ENPM 673: Perception for Autonomous Robots
Project 6

Shreya Gummadi
Sneha Ganesh Nayak
Ghanem Jamal Eddine

18 May 2019

Traffic Sign Recognition

In this project we do Traffic Sign Recognition. This is achieved in two steps:

- Traffic Sign Detection
- Traffic Sign Classification

Traffic Sign Detection

In this part of the project, the traffic sign is detected in the frames and a bounding box is returned which is further used for classifying the sign.
In order to detect the sign, the images are first processed. The following steps are carried out in the pre-processing:

1. The images are resized to (600,600). This is done in order to reduce the processing time of the pipeline.

2. Images are split into the 3 channels i.e. R, G and B.

3. Each channel is denoised using cv2.medianBlur with a kernel of size of 5.

4. Next the channels are normalized using cv2.normalize with the cv2.NORM_MINMAX parameter.

5. The intensities are normalized as suggested in the project description.
   For red channel: max(0;min(R -B;R - G)/R + G + B)
   For blue channel: max(0;min(B-R;B - G)/R + G + B)

Once the preprocessing is done. The traffic sign can be detected by Maximally Stable Extremal Regions or MSER.

Detection using MSER

MSER is run on the normalized intensity binary images. The MSER is run with maximum and minimum area parameters to ensure that noise detected is less. The regions with red and blue concentrations are detected and returned. These areas have maximum probability of having a traffic sign.
However, there are unwanted parts detected as well. To overcome this, HSV thresholding is also done.

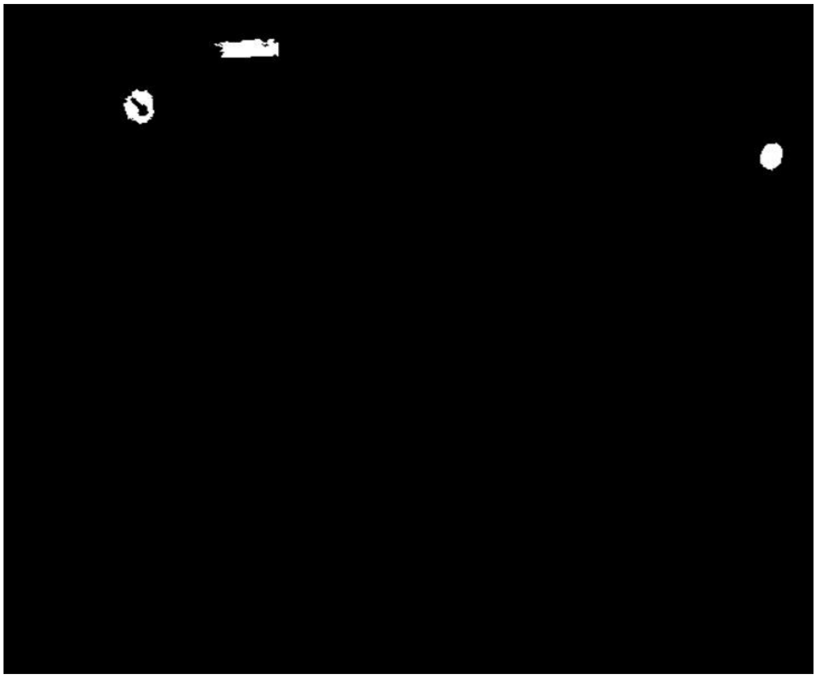

Fig 1. Blue regions detected using MSER

Detection using HSV

In this part the following image processing is done:

1. The image is converted into HSV format and denoised using cv2.medianBlur.

2. The image is normalised by dividing it by 255.

Once this is done, only the saturation and value channels are taken into consideration. We determined a threshold for the s and v channels for blue and red detection.

