Real-Time Traffic Sign Recognition for Advanced driver-assistance systems

Authors:
Orish Jindal
Hatim Alhazmi
Vaibhav Bishi

SCAN ME

An Introduction to the project

One of the critical milestones in the way of our dream of autonomous cars is automatic traffic sign recognition. Even before achieving that dream, the traffic sign recognition can assist the driver greatly as sometimes it is difficult for a human to recognize the traffic sign in conditions like rain, fog, lousy windshield visibility, or human factors like drowsiness, near-sightedness, etc.

With this motivation in mind, our project aims to implement real-time traffic sign detection and recognition. The first task to solve this problem would be to detect where the traffic sign is located in the picture taken by the dashboard camera, which is done by using a famous architecture called Faster R-CNN. The next task will be to classify the portion of the image that represents a traffic sign which we performed using a custom CNN architecture.

Data description

The dataset used in this project is the German Traffic Sign Recognition/Detection Benchmark which is a set of two separate datasets meant for traffic sign classification and detection.

Detection: GTSDB is a single-image detection assessment dataset consisting of 900 full street-view images in total belonging to 43 classes. The images are stored in the PPM format, while the corresponding annotations for the bounding boxes are given in CSV files that contain the filename, the traffic signs' region of interest (RoI) and labels.

Classification: GTSRB is a multi-class, single-image dataset consisting of 43 different classes with 26,727 traffic-sign images in total. We divided the dataset into training and test sets in 4:1 ratio. Moreover, data augmentation is performed on the images in the form of random horizontal flipping and perspective change for introducing variation in the dataset.

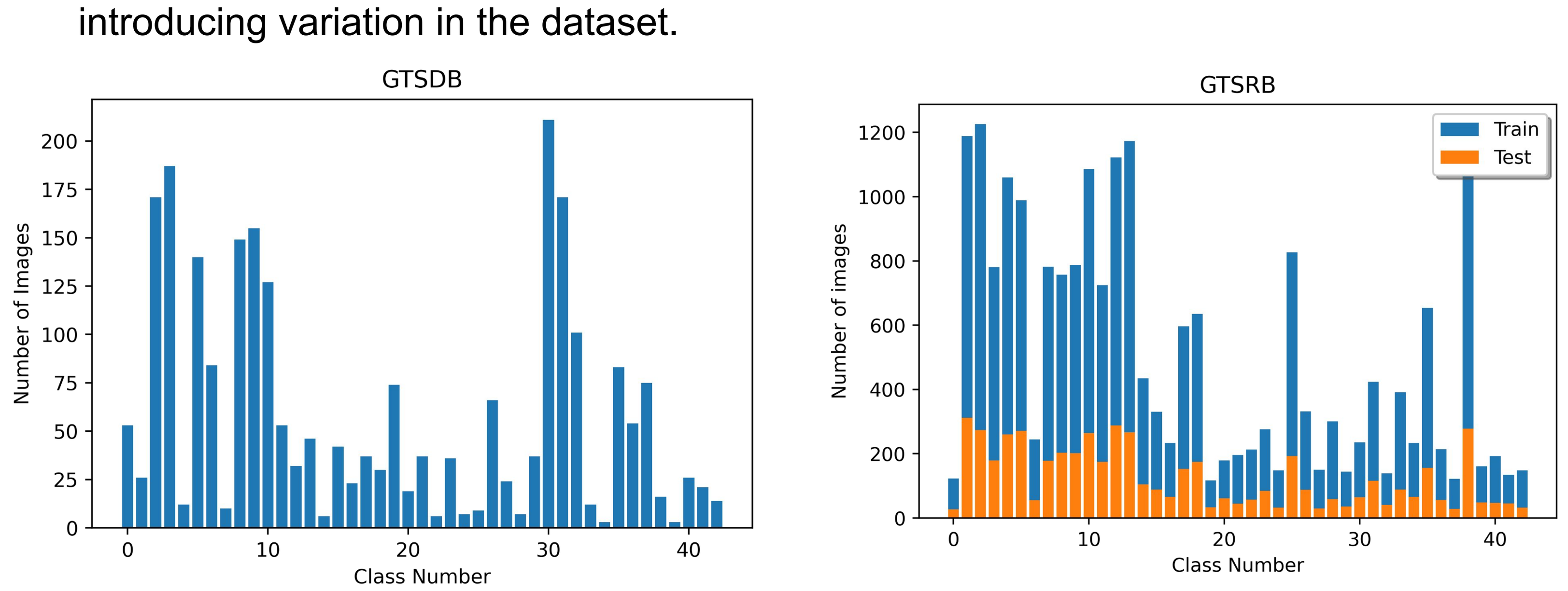


Figure 1: Datasets used.

