Evolution in Electrical and Electronic Engineering Vol. 2 No. 2 (2021) 703-708

EEEE

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/eeee
e-ISSN : 2756-8458

Vehicle Speed Estimation System by using Matlab

Che Mohd Safwan Najmi Che Manan[1], Wan Mahani Hafizah Wan Mahmud[1,2]*

[1]Department of Electrical Engineering, Faculty of Electrical and Electronics Engineering,
Universiti Tun Hussein Onn Malaysia, Batu Pahat, 86400, Johor, MALAYSIA

[2]Advanced Medical Imaging,
Universiti Tun Hussein Onn, Batu Pahat, 86400, Johor, MALAYSIA

*Corresponding Author Designation

DOI: https://doi.org/10.30880/eeee.2021.02.02.084
Received 04 July 2021; Accepted 01 August 2021; Available online 30 October 2021

Abstract: One of the biggest issues inside the UTHM community was the vehicle over speeding issue. As a result of this problem, all car drivers within inside the UTHM area exceed the speed limit. Besides that, The Vehicle Speed Estimation System by MATLAB simulation software's was created to address this issue. This system was built with the ESP32 Camera Module, and the video pictures were processed with the MATLAB software. Any vehicle exceeding the speed restriction within the UTHM area will be recorded and captured by this camera recorder system. The MATLAB will display the offender vehicle's exact speed as proof that it is exceeding the speed restriction in that location. This project will aid in the reduction of excessive speeding inside the Universiti Tun Hussein Onn Malaysia area. This project is also being created with the goal of lowering the accident rate in the UTHM area and generating a more controlled and effective speed movement. As a result, the entire UTHM community will take this matter more seriously and drive within the speed limit within the UTHM area, as well as build a new better driving environment within the UTHM area by paying closer attention to the car speedometer.

Keywords: Vehicle Speed, Speed Estimation System, MATLAB

1. Introduction

The Vehicle Speed Estimation technology was one of the most popular system that can help many things in many situations. This technology has been introduced since past 4 years but all of these technologies and system have been upgraded to become better and effective for the user. The current speed detection was handheld by the police department that allow them to check the car speed and been control by the police security department for any evidence on the scene. As the population grows, so does the number of vehicles on the road. As a result, traffic surveillance systems based on video capture

*Corresponding author: wanmahani@uthm.edu.my

and processing to evaluate speed have become critical concerns in preventing accidents. In Malaysia, road accidents are one of the leading causes of death and injury. Aside from that, road accidents are becoming more common, with a high percentage of individuals dying as a result [1]-[2].

According to the World Health Organization, around 1 million people are killed in traffic accidents each year (WHO). This is a severe issue that needs to be addressed in future planning [3]-[4]. This technology will detect over speeding vehicles on UTHM speeding area. The ESP32 Camera module will monitor and record all the video footage on the speeding area. The MATLAB software will do the speed calculation system. By doing this project, it will have many benefits to the driver to become more alert while driving, thus will lead to decrement road accident percentage [5].

By the end of this project, we should have a Vehicle Speed Estimation system that can calculate and estimate the vehicle's speed by using MATLAB software and Video Image Processing technique. It is hoped that from this project, all drivers especially in UTHM may become more alert with the speed limit, thus reducing the over speed cases which may leads to road accidents.

2. Materials and Methods

This study was carried out using MATLAB®. MATLAB is a programming environment for engineers and scientists developed by Math Works. Next, the ESP32 Cam Module and his device can monitor all the vehicle that cross into it in the video streaming session. Based on that video streaming footage, the offender video was processed on the MATLAB as a video image processing and the MATLAB will produce the actual speed of the offender vehicle.

This section goes through how to create a vehicle speed estimation algorithm in depth. The algorithm for video image processing was developed using MATLAB. The created algorithm is tested and simulated using a ready input video. Figure 1 shows how to create a vehicle speed estimation algorithm in depth. The algorithm for video image processing was developed using MATLAB. The created algorithm is tested and simulated using a ready input video. In order to compare the output, different videos are used. For distinct input photos, however, separate zones of interest must be defined.