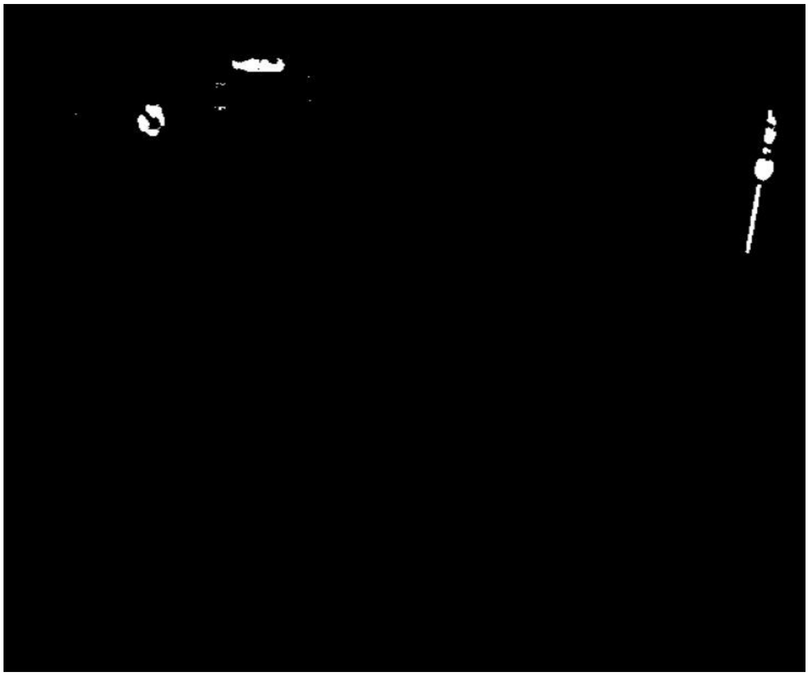

Fig 2. Blue regions detected using HSV

Combining MSER and HSV

We combined the red and blue regions detected by mser and by hsv thresholding to get rid of as much noise as we could and thus detect the traffic signs better.