Framework and Model description

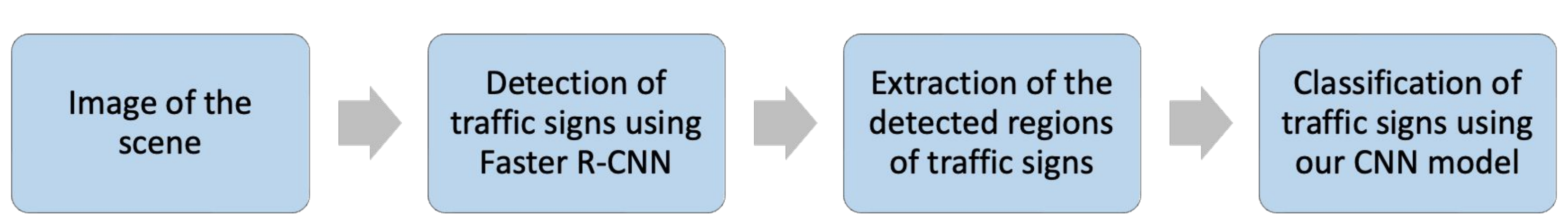


Figure 2: Framework for the project.

What is CNN:
We used Convolutional Neural Networks (CNN) which is a class of artificial neural network that analyzes visual data. It has multiple layers with three main categories: an input layer, an output layer, and a hidden layer that includes multiple convolutional layers, pooling layers, fully connected layers, and normalization layers. CNN performs well in visual data as they can learn more discriminative features in an image.

Detection: Carving-out the region of interest from the whole picture captured by the dashboard camera

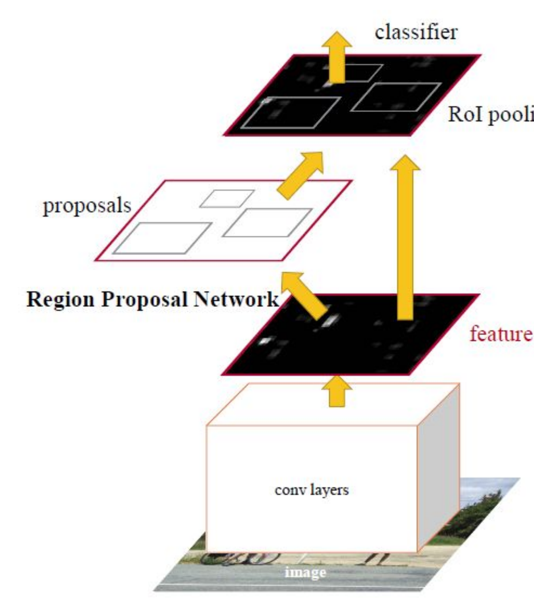


Figure 3: Faster R-CNN.

We used the Faster R-CNN [2] model with a pretrained ResNet50 backbone network which is widely used for the object detection tasks, the architecture of which is shown on Figure 3. It is specifically known for its **Region Proposal Network (RPN)** which is very effective in detecting the region where the object of our interest is located. It is a fully-connected convolutional network that simultaneously predicts object bounds and objectness scores at each position. The RPN is trained end-to-end to generate high-quality region proposals, which are used by Faster R-CNN for detection.

Classification: Recognizing the label of the traffic sign that is present in the RoI given as an output of Faster R-CNN