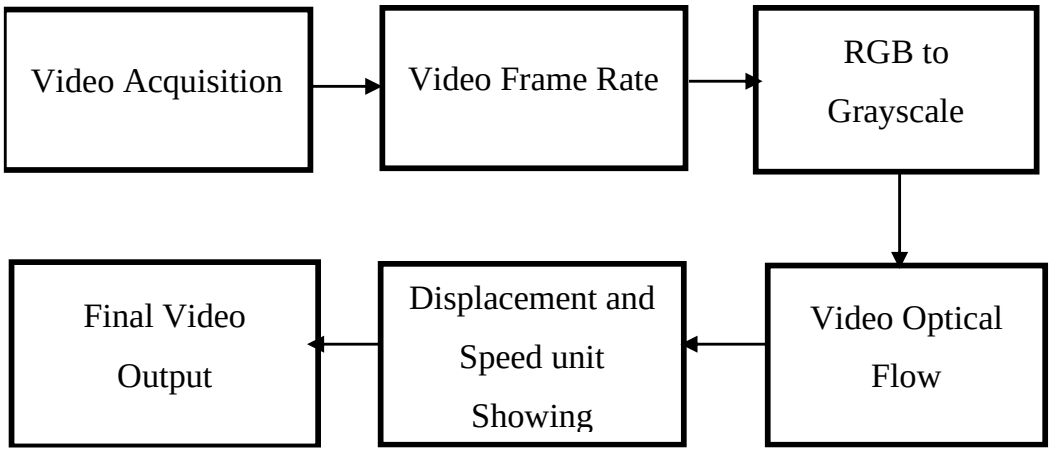

Figure 1: Block Diagram of Algorithm Development

2.1 Video Acquisition

The Video Image Processing in MATLAB uses existing or imported video as input. The video can be read using the MATLAB command videoFReader, and the input parameters will be shown and captured in the MATLAB workspace. The photos can be seen in MATLAB using the videoFReader command.

```
videoFReader = vision.VideoFileReader('Psm1.mp4');
v = VideoReader('Psm1.mp4');
%read video
videolink = 'C:\Program Files\MATLAB\R2016a\Psm1_Video';
paravideo = VideoReader(videolink);
t=paravideo.FrameRate;
```

2.2 Video Frame Rate

It will display the frame rate of any video in the MATLAB can be read by using the video frame rate command in the MATLAB. It will display the actual frame rate of the video. The ESP32 will monitoring for the recording and capture the video footage on the scene. The ESP32 also will conduct a live streaming video on the scene.

```
while hasFrame(v)
video = readFrame(v);
end
whos video
end
```

2.3 RGB to Grayscale

The four types of images are binary, grayscale, RGB, and indexed images. RGB images are color or images in which each pixel has a red, green, and blue color combination. The pixel value in a grayscale image varies from 0 to 255, with shades of grey in between. The photographs must be converted from RGB to grayscale before processing. The rgb2gray function is used as a result. Using this function, 24-bit RGB values are converted to 8-bit Grayscale values.

```
%OrVideoReader = vision.VideoFileReader(  videolink, ....
                                        %'ImageColorspace', 'RGB',...
                %'VideoOutputDataType', 'unit8');
```

2.4 Video Optical Flow

It is a pattern that depicts how objects, surfaces, and edges appear to move. The observer's relative motion to the scene can also help to construct the imagined scene. It is also possible to establish a distribution of apparent velocities of any movement and a brightness pattern in a picture.

```
%create optical flow
opticalFlow.Method = 'Lucas-Kanade';
converter = vision.ImageDataTypeConverter;
OpticalFlow = vision.OpticalFlow('ReferenceFrameDelay', 3);
OpticalFlow.OutputValue = 'Horizontal and vertical components in complex form';
shapeInserter = vision.ShapeInserter('Shape', 'lines', 'BorderColor', 'Custom',
'CustomBorderColor', 255);
```

2.5 Displacement and Speed unit showing

The speed (velocity) is the rate of change of distance (displacement) with time. The speed in this system will showing in km/h.

```
disp(cen); %show...
disp(size(cen));%show....
speed_str = [num2str(a),'Km/h'];
```

2.6 Final Output Video

The video image processing will show the final output in MATLAB. It shows the vehicle tracking and speed showing on that video processing in km/h.

```
%release video reader and writer
release(OrVideo);
release(OfVideo);
release(BwVideo1);
release(BwVideo);
release(ReVideo);
release(OrVideoReader);
```

3. Results and Discussion

Figure 2 shows the original video that been import to the MATLAB as the input video for the Video Image Processing. The video depicts the car's movement and motion within the UTHM speeding zone. The ESP32 Cam Module will record all video footage on the scene, which will then be processed in the MATLAB software to determine the car's true speed.

