This report demonstrates an object detection model that has been trained by the dataset I had made. Faster RCNN was used as an object detection model. It was trained on annotated traffic dataset with objects such as- Car, Motorcycle, leguna, pedestrian, bus, truck, CNG, rickshaw. The goal is to train an object detection model to detect mentioned objects. This model is highly suitable for building traffic monitoring systems and autonomous driving systems.

Model:

Faster RCNN is an object detection model. It is a state-of-the-art object detection architecture of the R-CNN family. Faster R-CNN architecture consists of two components

1. **Region Proposal Network (RPN):** Region Proposal Network (RPN) is an essential component of Faster R-CNN. It is responsible for generating possible regions of interest (region proposals) in images that may contain objects. It uses the concept of attention mechanism in neural networks that instruct the subsequent Fast R-CNN detector where to look for objects in the image. The key components of RPN are given blow:
    a. **Anchors boxes:** Anchors are used to generate region proposals in the Faster R-CNN model
    b. **Sliding Window approach:** The RPN operates as a sliding window mechanism over the feature map obtained from the CNN backbone. It uses a small convolutional network (typically a 3×3 convolutional layer) to process the features within the receptive field of the sliding window.
    c. **Objectness Score:** The objectness score represents the probability that a given anchor box contains an object of interest rather than being just background. In Faster R-CNN, the RPN predicts this score for each anchor. The objectness score reflects the confidence that the anchor corresponds to a meaningful object region.
    d. **IoU (Intersection over Union):** Intersection over Union (IoU) is a metric used to measure the degree of overlap between two bounding boxes
    e. **Non-Maximum Suppression (NMS)**: NMS is used to remove redundancy and select the most accurate proposals, based on the objectness scores of overlapping proposals and keeps only the proposal with the highest score while suppressing the others.

2. **Fast R-CNN detector:**
   a. **Region of Interest (RoI) Pooling:** The first step is to take the region proposals suggested by the RPN and apply RoI pooling. Region of Interest pooling is used to transform the RPN's variable-sized region proposals into fixed-size feature maps that may be fed into the network's subsequent layers.
   b. **Feature Extraction:** The RoI-pooled feature maps are fed into the CNN backbone (the same one used in the RPN for feature extraction) to extract meaningful features that capture object-specific information. It draws hierarchical features from regional proposals.

   c. **Fully Connected Layers:** The RoI-pooled and feature-extracted regions then pass through a series of fully connected layers. These layers are responsible for object classification and bounding box regression tasks.

      1. **Object Classification:** The network predicts class probabilities for each region proposal, indicating the possibility that the proposal contains an object of a specific class. The classification is carried out by combining the features retrieved from the region proposal with the shared weights of the CNN backbone.
      2. **Bounding Box Regression:** In addition to class probabilities, The network predicts bounding box adjustments for each region proposal. These adjustments refine the position and size of the bounding box of the region proposal, making it more accurate and aligned with the actual object boundaries.
         The first layer is a softmax layer of N+1 output parameters (N is the number of class labels and background) that predicts the objects in the region proposal. The second layer is a bounding box regression layer that has 4* N output parameters. This layer regresses the bounding box location of the object in the image.

Methodology:

Dataset preparation:

1. Extracted object bounding boxes with labels.
2. Divided the dataset into training (80%), validation(10%) and testing(10%).

Hyperparameters:

1. **Optimizer:** AdamW
2. **Learning rate**: 0.0001
3. **Batch size:** 4
4. **Epoch :** 30

Evaluation metrics and results:

Metrics:

1. **mAP (Mean Average Precision):** The mean Average Precision (mAP) is a metric that measures the accuracy of a model in identifying and classifying objects within an image. It combines precision and recall to give a comprehensive measure of a model's performance. It is popular for measuring the accuracy of object detection models like- Faster RCNN, Yolo, SSD.

Result:

The model achieved 66.08% which is good for detection of objects in traffic images. The training loss of the model was 0.0783.