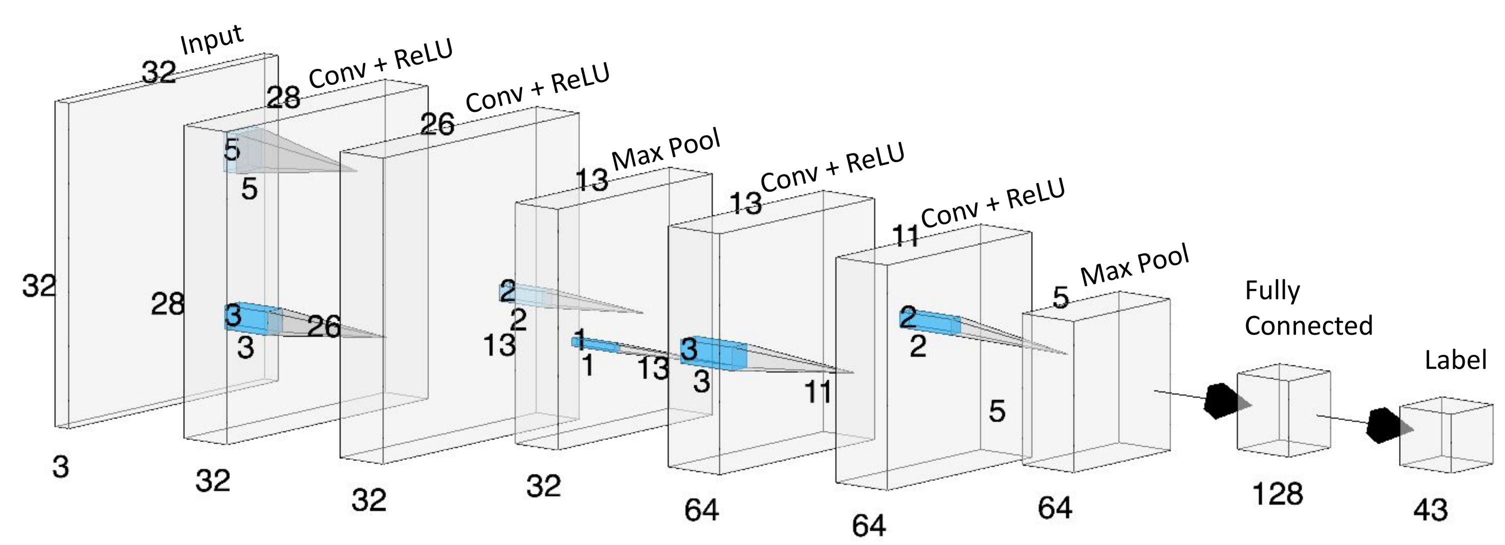


Figure 4: Our CNN Model Architecture.

The architecture of the CNN model we implemented [1] is elaborated in figure 4. We made use of Cross-entropy loss and Adam optimizer while training our CNN model. With some experimentation, we tuned our hyperparameters as:
Learning rate: 0.001, **Scheduler with gamma:** 0.97, **Batch size:** 128, **Epochs:** 50

Performance and Discussion

We evaluated the performance of our custom CNN model and the Faster R-CNN model. Figure 5 shows the performance of Faster R-CNN for correct detection of the traffic signs' regions. Our classification model is able to achieve an accuracy of 98.42% on the traffic sign test data as shown in Figure 6. We tested both the models on all the 741 full street-view images for classifying the detected traffic signs and our model outperforms Faster R-CNN as shown in the form of a bar graph in Figure 7.

