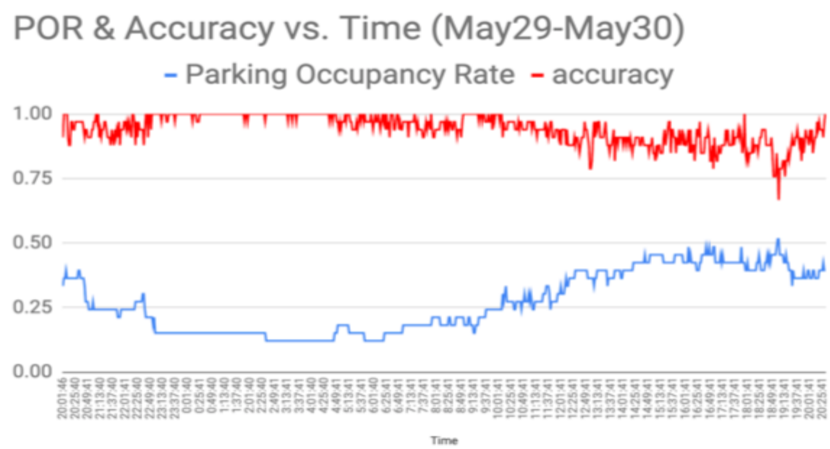

Figure 12: Parking Occupancy Rate versus Accuracy

We can notice that in most time, the accuracy rate reaches around 90%. However, once the parking occupancy rate increases, the accuracy rate is lowered down. This is because the overlapped vehicles influence the object detection which cannot exactly distinguish every vehicles in the parking space. Although the accuracy rate is lowered, it still can reach around 75% - 80 %. This shows that YOLOv3 can find most vehicles even though the vehicles are overlapped.

3.2.2 Parking Occupancy Rate versus False Negative Rate

The following picture shows parking occupancy rate versus false negative rate.

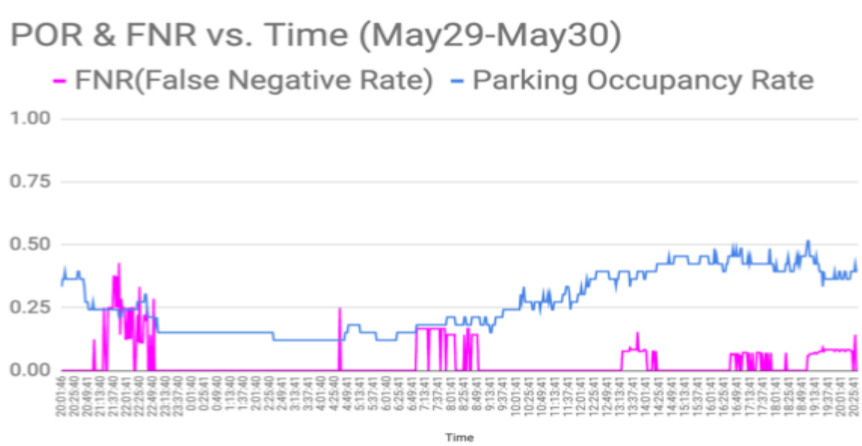

Figure 13: Parking Occupancy Rate versus False Negative Rate

The meaning of false negative rate is showing how many cars are identified as a negative(empty space).Because the overlapped vehicles, some vehicles would not be identified and be regarded as empty space. This causes the increase of false negative rate and lowers the accuracy of the system. In the figure 13, we can notice that, in most time, false negative rate maintains around 0%. However, there are some points that false negative rate increases, but it does not keep increasing when parking occupancy increases. Most increased false negative rates maintain between 10% - 25%. This shows that YOLOv3 keeps good false negative rate even though the vehicles are strongly overlapped each other.

3.2.3 Parking Occupancy Rate versus True Negative Rate

The following picture shows parking occupancy rate versus true negative rate.

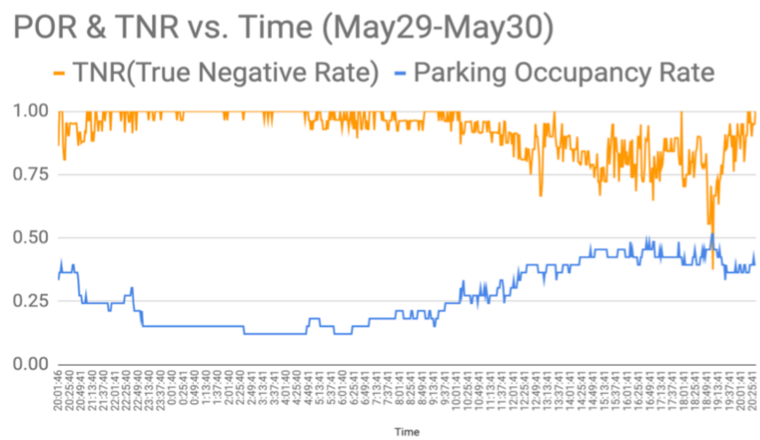

Figure 14: Parking Occupancy Rate versus True Negative Rate

From figure 14, we can notice that when parking occupancy rate is low, the true negative rate is high. However, once parking occupancy increases, the true negative rate lowers down. This is because the overlapped vehicles causes difficulty for YOLOv3 to distinguish the empty spaces between vehicles. Hereby, when the occupancy rate increases, the true negative rate decreases.

4 Conclusion

From our camera system data, we observed that once the parking occupancy rate increased, the accuracy rate decreased. The reasons for this are twofold. The first, also more general, reason is that with the base number of cars increasing, the error rate would increase accordingly, which is the nature of any detection system. The second reason is specifically related to the choice of camera angle to the parking lot. Our camera system was placed in the side view of parking. From this view, the vehicles in the parking lots would overlap each other and made the camera system more difficult to distinguish vehicles. If the number of vehicles in parking spaces increases, YOLOv3 could not distinguish more vehicles and decrease accuracy rate. Hereby, we can observe that as the number of vehicles increased, the accuracy rate decreased.