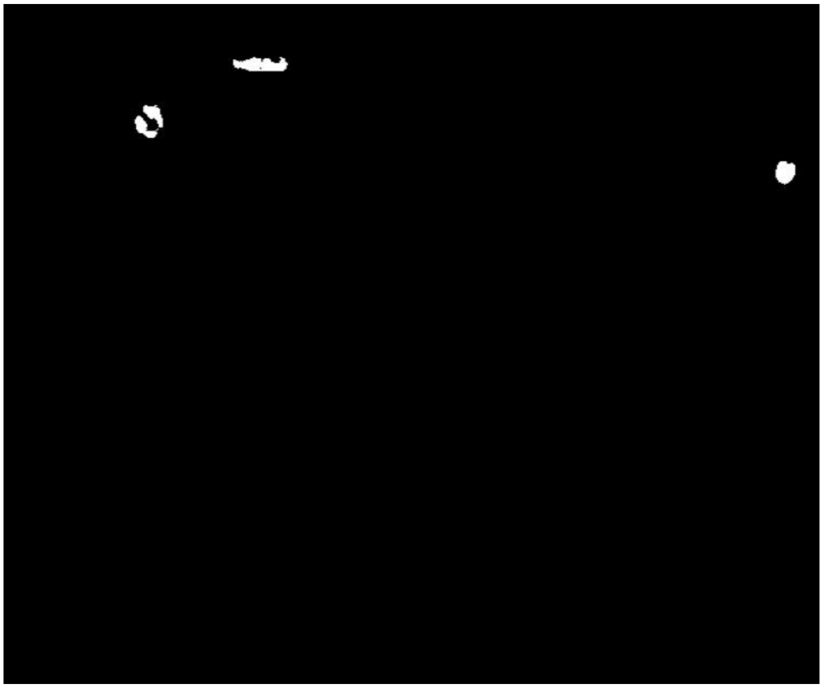

Fig 3. Combined MSER and HSV blue region

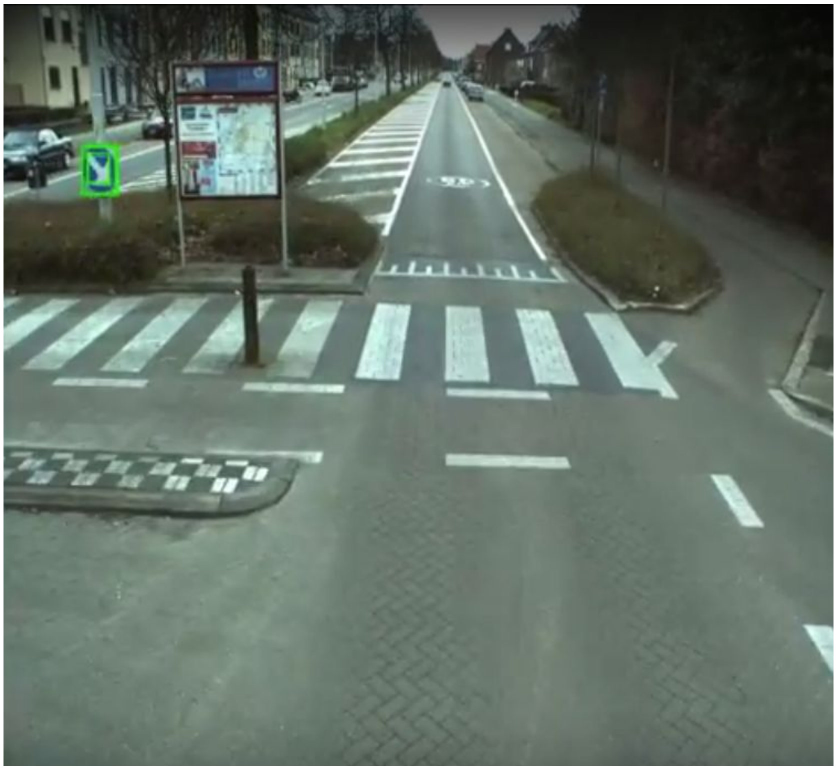

Figure 4. Blue sign detection

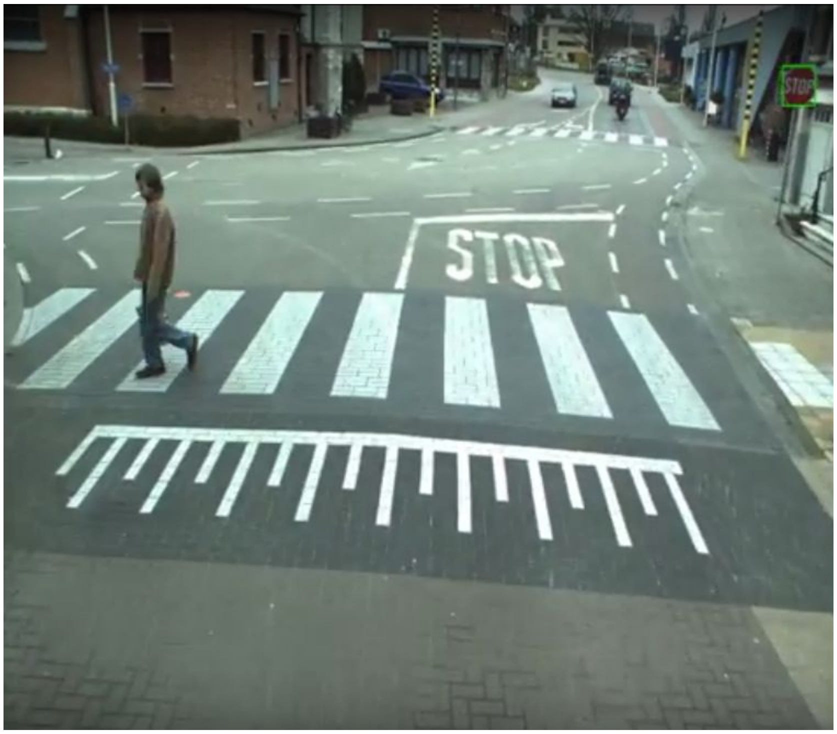

Figure 5. Red sign detection

Traffic Sign Classification

HOG Descriptor Extraction:

Pre-Processing:

- The images are first gray-scaled
- A median filter is applied, using cv2.medianBlur.
- After this the image is resized to 64x64, on which the HOGs are then computed.

Computation of the HOG's:

This is divided into 2 steps:

- In the first step the Histogram of Oriented Gradients from each image belonging to the traffic Signs numbered 45, 21, 38, 35, 17, 1, 14 and 19, are extracted. The images used for extraction of HOG's in this step are from "training_selected" which is a subset of the images in "Training", which consists of images belonging to traffic signs numbered 45, 21, 38, 35, 17, 1, 14 and 19.
- In the second step, the Histogram of Oriented Gradients from each image belonging to all the negatives are extracted. The images used for extraction of HOG's in this step are from "training_negavtives" and "neg"; "training_negavtives" is a subset of the images in "Training", which consists of images belonging to traffic signs excluding the traffic signs numbered 45, 21, 38, 35, 17, 1, 14 and 19 , whereas 'neg' consists of images that are false positives in the Detection phase.

