

EE\_ 600 Project:

Comparative Analysis  
of  
Faster R-CNN & SSD  
on the  
Lisa Traffic Dataset

By:

Chowdhury Erfan Shourov & Mohammadshayan Latifi

Course: Machine Learning (EE600)

Date: 12/8/2019

## **I. Introduction:**

The objective of this project is to compare two State of Art (SOA) Object Detection models, the Faster Region-based Convolutional Neural Network (Faster R-CNN) and the Single Shot Multi-Box Detector (SSD). The Faster R-CNN and the SSD has competitive accuracy in classic image challenges like the PASCAL VOC [5], COCO [7] and the ILSVRC [6] datasets. While the Faster R-CNN model outperforms the SSD in terms of accuracy, it only has a 5-9 fps which is not ideal for real time object detection [4]. The SSD, while slightly less accurate, has up to 59 fps which makes it suitable for real-time object detection [4]. Both the models were trained using the Tensor Flow Object Detection API [9], which is widely used in the industry, to implement various machine learning techniques.

## **II. Faster R-CNN Architecture:**

The Faster R-CNN is the third and final version in the family of RCNNs. It is an extension of the RCNN and an extension of the Fast R-CNN. In order to understand Faster RCNN, it is important to discuss these two previous versions. The RCNN (Region-based Convolution Neural Network) has the following structure.

The idea is to use Selective Algorithm to obtain region proposals (i.e. parts in an image that could be a potential object) and then passes it to a Convolutional Neural Network (CNN) followed by a Support Vector Machine (SVM) which would classify the object. [1] CNNs have layers which are rich at extracting features such as edges, blobs etc. A thorough description of the Selective Algorithm is out of the scope of this report. It can be assumed to be like a black box that provides region proposals for an image. Unfortunately, this method is extremely slow. One month after the publication of the R-CNN, the Fast R-CNN was published. In this version, the region proposals still resulted from the same Selective Algorithm, but the entire image is first sent through a CNN and a convolutional feature map is projected. After that the regions proposed from the Selective Algorithm is extracted in fixed length using a Region of Interest (ROI) Max Pooling layer. The output of the ROI Pooling layer is connected in parallel to two fully connected layers ; one is a soft max classifier which would give the confidence level of the object detected and the other one providing the x, y coordinates which would be the coordinates of the bounding box that defines the object.[2] Even though this increased the speed of the detection as compared to R-CNN , because of the Selective Search module , it was not fast enough and nowhere near real time object detection. Hence the default solution was to use a better region proposal algorithm that took less time than the Selective Search module. This led to the Faster R-CNN where the Selective Search Module was replaced by a novel Regional Proposal Network (RPN) [3]. The RPN is a CNN which proposes a region that could potentially have an object and this is passed to the Fast RCNN detector. The RPN network simultaneously predicts object bounds and confidence scores on those objects at each position.

The R-CNN achieved a mAP of 53.3% on the Pascal VOC challenge. The Fast RCNN achieved a mAP of 66% on the same challenge whereas the Faster RCNN achieved a mAP of 78.8 %. To understand what the mAP metric is, please refer to section V.

The difference in all the three architectures are provided in the Figure [1]. Figure [2] and Figure [3] below:

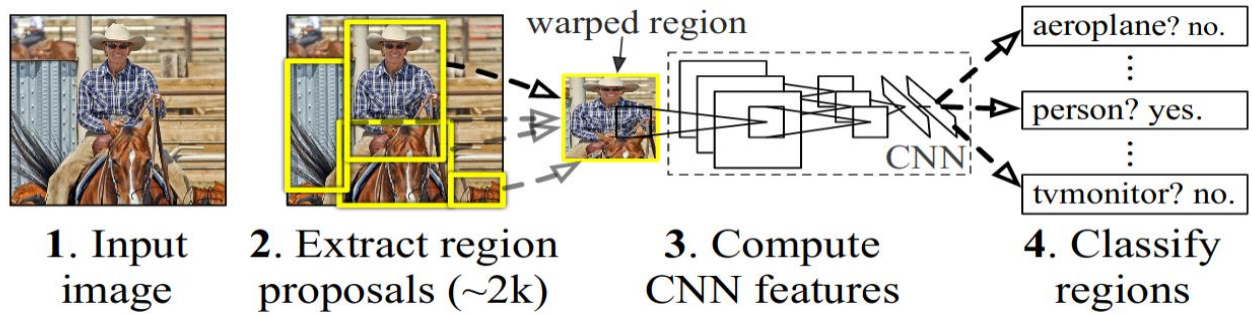


Figure 1: Architecture of a Region-based Convolutional Neural Network (R-CNN)

