Project: Analyzing the Impact of Hyperparameters on the Average Testing Time of a Faster R-CNN Model

Submitted By: Aadit Juneja, Adlai E. Stevenson High School

Dept. of Engineering

Northwestern University

Date Submitted: August 5, 2022

MDS Summer Class 2022

Table of Contents

ABSTRACT 3

INTRODUCTION 3

BACKGROUND AND MOTIVATION 4

Figure 1 4

DATASET INFORMATION 4

HOW FASTER R-CNN WORKS 5

Figure 2 5

Figure 3 6

Figure 4 6

RESULTS 6

Table 1 6

CONCLUSION 7

CODE 7

Analyzing the Impact of Hyperparameters on the Average Testing Time of a Faster R-CNN Model

Aadit Juneja

ABSTRACT: With the advent of self-driving cars, object detection algorithms are necessary for these machines to determine what objects are in their path. Object detection is a task defined as drawing a bounding box around each object and classifying it from a predefined set of possible classifications. Object detection must be completed quickly in order for it to be practical. Faster R-CNN is one modern object detection algorithm developed in 2016 commonly used today by industry professionals. It builds on its predecessors, R-CNN and Fast R-CNN, by replacing the initial Selective Search algorithm, which generates regions of interest in which an object may exist, with a learnable, deep learning based Region Proposal Network to generate regions of interest. The Region Proposal network works via convolution operations and pooling the initial image, then generating various boxes (called anchors) from the feature map to the initial image. Each anchor is then tested against the ground truth objects using the IoU method to determine if that anchor is background or an object. If it is an object, it is then considered a region of interest by the Region Proposal Network which is passed onto the 2nd stage of the Faster R-CNN model. In this stage, a region of interest goes through more convolutional layers and pooling layers, and then goes through some fully connected layers that output a classification for the region of interest, as well as 4 coordinates of the bounding box. The entire model is built on a VGG16 model to generate the initial feature maps. The number of fully connected layers at the end of the model are varied, and it was found that as the number of fully connected layers is increased from 3 to 15, the average testing time stays roughly the same, though the recall rate increases, likely due to the large number of parameters in the VGG16 model.

KEYWORDS: *Object Detection, Deep Learning, Faster R-CNN*

INTRODUCTION

Autonomous vehicles have been on the rise in the past couple of years. Companies like Nuro, Waymo, and Motional are working to create fully autonomous vehicles to complete various tasks that humans would otherwise do by driving around. An important task autonomous vehicles must complete is called object detection. Object detection is a task in which a frame is fed in, and each object (as defined by a pre-chosen set of possible classifications) is classified with a bounding box drawn around them. In the 2000s Faster R-CNN, a model developed in 2016, is one popular approach to this task. It is vital that these object detection algorithms have a fast runtime, since they need to detect objects and avoid them before any crashes occur. Additionally, they must be highly accurate for the same reason. The research conducted examines how a Faster R-CNN model's recall rate and average testing time change when the number of fully connected layers are varied.

BACKGROUND AND MOTIVATION

In 2014, the Navia shuttle became the first commercialized self-driving car. When used on the open road, it is imperative that self-driving cars can process new information as fast as possible. As such, the task of object detection must have a fast runtime in autonomous vehicles.

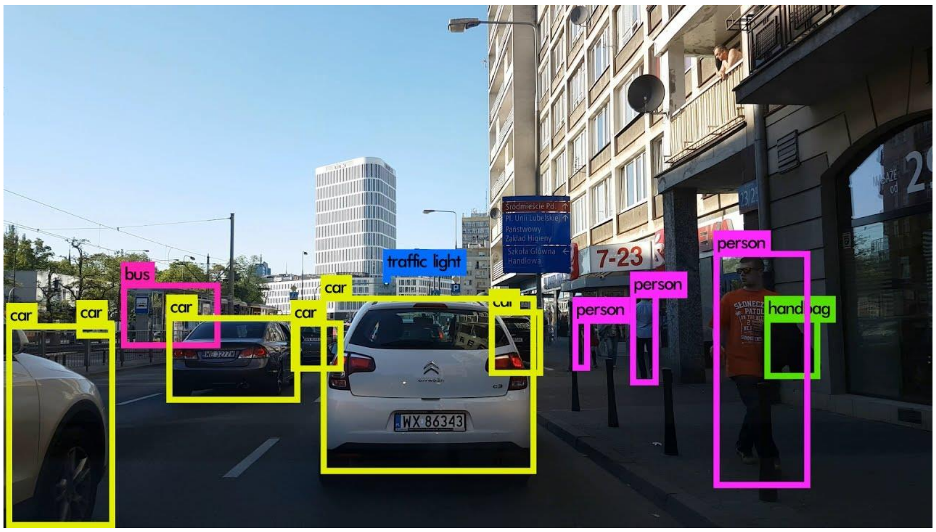

Fig 1: An example of the task of object detection. Cars, pedestrians, and other classes have each instance labeled with a bounding box and the classification.

Faster R-CNN, a model developed in 2016 by researchers Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, is a state of the art object detection algorithm. This project aims to determine how varying the number of fully connected layers at the end of a Faster R-CNN model will impact the average testing time of the model.