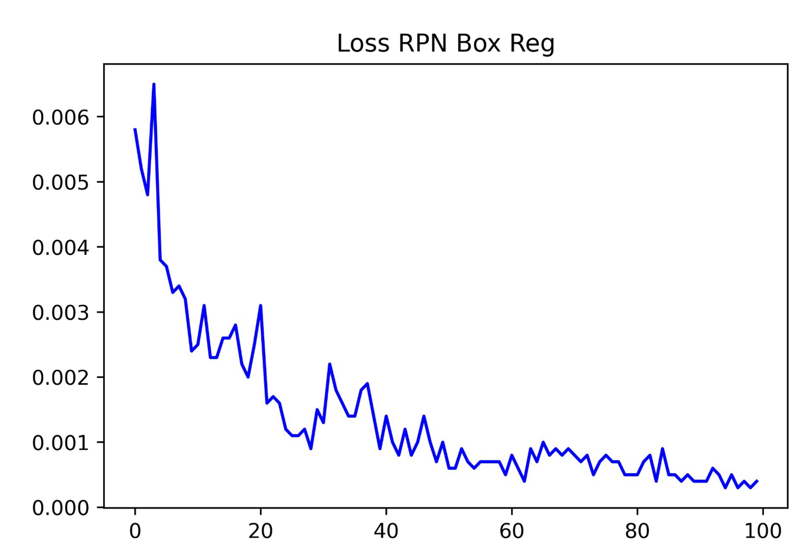


Figure 5: Loss for RoI as bounding box for detection

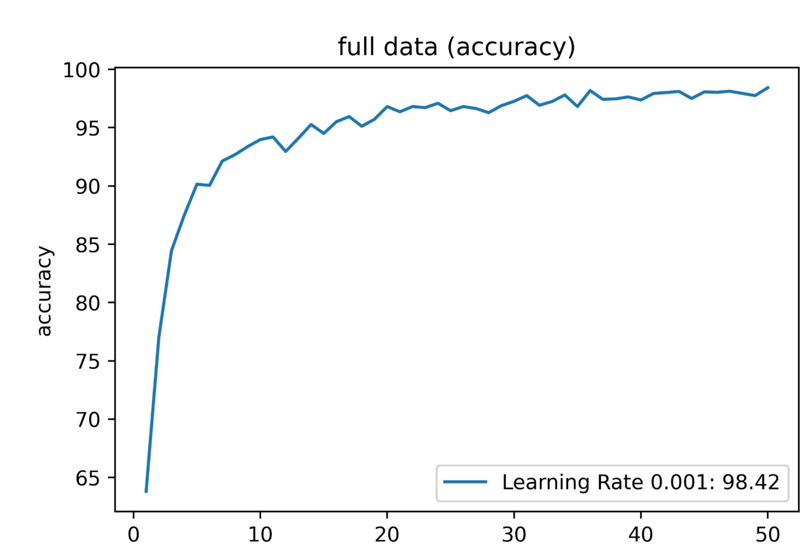


Figure 6: Number of epochs vs Accuracy for classification

Faster R-CNN can be used for the complete task of detection as well as classification of the objects. Although its detection part was fairly accurate (the architecture responsible for drawing the bounding box around the RoI), we found that its classification of the objects was not performing very well on full images because of the low training sample size found in GTSDB. Hence, we decided to use a separate model for the classification, which was trained on GTSRB dataset with some relevant data augmentations. This significantly improved the classification accuracy since the sample training size now is much larger.