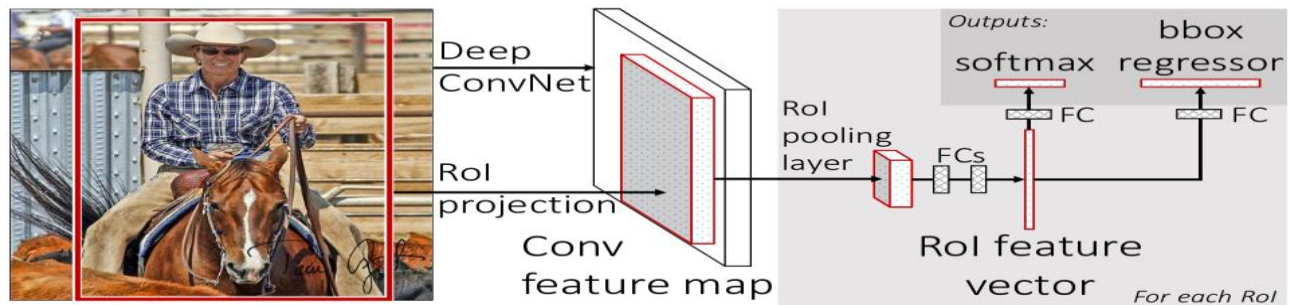


Figure 2: Architecture of a Fast Region-based Convolutional Neural Network (Fast R-CNN)

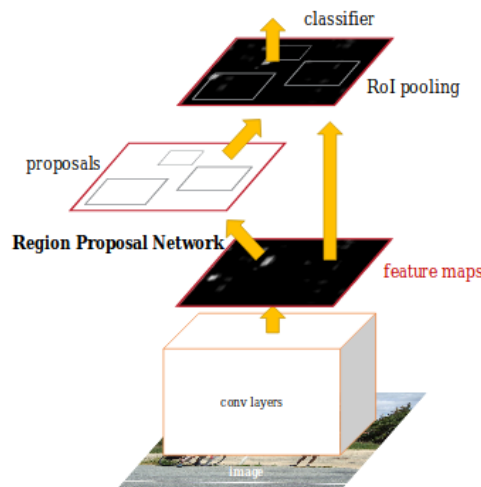


Figure 3: Architecture of a Faster Region-based Convolutional Neural Network (Faster R-CNN)

### III. Single Shot Multi-Box Detector (SSD):

The Single Shot Multi-Box Detector (SSD) uses a single deep learning network [4]. The SSD model completely removes the region proposal generation module and encapsulates everything in one single network. Eliminating the region proposal generation makes it faster which is the fundamental improvement in terms of speed.

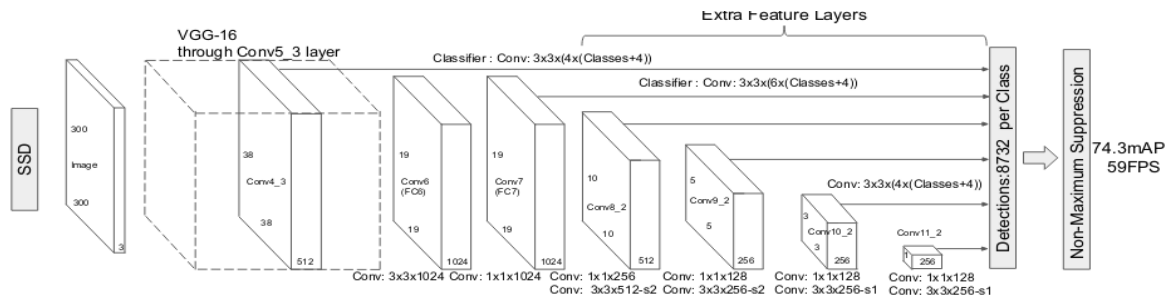


Figure 4: Architecture of a Signal Shot Multi-Box Detector (SSD)

It can be seen from the image that the initial image is passed through a VGG Convolutional Network [10] till the 6<sup>th</sup> layer after which it detaches the rest of the layer including the fully connected layers. Instead, a new set of CNNs are added. These CNNs progressively becomes smaller and smaller. Also, each of the Convolutional layers connect to the final detection layer. This implies that the network naturally handles objects of various sizes and scales. The SSD achieved 74.3 % mAP on VOC 2007 test at 59 FPS [4].