DATASET INFORMATION

The dataset used was Google Open Images. 3 classes were selected for study: cars, persons, and bicycle wheels, in equal quantities. These classes were selected to make the model relevant for autonomous vehicles. All data regarding segmentation masks or bounding boxes for other classes in images were ignored, and only these classes were considered.

HOW FASTER R-CNN WORKS

The Faster R-CNN algorithm (2016) can be split into 2 parts: generating region proposals, and processing region proposals. The processing of region proposals part of the model acts the same as Faster R-CNN's model, Fast R-CNN model (2015). The difference lies in generating region proposals. Faster R-CNN uses a learnable Region Proposal Network (RPN), while Fast R-CNN used a fixed algorithm called Selective Search.

Firstly, the RPN will be explained. The images are initially sent through a pretrained model called VGG16 to generate a new feature map. Each point in the feature map is then mapped back to boxes (called anchors) on the original image through a combination of 3 aspect ratios and 3 scales, for a total of 9 anchors per point on the feature map.

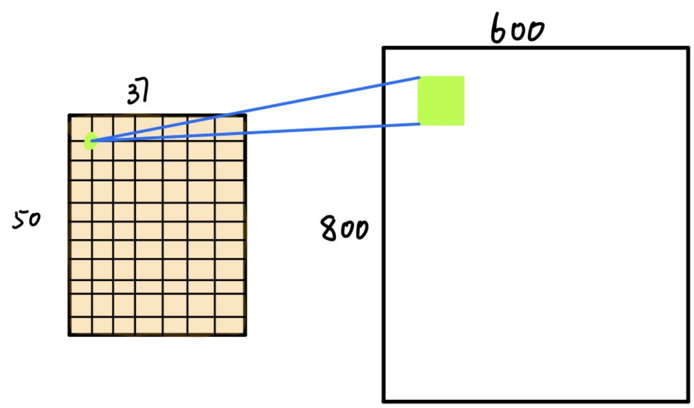

Fig 2: An example of a point in the feature map being mapped to an anchor.

Anchors are sent through convolutional layers and trained to output a classification (of whether anchor is an object or background) and a regression (for bounding box coordinates) using the intersection over union (IoU) method. Anchors classified as objects are called Regions of Interest (RoI) and are the output of the RPN.

Fig 3: An example of how the intersection over union method can be used to compare ground truth bounding boxes to RPN generated bounding boxes.

RoIs are then sent through convolutional layers and pooling layers. Then the feature map is flattened and sent through fully connected layers, outputting a classification (of bicycle wheel, person, or car) and a regression (bounding box coordinates).

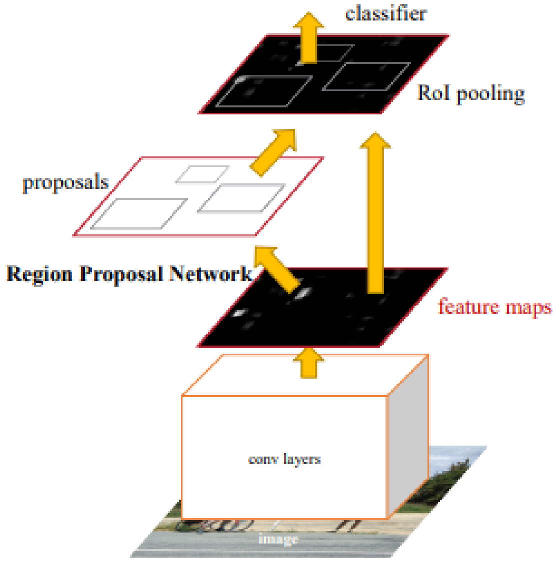

Fig 4: An overview of Faster R-CNN.

RESULTS

Table 1: Model behavior after varying the number of fully connected layers.

Number of FC Layers	Recall Rate	Average Testing Time (s)
3	72.56%	0.691

5	75.88%	0.691
7	76.91%	0.692
15	78.26%	0.693

We see a very negligible difference in the average testing time when the number of fully connected layers are varied, likely due to the presence of the pre-trained VGG16 base model, which has over 138 million parameters. Each additional fully connected layer only adds about 1000 more parameters, meaning the average testing time will not change much. However, the recall rate does increase in this scenario, meaning the model has increasing accuracy without needing much extra computation time.

CONCLUSION

Overall, when VGG16 is used as a base model for Faster R-CNN, changing the number of fully connected layers will not change average testing time by much. However, other base models exist, such as ResNet 50 (23 million parameters) and LeNet-5 (60,000 parameters). Conducting the same research with these base models rather than VGG16 may allow for us to see a more notable difference in average testing time due to the fewer number of parameters. In terms of my learning, this experience allowed me to dive into computer vision and understand the cutting edge models concerning the field today. Furthermore, it was interesting observing the connection between my project and technologies like self-driving cars. Overall, the MDS research experience was incredibly rewarding and one that I would highly recommend.

CODE

See code at https://github.com/ajuneja23/Faster-R-CNN-Research