Multi-Class SVM:

- Here a multi class SVM is trained with the HOG's that are extracted in the previous step.
- The HOG's computed are trained with a particular label, depending on the Traffic Sign number. The images used for training in this step are from "training_selected" which is a subset of the images in "Training", which consists of images belonging to traffic signs numbered 45, 21, 38, 35, 17, 1, 14 and 19.
- The HOG's of the negatives are also trained using the label 'negatives'.
- The classifier is then fitted to the model, which it must learn from.

Combination of Detection and Classification

Here, the detection and classification phases work hand in hand to correctly detect and classify the traffic signs. The steps are as follows:

- After the classifier is trained, the part of the image that is detected as a traffic sign is cropped and resized into a 64x64 image and the HOG descriptors are extracted.
- These descriptor are then checked against the classifier to predict the label that it is classified under. By predicting, we can determine the image from the training set that best matches the detected traffic sign.

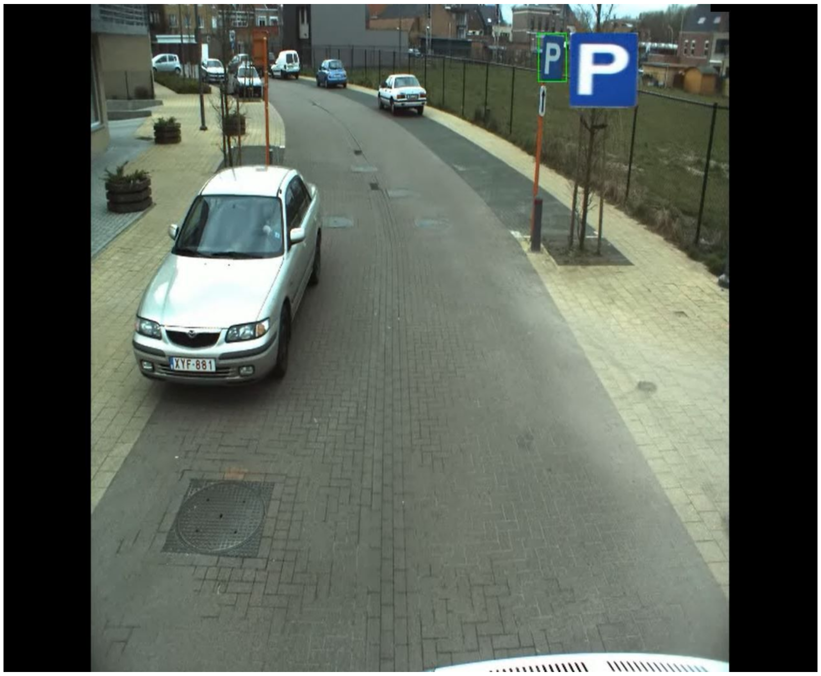

Fig 6. Sign Detection and Classification (Traffic Sign-45)