#### IV. The Data set:

The Lisa Traffic Sign Dataset (LTSD) is a set of videos and annotated frames containing US traffic signs [8]. The LTSD contains 47 US traffic signs. It consists of 6610 frames with 7855 annotations. The Faster R-CNN and the SSD models can be made to prepare this dataset for autonomous vehicles in order to guide itself through streets without human conduction by recognizing the signs and act accordingly. Because training on the full 47 different labels of the road signs will be computationally exhaustive and may take weeks to train, only three labels were considered for this particular project; “Stop Sign”, “Pedestrian Crossing” and “Signal Ahead”. However, the results of these simulations can be generalized for the full data set.



Figure 5: An example during training from the data set where the left image is what the network is trying to detect and the right image is the ground truth (actual bounding box) [8] associated with that image. This example was taken at the initial starting of the training and it can be seen that the model fails to identify the sign. This will lead the model to redefine its current weights.





Figure 6: Another example during training where the model prediction (Left) matches the ground truth (Right) [8]

## V. Performance Metrics for Object Detectors:

Evaluation of object detector performances are based on the combination of two evaluation metrics: Intersection Over Union (IOU) and mean Average precision (mAP). Consider the following example.

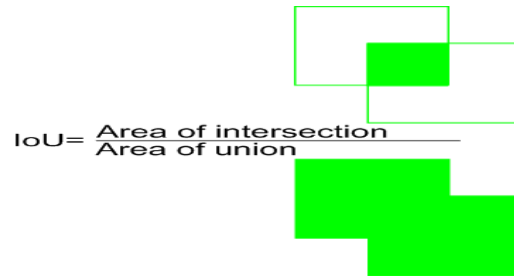
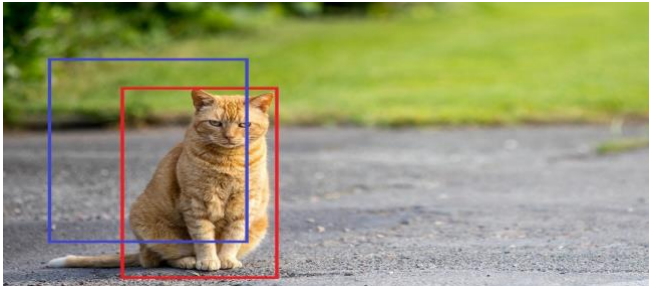


Figure 7: Example of a cat (left) being detected where the red box is the ground truth and the blue box is the bounding box generated by a detector. On the right, visual that explains the concept of IOU performance metric. Both images were taken from [11].

In the above image, Figure [7], the red bounding box is the ground truth bounding box (i.e. the true label) whereas the blue bounding box is the predicted bounding box (i.e. the prediction made by the object detector).

The IOU, Figure [7], is just a simple ratio of the area of intersection to the area of the union of these bounding boxes. It can be observed from the equation that the greater the value of overlap in the numerator, the higher the IOU. An IOU of greater than 0.5 is considered to be an above average prediction.

The mAP is another evaluation metric used for the object detection. To discuss how to calculate mAP is out of the scope of this project. But in short, the calculation of the mAP is based on the precision and recall values. The Precision is the number of correctly predicted objects in an image a Recall is how good the detector is doing finding all the objects in the image. The Precision and Recall values depend on True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) Values. TP means that the detector is correctly predicting an object. TN means the detector is correctly detecting

the absence of an object. FP means that the detector is falsely predicting an object when there is none. Finally, FN means the detector is failing to report the location of an object entirely. The following equations are used to calculate the Precision and Recall.

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

The idea is to calculate the Precision at different Recall values. This yields a list of precision values as the process is repeated for typically the top-1 to top-10 predictions (a.k.a Rank). The average of the top-N values results in the mean Average Precision.

## **VI. Simulation Results:**

This section discusses the findings of the Faster R-CNN and the SSD model on the LTSD with respect to the mean absolute precision (mAP) at 0.5 IOU and 0.75 IOU, Loss 1, Loss 2, the evaluation loss and the frame rate.

