SPEC PRO

Speed and Congestion Detection using Proximity Sensors

Prateesh Goyal 120050013
Pratyaksh Sharma 120050019
Viplov Jain 120050084
Ramprakash Krishnan 120050083

Project Overview

- The aim of the project is to build an automatic system for detection of overspeeding vehicle, their vehicle number and using it for automatic challan
- Use IR proximity sensors embedded in road to detect speed of vehicles and if the vehicle was speeding then take an image of the speeding vehicle and detect the number on the number plate using image processing techniques.
- Once the number is obtained, the number along with relevant details such as speed, time , location etc is to be emailed to the concerned party. There should be a GUI to display all the challan recorded by the system along with relevant details and picture of speeding vehicle.

Requirements

- Detect the speed of any vehicle passing over sensors
- Use image processing techniques coupled with OCR (Optical Character Recognition) to identify the number plate
- Send an email to the concerned party regarding the overspeeding vehicle. It should also include relevant details about the incident
- Build a GUI to display the complete history of challans

Project Plan

The task was divided into 4 parts and each person worked independently to achieve the final goal-

1. Image Processing to detect the number of vehicle - Prateesh
2. Use OCR and integrate all the components of the system into one - Pratyaksh
3. Speed detection using IR sensors and GUI - Ramprakash
4. Send an email - Viplov

Weekly progress was made and roughly same amount of work was done each week. Each of the processes were improved iteratively.

Block Diagram of Prototype

Innovation and Challenges

We use cheap proximity sensors and cheap low-resolution webcam (low shutter speed) to bring down the cost of the system.

The initial challenge is to correctly capture the image of the vehicle whose speeding is detected. Challenging because camera is fixed, and processing/transmission delays need to be taken into account.

Another of the main challenges is to identify the number plate from a captured image (blurred due to moving vehicle).

Tasks Completed

Capturing the image of moving vehicle correctly:

Triggering the camera after overspeeding of the vehicle has been detected might be too late: vehicle would have already passed the frame of view.

We continuously capture images at 10ms intervals, and then by using the knowledge about delays (calculated/tuned very precisely by hand), we are able to pick the correct image which shows the vehicle number plate clearly (though blurred due to motion).

Tasks Completed

Identifying the number plate from the image captured:

We use multiple image processing steps: colored patch detection, adaptive cropping, Otsu’s algorithm for thresholding (binarizing the image), and Gaussian denoising before passing the image to Tesseract OCR trained to detect single lines of digits.

In our tests with multiple number plates, different speeds of moving vehicle, and different lighting conditions, we were able to achieve ~80% accuracy on the end-to-end detection.

Description

- Proximity Sensors - The sensors detect if there is a vehicle above them or not. The sensors feed the data to TIVA board continuously
- TIVA Board - On basis of input from Digital IR sensors, TIVA board deduces the speed of vehicles and if detects over speeding it signals the laptop to take the image
- Camera - The camera is continuously taking images and only some recent images are stored. On receiving signal from TIVA, we first pick appropriate image and
- Laptop -We apply image processing techniques to get the number plate region, then auto binarize the image and feed it to tesseract to obtain the vehicle number. One we have the number mail the relevant details to the concerned party. The history of challans can also be viewed by the use of GUI.

Review and Testing

The experiment was conducted many times at different firebird speeds (2 speed modes) and with different number plates(012, 677, 517). Precisely each was run for 5 times.

Parameters - Speed, Different Number Plate

Estimation of speed and vehicle number was tested.

Speed was correctly estimated in all the cases, this was verified by comparing the output and manually measuring speed by running the robot over a long distance and using stopwatch to calculate speed.

Observations

For each of the configuration below table indicates number of correct number recognition out of 5 tries

	0 1 2	6 7 7	5 1 7
Low Speed	4	4	5
High Speed	3	4	4

Total accuracy in number detection = (24/30)*100 % = 80%

Re-usability features

All our code (across python, TIVA C and Java) adheres to respective language guidelines and is cross-platform.

The modules for GUI, TIVA control, various steps in image processing, serial input client on PC, are split into independent files with the functionality of each specified clearly.

The hardware used is the standard TIVA board with IR proximity sensors (range 3 cm): allows for easy upgradation of equipment (upgraded sensor etc.).

Future Enhancements

- The image processing part of the project can be improved so that it doesn't require, a green identifier to detect the number plate
- .At the moment Tesseract(ocr) works on standard text only, and the tesseract can be specifically trained to recognize number plate
- The accuracy of the number detection algorithm can be further increased by applying advanced image processing techniques.
- Debluring of input image can be further improved