

## Vehicle Detection Project

In this project, I have used the Deep Learning method in order to detect the vehicles. I have used an open source neural network called YOLO tiny, which uses the regression model in order to spot the vehicles and takes the whole image to process.

The YOLO model divides an image into smaller grids. Then, each grid has number of cells. For this task, I have followed the YOLO principles (paper can be found here: ) and used 7x7 for the single grid (or 49 total cells per grid). This means, that we will get predictions for each 20 classes presented by the YOLO tiny pretrained model for the each cell. That is  $20 \times 49 = 980$  predictions. Then, the number of boxes to be found is  $B=2$ , which means that in the whole grid (49 cells) we will obtain  $49 \times 2 = 98$  boxes. Finally, the model will give us 4 coordinates, x,y,w,h. Therefore, in each cell we will get 42 (4 points in each B-box) = 8 and for the whole grid we get  $8 \times 49 = 392$  intersection points. Therefore, the whole number (ordered by class predictions, B and intersections) of data we obtain is  $980 + 98 + 392 = 1,470$ .

The network consists of 9 convolutional layers, 6 of them being followed by the Max Pooling and all of them with an activation function LeakyReLU. Then, the network has 2 dense layers, second one followed by the LeakyReLU and finally the last dense layer with 1,470 points.

In order to obtain the best results, first, I have cropped the image in y dimension, as the vehicles can be seen only in a limited area of the image. Then, I have resized the image to 448x448 and transformed the shape of it to 2,0,1 – as YOLO network accepts only in this format.

Next, I have normalised the values in range -1 to 1, in order to get the maximum advantage of the activation functions.

After preparing the image, it was sent to the CNN and the weights for the network have been already trained and available on the YOLO website. Having an array of shape (1470,1), it was time to draw the rectangles. First, I have selected only class 6 out of 20 available by the YOLO tiny model. Class 6 is “Vehicles”. Then, I have divided an array and created new arrays that store the class probabilities, the coordinates (x,y,w,h) and the array that stores the confidence scores for each box.

After detecting the probabilities for each box and finding any overlapping boxes (if boxes overlap, we take the box with the higher probability score). Then, the boxes are drawn according to the cropped coordinates in order to place them correctly

NOTE: the video can be seen here <https://youtu.be/L0zJcnZlrpU>

Discussion:

The YOLO tiny is a very robust model, that can work up to 150 frames per second. It is quite light network but therefore it lacks some accuracy. YOLO tiny can be replaced by just YOLO, which can still be very robust as the video we have is not even near to 150 frames per second.