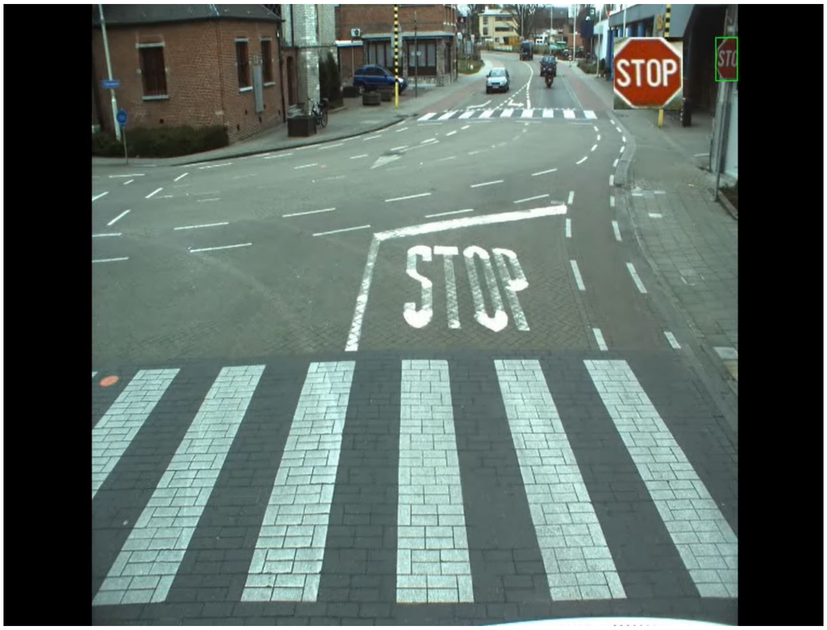

Fig 7. Sign Detection and Classification (Traffic Sign-21)

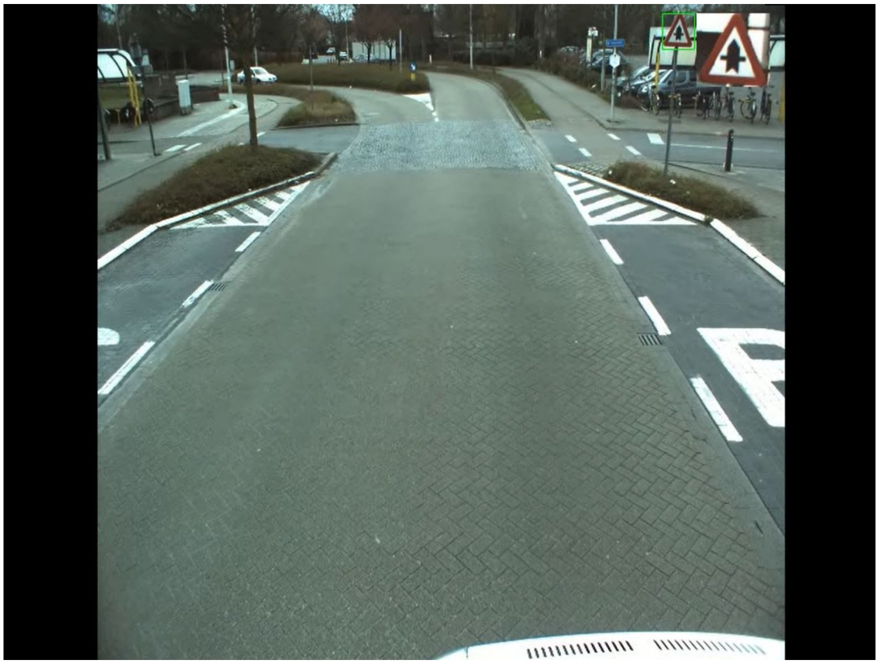

Fig 8. Sign Detection and Classification (Traffic Sign-17)

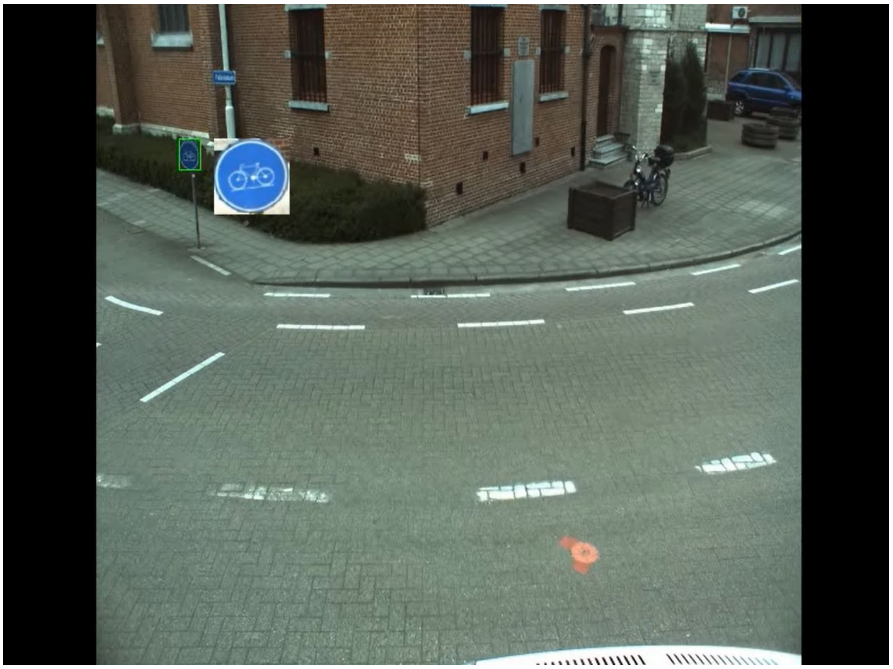

Fig 9. Sign Detection and Classification (Traffic Sign - 38)