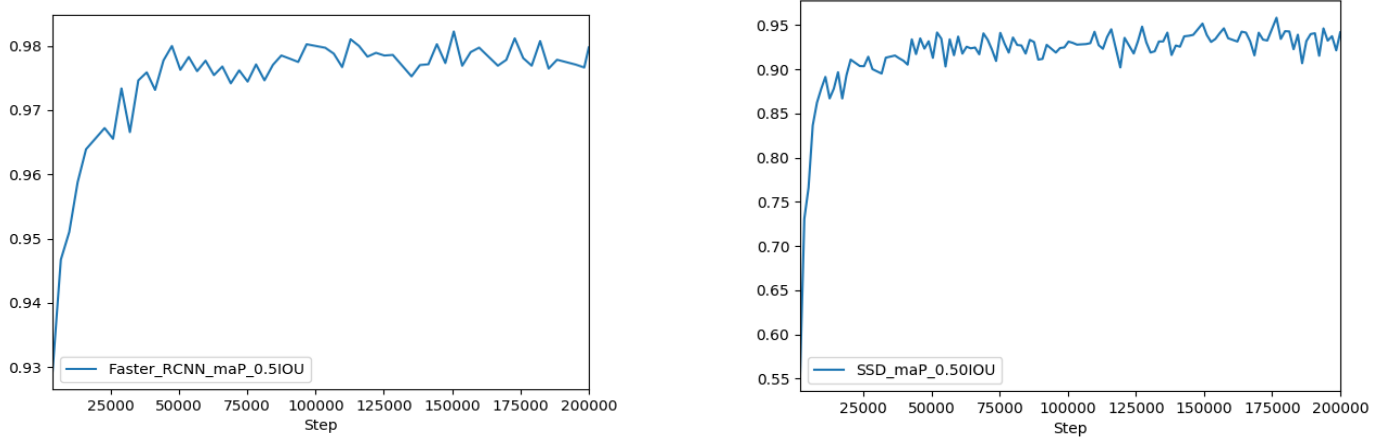


Figure 8: mAP of the Faster R-CNN and the SSD on the LISA Traffic Dataset at 0.5 IOU

From Figure [8], it can be seen that the mAP @ 0.5 IOU for the SSD model is lower than the Faster-RCNN. The same can be observed for the mAP@ 0.75 IOU Figure [9]. From Figure [10,11], it can be seen that both loss functions are lower for the Faster R-CNN than the SSD. This means that the Faster R-CNN is highly accurate when predicting than the SSD model. In terms evaluation loss, Figure [12] between the two, Faster R-CNN still surpasses the SSD model. It can be seen that in every aspect the Faster R-CNN is better than the SSD model. However, this comes at a price. The Faster R-CNN model only has a frame rate of 5-9 fps which is not ideal for real time object detection. The SSD, on the other hand, has up to 59+ fps, which is way above par for real object detection.

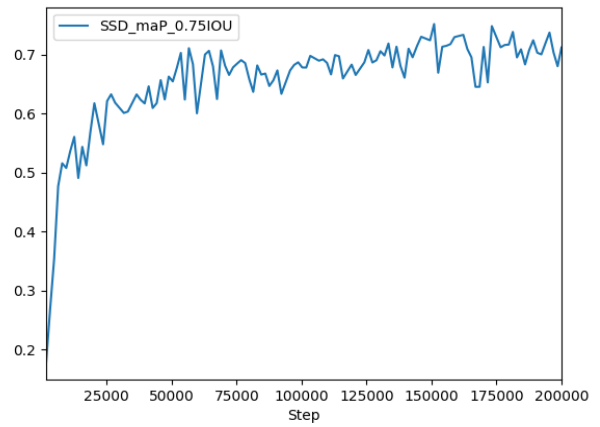
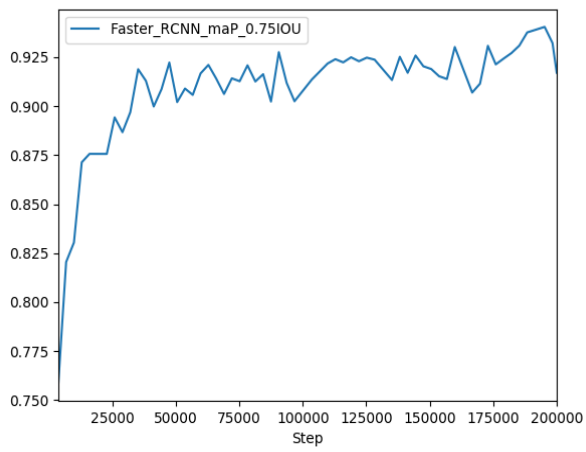


Figure 9: mAP of the Faster R-CNN and the SSD on the LISA Traffic Dataset at 0.75 IOU