Figure 2: The original image that export from the ESP32 live streaming video footage

Figure 3 shows the displaying the calculated speed from the MATLAB software. The MATLAB Video Image Processing will show the actual speed of the car in Km/h. The real speed of the car was 46.7 km/h, according to this result. The live streaming video of the ESP32 Cam Module was used to record the entire video. The ESP32 Cam Module's live streaming video may also record and capture every scene that passes across it.

Figure 3: Displaying the calculate speed from the MATLAB software

Figures 2 and 3 show the original image export as well as the MATLAB and the final speed output from the MATLAB. The ESP32 Cam Module will record all video footage on the scene, which will then be analysed in MATLAB software to determine the car's true speed. The exact speed of the car in Kilometres per hour will be displayed using MATLAB Video Image Processing. The real speed of the car was 46.7 km/h, according to this result. The ESP32 Cam Module's live streaming video may also record and capture every scene that passes across it.

The speed data comparison between MATLAB and the real speed from the car speedometer is shown in Table 1. The speed range for the first test was 10-20 km/h, and the MATLAB speed was 18.28 km/h, whereas the real speed from the real car speedometer was 20.00 km/h. Then, at a speed of 20-30 km/h, the MATLAB speed was 28.11 km/h, but the true speed from the car speedometer was 30.00 km/h. Finally, the speed between 30 and 40 km/h was tested, and the MATLAB result was 38.80 km/h, whereas the real-time speedometer on the car indicated 40.00 km/h. Although there were a few technical challenges with the device, all of the results from the comparison of MATLAB speed output and real car speedometer speed were close to success.

Table 1: The speed data comparison between MATLAB and Real speed from the car speedometer

No	Speed Test (Km/h)	MATLAB Final Speed Result (Km/h)	Real speed from car speedometer (Km/h)
1	10-20	18.28	20.00
2	20-30	28.11	30.00
3	30-40	38.80	40.00

4. Conclusion

In this study, the traffic conditions and scene on the road in the UTHM community area were used. Any vehicle's speed can be tracked using the developed algorithm. The proposed algorithm accomplishes the objectives. The system has a flaw in that it can recognize but cannot accurately track their speed. When an image or video is captured, a number of factors might influence the quality and noise of the image or video. One of them is when there is bad weather, such as rain, which causes the picture and video recording results to blur and degrade in quality. Besides that, one of the factors that affected the video or image recording result was the darkness. This issue will cause the outcome to be overly dark, and the vehicle's image will not be as clear as it would be on a normal sunny day. Using these techniques, vehicle detection can be improved and errors reduced.

Acknowledgement

The authors would like to thank the Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia for its support.

References

[1] M. Rath, "Smart Traffic Management System for Traffic Control using Automated Mechanical and Electronic Devices," IOP Conf. Ser. Mater. Sci. Eng., vol. 377, no. 1, 2018, doi: 10.1088/1757-899X/377/1/012201.

[2] Gupte, S.; Masaud, O.; Martin, F,K,R; Papanikolopuolos, N,P. Detection and classification of vehicles, IEEE Trans, Intell, Transp, Syst, 2002, 3, p. 37-47.

[3] Moazzam, Md.G., Haque, M.R. and Uddin, M.S. (2019) Image-Based Vehicle Speed Estimation. Journal of Computer and Communications, 7, p. 1-5.

[4] Different Approaches for Motion Estimation", E.F.(4th-6th June 2009), International Conference on control, automation, communication and energy conservation -2009.

[5] Lazaros Grammatikopoulos, George Karras, Elli Petsa, E .F.(November, 2005),"Automatic Estimation of Vehicle, Speed from Uncalibrated Video Sequences" International Symposium on Modern Technologies, Education & Professional Practice in Geodesy and related fields, Sofia, p.03 – 04