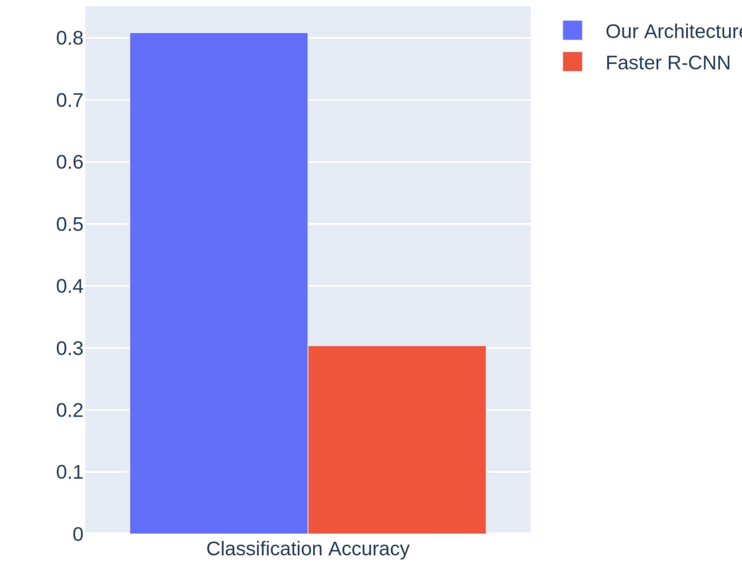
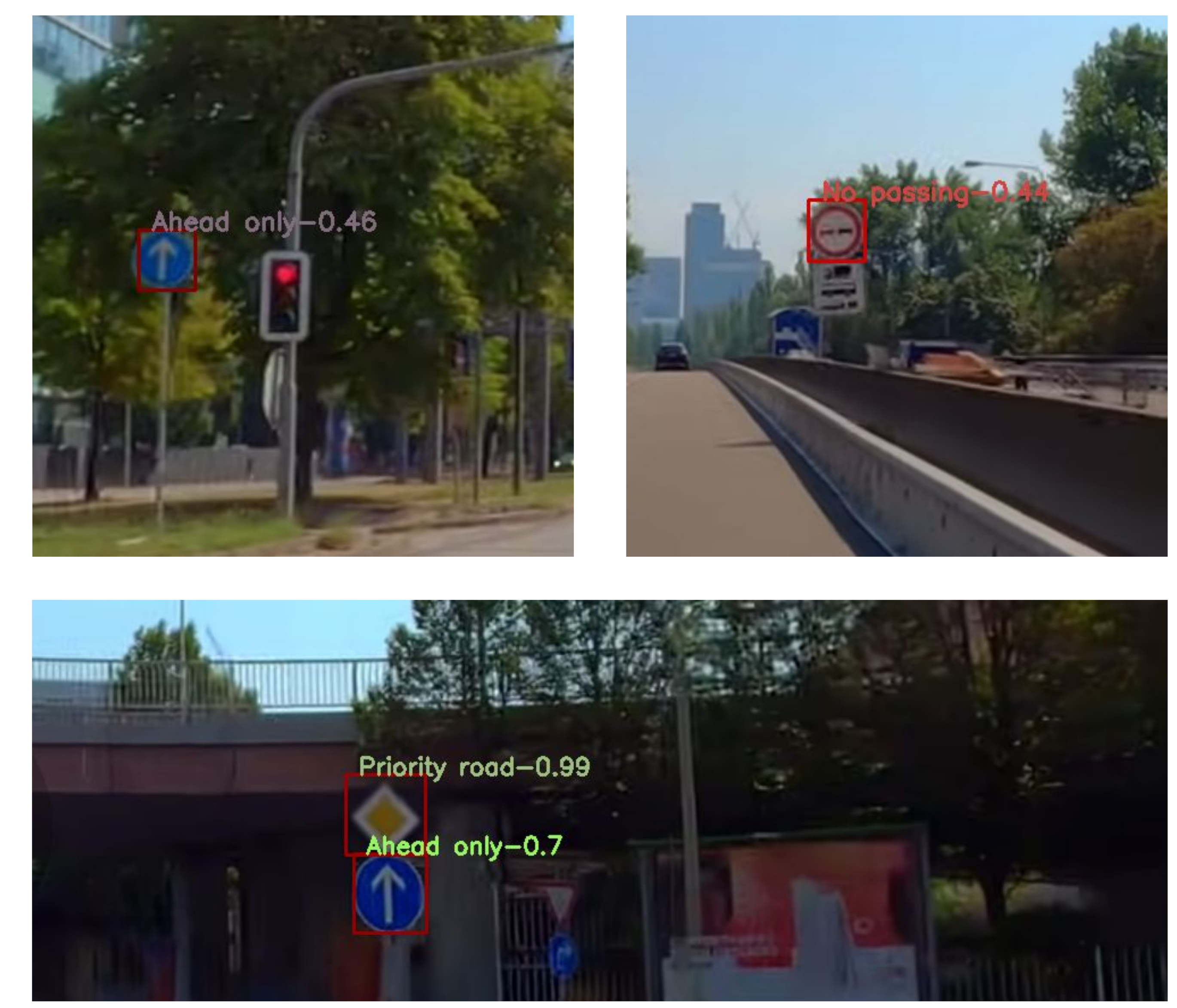


Figure 7: Classification Accuracy of Faster R-CNN vs Our CNN Mode tested on 900 images.

Performance in real world:

We tested our model on some random youtube videos showing German city-roads from a camera mounted on the car. The screenshots of the videos after our model was implemented are shown below.



Conclusion and Future Work

We observe that our model is susceptible to detecting certain objects that look very similar to a traffic sign but in fact are not. We can incorporate a new class into the dataset which would cater to this specific issue such that our model is accurately able to distinguish between a real sign and a similar looking object. This can avoid any possible false alarms and avoid any unnecessary disruption during travel on the road.

In future, this project can be extended further to integrate it with another CNN model that is detecting the driver's dizziness or lack of attention on the road through an interior camera. This conjunction will let us enhance driver safety as the driver can be fairly alerted by a sound system if there is an 'immediate action requiring' traffic sign on the road and the driver has not paid attention to it. This can further be integrated with the vehicle's braking system so that the car be brought to a stop if the driver's reaction time is slow so as to avoid harm to the driver and others on the road.

References

[1] Jayant Mishra and Sachin Goyal. An effective automatic traffic sign classification and recognition deep convolutional networks. Multimedia Tools and Applications, pages 1–20, 2022.

[2] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 1 June 2017, doi: 10.1109/TPAMI.2016.2577031.