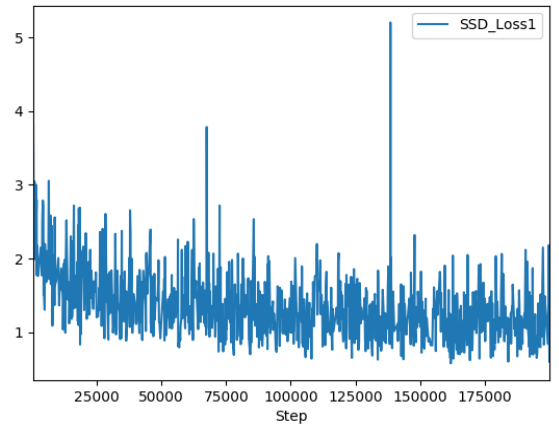
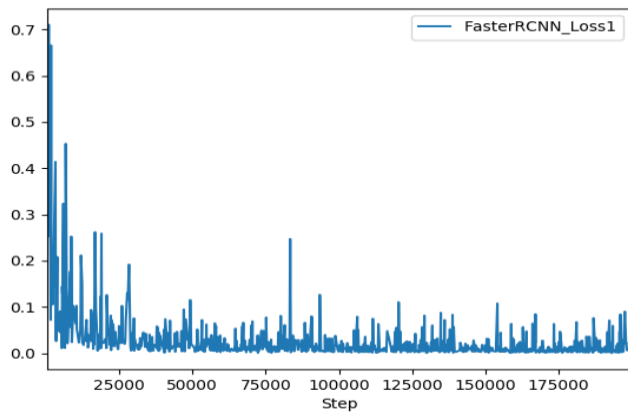


Figure 10: Loss 1 of the Faster R-CNN and the SSD on the LISA Traffic Dataset

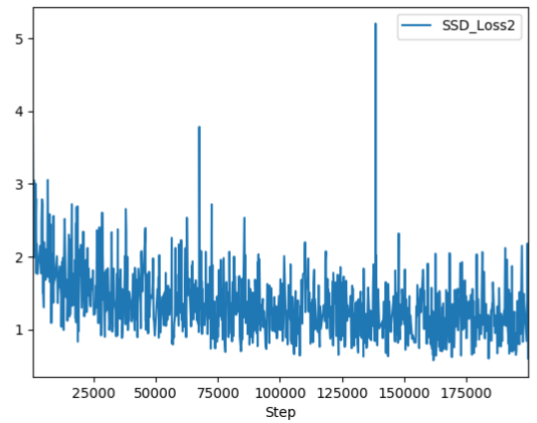
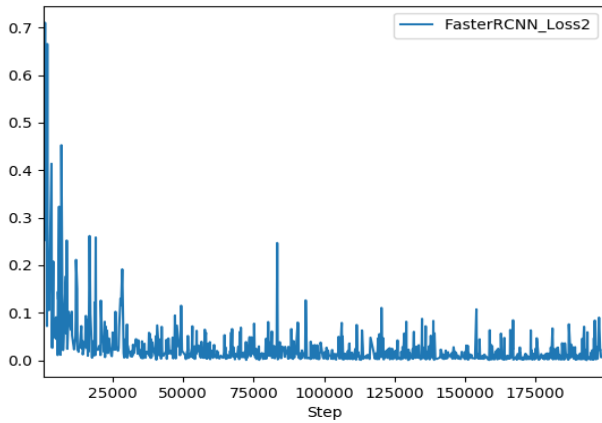