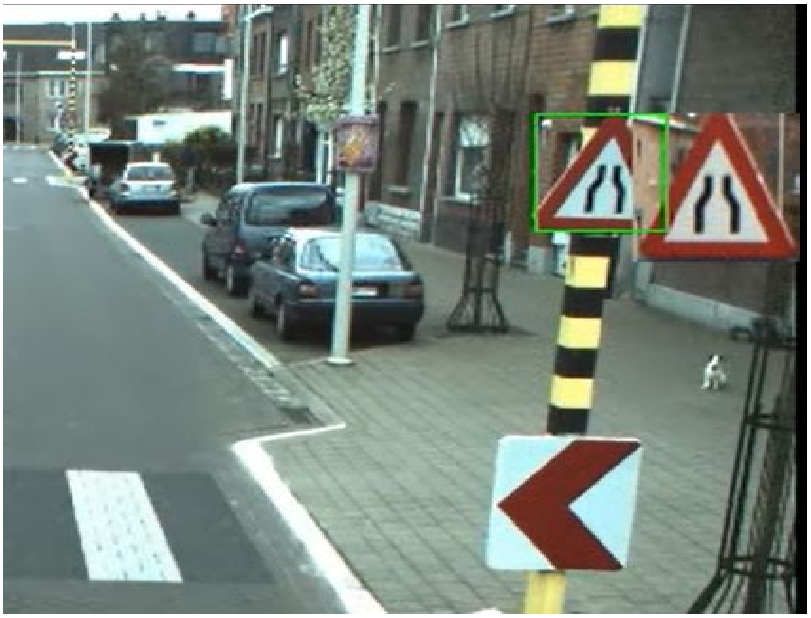

Fig 10. Sign Detection and Classification (Traffic Sign - 14)

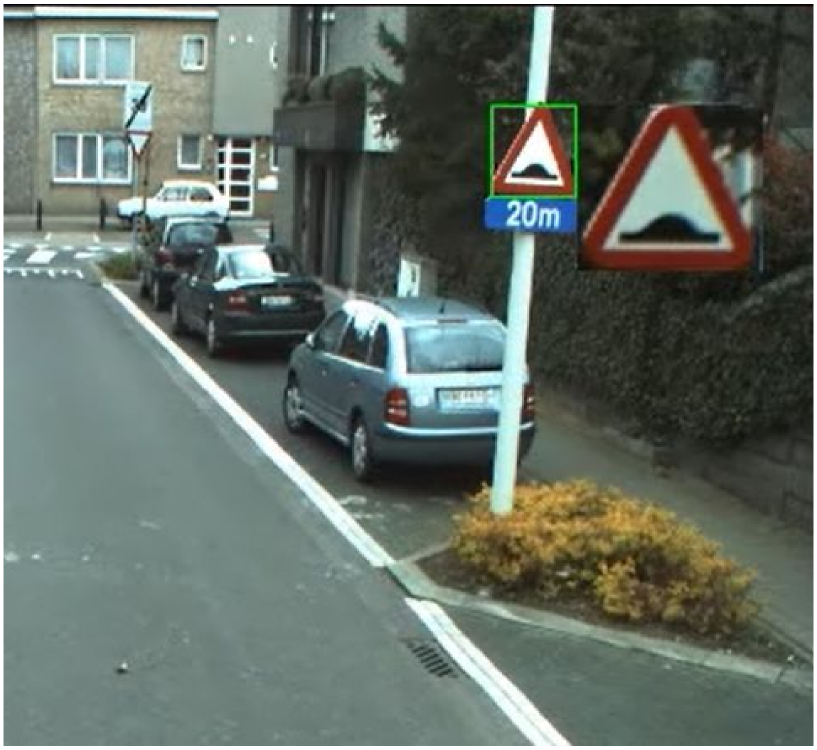

Fig 11. Sign Detection and Classification (Traffic Sign - 1)

