Project 2 Report:
Lane Tracking with Road Sign Recognition

1. Road Sign Recognition

1.1. Design

In project 2, we need to recognize a STOP road sign and a SCHOOL road sign using a second laptop and communicate successfully that a particular road sign has been detected from the second laptop to the first laptop. Then, implement a vehicle control algorithm to stop the vehicle at the stop sign for 2 seconds and then run at a high speed till the school sign is detected. Once recognizing the school sign, the vehicle slows down till it detects the stop sign again.

In the road sign recognition part, we use a machine learning method to do it. The detector is based on the Viola-Jones algorithm. In this Cascade object detector, it can detect object categories whose aspect ratio does not vary significantly, like road signs. The detector detects objects in images by sliding a window over the image. The detector then uses a cascade classifier to decide whether the window contains the objects of interest. The size of the window varies to detect objects at different scales, but its aspect ratio remains fixed. The detector is very sensitive to out-of-plane rotation, because the aspect ratio changes for most 3-D objects. Thus, we need to train a detector for each orientation of the object. Also, a Cascade detector can only detect one type of object. In our case we want to detect two different traffic signs, we will need to train two Cascade object detector separately.

Here, we use Image Labeler in MATLAB to label objects of interest with bounding boxes to create positive images. And when we are training stop-sign detector, we include negative images that contain school zone sign and vice versa. In the real test, sometimes we found the other objects would also be detected as road signs by mistake. And we took lots of the pictures for those objects from different orientation and include them into the corresponding negative image folder. We found that the quality of the positive images also affects the results much, so we abandoned distortion images and just use clear images which include the whole sign. Also, in project 2, two different signs need to be detected, so two detectors need to be trained and run at the same time.

Figure 1 Road signs

In project 2, we also used ‘snapshot’ to obtains pictures from the webcam camera and set the resolution as ‘320x240’. We used ‘size’ to get the size of the image and ‘rgb2gray’ to convert the image into a gray scale. The same endless loop is implemented in the code as project 1.

1.2. Results

We used 106 positive images and 214 negative images for STOP sign, 147 positive images and 309 negative images for SCHOOL sign. We set false alarm rate as 0.3 and number of cascade stages as 10.

The results are fine. There are no wrong detections of other objects and could detect two signs correctly.

Figure 2 Sign detection result

2. Communications

2.1. Design

In the project 2, the detection of the traffic signs was executed using one laptop and the vehicle motion in the lane was controlled using second laptop. As in the project, the vehicle was supposed to follow a set of motion control commands on the traffic sign detection, there was a need for setting up a fast and reliable communication protocol between the two laptops. UDP and TCP were the two communication protocol options available for us for setting up communication between the two MATLAB sessions. Here, we had more importance for fast communication than high reliability of the communication, so we used UDP instead of TCP protocol.

In this communication part, we create UDP communication in MATLAB. By creating and customizing a UDP object on each MATLAB session, we can configure each object to communicate to the other through data reading and writing commands.

In our code, we execute ‘delete(instrfindall)’ at the start of the code because the UDP object will not be shut down even if the code is stopped, and the port will be occupied when we try to run the code again. ‘fprintf’ is used to give the information to the second laptop and ‘fscanf’ is used to receive the information from the first laptop. We used ‘udpB.BytesAvailable’ to find whether there is new message available in the buffer and ‘flushinput (udpB)’ to clear the

input buffer after reading the message to make sure that latest message will be read in the next loop.

2.2. Results

In the project, the code ran well and there was no problem with the communication between the two laptops. Just notice that the IP address would change every time we connect the university network.

```
delete(instrfindall)
ipA = '198.21.196.20';
portA=9090;

ipB='198.21.239.44';
portB=9091;

udpA=udp(ipB,portB,'LocalPort',portA);
fopen(udpA);
```

Figure 3 UDP communication in MATLAB

3. Controls

3.1. Design

In the project 2, we need to decide when to stop before the STOP sign and when to reduce speed before the SCHOOL sign. Here, we set counters to determine how many times a sign has been detected. Also, run time for each loop considered to determine the number of times a sign is detected within a particular time period. The threshold for STOP sign is 45 times in 2 seconds. The threshold for SCHOOL sign is 12 times in 0.6 seconds. To avoid cumulative errors, filter parameters for STOP sign refreshed when detector detected school sign and vice versa. After meeting these conditions, the first laptop would send the corresponding message. The following figure is the logic for the controller. Once the message from laptop A is received, we need to judge whether it is a signal for STOP sign or SCHOOL sign. If it is a STOP sign. We set the vehicle stop for 2 seconds then apply a higher speed. If it is a SCHOOL sign, we set the speed as a slower speed.

In fact, what matters in this part is the threshold for image counters and runtime. Because this is correlated with the vehicle speed. We need to ensure the vehicle speed is the same every time we test the vehicle. However, the fact is the battery level is different every time. And this will cause the difference in the vehicle speed although the control signal is the same.

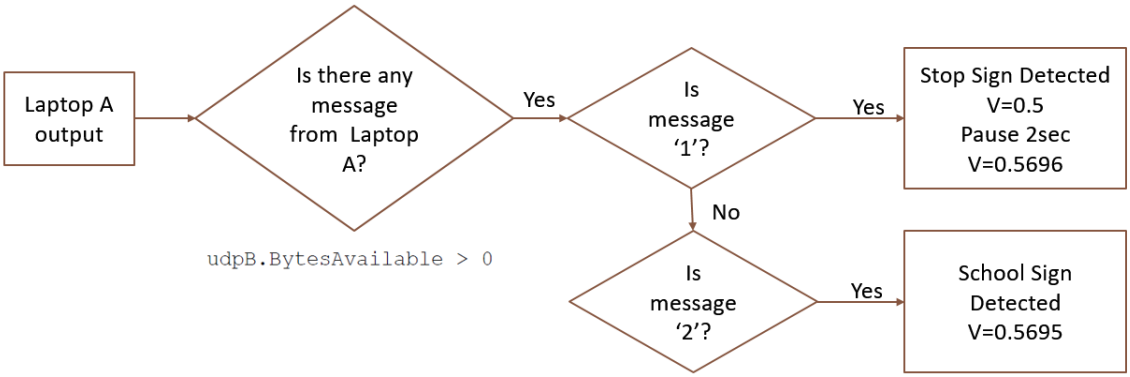

Figure 4 Control logic

3.2. Results

The vehicle could stop just before the STOP sign and reduce the speed before the SCHOOL sign. The codes worked well during the whole test.

4. Conclusion and Discussion

As a conclusion, the vehicle could maintain the lane for all the laps. The camera could detect the school sign and stop sign. Vehicle halted on the detection of STOP sign and then accelerated to a high speed value. On detection of SCHOOL sign, the vehicle dropped its speed to a lower value. As I mentioned before, the most important thing in this project 2 is the control of the speed. Suitable speed would ensure a better performance of the lane detection and road sign detection. Also, the distance stopped before the STOP sign is also related to the vehicle speed. We should tune the speed and the threshold for counters of images and run time at the same time, or the vehicle could stop for twice before the STOP sign or stop after passing the STOP sign. For the recognition part, the thing is to include enough negative images to train the detector. This would ensure there are no other objects detected as a sign.