Figure 11: Loss 2 of the Faster R-CNN and the SSD on the LISA Traffic Dataset

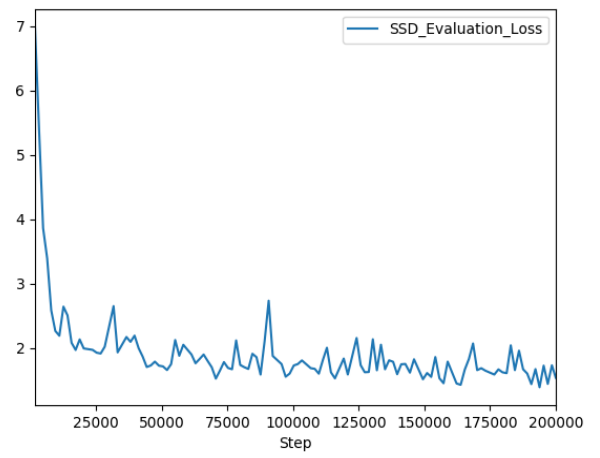
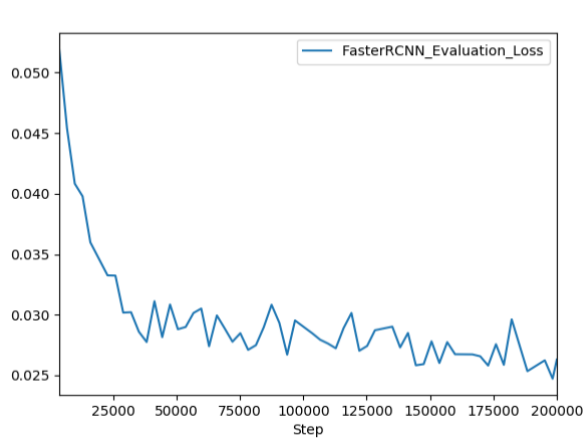


Figure 12: Evaluation Loss of the Faster -CNN and the SSD on the LISA Traffic Dataset



Figure 13: Faster RCNN (left) and SSD(right) models both correctly detecting “STOP” sign but with different confidence levels. Faster RCNN has a confidence of 1.00 on detecting the “STOP” sign and the SSD model has a confidence of 0.90.

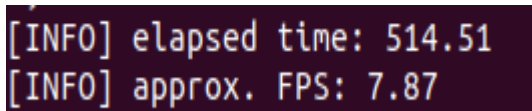




Figure 14: Faster R-CNN (top) and SSD (bottom) models both correctly detecting “STOP” sign and “Pedestrian Crossing” but with different confidence levels. Faster RCNN has a confidence of 1.00 for both the signs but the SSD model detects the stop sign with a confidence of 1.00 while the “Pedestrian Crossing” with a confidence level of 0.32.

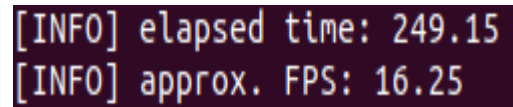
Both the models were evaluated by taking images around the San Diego State Campus; images that the models have never been trained on. Figure 13 and Figure 14 show some of the examples and the caption explains the difference in prediction between the Faster RCNN and the SSD.

However, for real time object detection, what gives the SSD an edge over the Faster RCNN is the frame rate. A video was captured of length 3 min and 22 seconds and the frame rates were compared. The Faster RCNN, from Figure 15, can be seen has a fps of 7.87 which is not ideal for real time object detection. However, despite having less accuracy, the SSD model has a frame rate of 16.25, Figure 16, which is above par and a suitable choice for real time object detection.



```
[INFO] elapsed time: 514.51
[INFO] approx. FPS: 7.87
```

Figure 15: Faster R-CNN Frame Rate



```
[INFO] elapsed time: 249.15
[INFO] approx. FPS: 16.25
```

Figure 16: SSD Frame Rate

## **VII. Conclusion**

In this project, the two SOA models used for object detection, namely the Faster RCNN and the SSD models, were compared with each other. Both were tested on the LTDS and the performances were measured in terms of mAP, loss functions, evaluation loss and the frames per second. It was observed that in most cases, the Faster RCNN outperforms the SSD. However, because the objective is to detect traffic signs in real time, SSD has the edge of having a faster fps which is ideal for real time object detection. It can therefore be generalized that when the goal is to identify objects in an image, the Faster RCNN is an appropriate and better model. However, when it is about detecting objects on a video stream, the SSD is the best choice between the two.

## **VIII. References**

- [1] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580-587).
- [2] Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 1440-1448).
- [3] Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 1440-1448).
- [4] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016, October). Ssd: Single shot multibox detector. In *European conference on computer vision* (pp. 21-37). Springer, Cham.
- [5] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2), 303-338.
- [6] Large Scale Visual Recognition Challenge (ILSVRC). (n.d.). Retrieved from <http://www.image-net.org/challenges/LSVRC/>.
- [7] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014, September). Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740-755). Springer, Cham.

- [8] Laboratory for Intelligent and Safe Automobiles - CVRR - UCSD. (n.d.). Retrieved from <http://cvrr.ucsd.edu/LISA/lisa-traffic-sign-dataset.html>.
- [9] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Kudlur, M. (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)* (pp. 265-283).
- [10] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- [11] Rakshit, S. (2019, April 15). Intersection Over Union. Retrieved from <https://medium.com/koderunners/intersection-over-union-516a3950269c>.