# Deep Learning - Lab 04

# IT21277122 – Withanagamage J.C.

## **Question 7**

The reason there are 1783 boxes is that the YOLO model splits the input image into a grid, typically consisting of 19x19 cells in this case. Each of these cells predicts 5 bounding boxes, which corresponds to 5 anchor boxes. Initially, this results in 19x19x5 = 1805 boxes. However, the yolo\_filter\_boxes function then applies a confidence threshold, removing boxes with scores below this threshold. After this filtering process, 1783 boxes remain, each characterized by four parameters: width, height, x-center, and y-center.

The maximum number of boxes is achieved when all initially predicted boxes have confidence scores above the threshold, resulting in all 1805 boxes being retained. On the other hand, the minimum number of boxes is reached when none of the boxes meet the confidence threshold, leaving zero boxes. Therefore, the number of boxes after filtering can range from 0 to 1805.

## **Question 8**

Anchor boxes provide a major advantage by enabling the model to assess all possible object predictions in parallel, rather than using a sliding window technique that requires separate predictions for each potential location in the image. This efficiency is achieved because anchor boxes have predefined shapes and sizes, allowing the model to predict objects of various dimensions in a single pass through the image, which significantly accelerates the detection process.

The sizes of anchor boxes are usually determined through a mix of manual selection and data analysis. A common approach is to use a clustering algorithm, like K-Means, on the dataset. This algorithm identifies the most frequent bounding box shapes and sizes within the dataset, which are then used as the anchor boxes. By customizing the anchor boxes to match the specific characteristics of the data, the model can more accurately predict the objects it detects.

## **Question 10**

****



In this instance, no bounding boxes were detected despite the presence of vehicles and traffic lights. This lack of detection could be due to the unique shape of the vehicle, which might not have matched well with the anchor boxes or the model's predictions. Furthermore, the traffic lights might have gone unnoticed because of the lighting conditions in the image, which could have affected the model's ability to identify them.

In the second image, both a vehicle and a traffic light were successfully detected. This observation supports the earlier assumptions, as the vehicle was detected even though only half of it was visible, indicating that the vehicle's shape or partial visibility in the previous image could have impacted the detection results. Additionally, the detection score for the traffic light in this image was very close to the threshold, which could explain why traffic lights in other images with less favorable conditions were not detected.

## **Question 11**

max\_boxes = 30



**

Despite adjusting the max\_boxes value, there appears to be no noticeable improvement.

score\_threshold = 0.3

*A street with traffic signs

Description automatically generated*



By reducing the score threshold from 0.6 to 0.3, the model has detected more objects, including both traffic lights and vehicles. However, in the first image, it incorrectly labeled one object as a "bus" when it should have been identified as a "truck."

iou\_threshold = 0.8 and iou\_threshold = 0.2 on 2 separate runs

*A street with a green light

Description automatically generated*

*A crosswalk with a bicycle lane and traffic lights

Description automatically generated*

I adjusted the IoU threshold values to 0.8 and 0.2 and tested the model in two separate runs, but there was no significant improvement in the results.