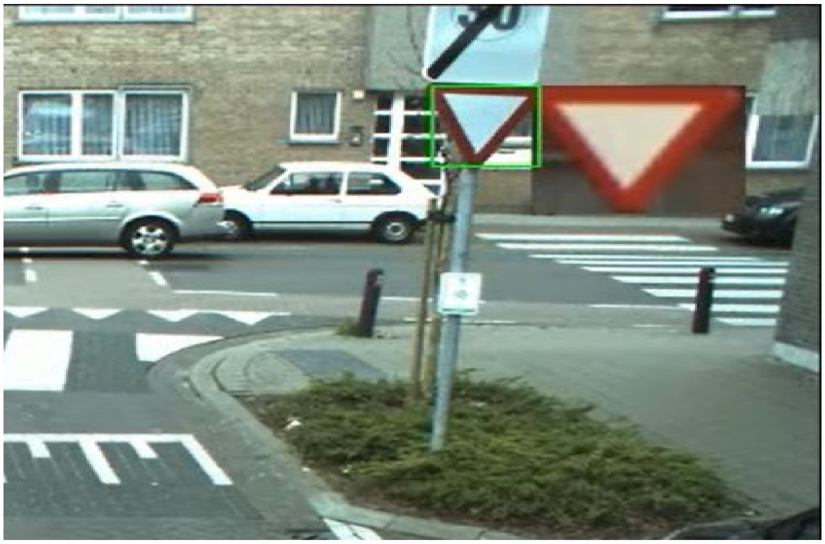

Fig 12. Sign Detection and Classification (Traffic Sign - 19)

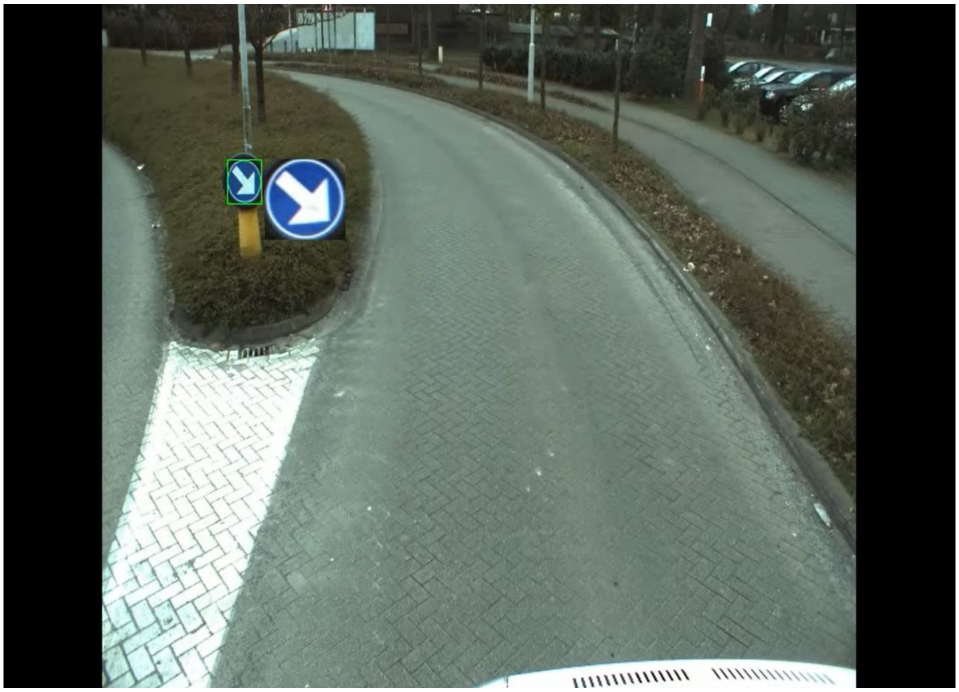

Fig 12. Sign Detection and Classification (Traffic Sign - 35)

Conclusion

Except for a few frames, the traffic signs seem to be detected and classified accurately.