

# IMPLEMENTATION OF LANE TRACKING BY USING IMAGE PROCESSING TECHNIQUES IN DEVELOPED PROTOTYPE AUTONOMOUS VEHICLE

Sertap Kamçı, Dogukan Aksu and Muhammed Ali Aydin

Computer Engineering Department, Istanbul University-Cerrahpasa,  
Istanbul, Turkey

## **ABSTRACT**

*Today, unmanned vehicle technologies are developing in parallel with increasing interest in technological developments. These developments aim to improve people's quality of life. Transportation, which is a part of human life, has taken its share from this developing technology. With the development of artificial intelligence, it is aimed to provide the necessary assistance to the driver in transportation and to provide ease of driving. This development has been increased with ADAS (Advanced Driver Assistance Systems) in vehicles, but it is not possible to experience a completely driverless and comfortable road. With all these demands and conditions, autonomous vehicles have quickly attracted attention. While ADAS is a warning system, all accident risks that may arise from the driver rather than the warning to the driver in autonomous vehicles are minimized by the vehicle.*

*In this paper, we present an autonomous vehicle prototype that follows lanes via image processing techniques, which are a major part of autonomous vehicle technology. Autonomous movement capability is provided by using various image processing algorithms such as canny edge detection, Sobel filter, etc. We implemented and tested these algorithms on the vehicle. The vehicle detected and followed the determined lanes. By that way, it went to the destination successfully.*

## **KEYWORDS**

*Autonomous Vehicle, Lane Detection, Image Processing, HSV Color, RGB Color, Canny Edge Detection, DC Motor, Region of Interest (ROI), Vanishing Point, Sobel Filter*

## **1. INTRODUCTION**

In order for the vehicles to be able to watch safely in a series, the part of the vehicle path separated by lines is called lane. A good lane monitoring enhances the enjoyment of going on the road. Solving heavy traffic and solving other disruptions can become more streamlined with drivers taking some responsibility. If an autonomous vehicle is not traveling, the driver should: After providing the road and traffic control without deciding to pass the vehicle, first to check the mirrors according to the direction of departure, to check whether there is a vehicle in the blind spot and wait for the vehicles to pass, to give the appropriate signal to the direction of exit, not to get too close to the passing vehicle and after passing the vehicle to pass immediately, to overtake in cases where overtaking is prohibited, service vehicles and buses minibuses briefly public transport vehicles to pay attention to the superiority of the transition. In addition to all these responsibilities, it is also not necessary to test how dangerous overtaking is wrong. It is also important to not leave the current lane and center the lane. Because in this case the accident is inevitable and people in other vehicles have added to the risk of having an accident. But thanks to autonomous vehicles, the passengers pass a comfortable journey without the need of the driver's responsibilities.

The algorithm of autonomous vehicles allows the protection of the lane. Gets an idea with the algorithm written about the lanes to be traveled according to the route of the vehicle and determines the path accordingly. The vehicle has an idea about the lanes to be traveled according to the route of the vehicle and determines its path accordingly.

The algorithm in the study is on lane tracking. The algorithm is designed to track lanes. First of all, with the operation of the autonomous vehicle, the image on the road is taken through the camera. The received image is a video image and is divided into image frames called frames. Using specific image processing algorithms, the lanes in these frames are detected. When the determination of the lane, the vehicle is given the instructions to continue straight to the vehicle according to the angles determined by the lane and to turn right or left.

This paper is organized as follows: Section 2 contains relevant studies and provides a comprehensive overview of the different detection methods that are used to the project. The functions and models used in Chapter 3 are presented. The experience gained in Chapter 4 is explained and the project output is given. In Chapter 5, the operating logic of the car following the lane is explained with a flow diagram. The results obtained are given in Chapter 6.

## **2. RELATED WORKS**

The interest in autonomous vehicles has been increasing rapidly in recent years. In this paper, image processing algorithms for lane detection and tracking are mentioned.

There are several techniques for performing lane detection [1,2]. First of all, the image processing methods used in the project should be examined to find the lines. Thereafter, lines should be identified from image frames where image processing techniques are applied. After this step, the ROI between the lines should be found. In addition, a virtual line is drawn in the middle of these lines to make the path look like a curve. With this curve, the bend of the way is determined by finding the angle. With this angle, the vehicle has an angle of movement and the vehicle moves in a straight, right-handed, left-handed direction according to the specified angle.

In this paper, we examine the related studies in three stages. These steps are: image processing techniques, detecting lane lines and finding ROI.

### **2.1. Image Processing Techniques**

In the design of the algorithm, firstly the mutation from RGB color space to HSV color space is made. According to some studies, it is more convenient to use HSV color space when we want to differentiate an object of a certain color in any computer vision/image processing application [3]. In addition, the HSV color space provides clarity of the colors in the image. HUE is also used to distinguish colors more clearly.

Color images contain more information than gray level images. For this reason, in some related studies, the edge detection process for color images has been examined. Grayscale method is one of these methods. The cost of detecting ROI in the image was reduced by the Grayscale image conversation method and the process was accelerated [4,5].

### **2.2. Determination of Lane Lines**

Canny edge detection, Hough transformation and Sobel filter methods are the methods used to find the lane lines.

Canny edge detection is a method of edge detection. The lanes are lines and edges and the purpose is to discover optimal edge detection. In this way, the edges in the image as well as unwanted noise called pixels to eliminate the full image recognition is done [6]. Hough transformation method is a method that finds and shows shapes. Since the lane lines have a shape, lane lines can be found by the Hough transform method [7].

Sobel filter method is a separate method used to find the edge.

It is seen in all the studies obtained the use of the edge detection algorithm in images with a grayscale colour space is closer to the edge information in the actual image.

### **2.3. Finding Region of Interest (ROI)**

The area of the lane is called the ROI. While some algorithms have an ROI up to the point where the lines can be detected by combining the lane lines in the image, some algorithms have methods for finding the ROI based on vanishing point detection techniques [8].

There are two common methods for the detection of ROI. Firstly, left and right lane lines are found. In the first method, the lane lines are stretched and intersect at one point. A triangular region is formed. The area within the triangular region up to the region where the lane are detected forms a rectangular region. This field returns the ROI. The deviation difference of the angles and the obtained region is calculated and the direction of the path is calculated [9,10].

Another method is to determine the ROI by determining the skyline. Horizon line is the line where the earth globe and sky intersect when viewed in nature. With the determination of the horizon line, the lane lines are also combined and the intersecting area shows the ROI.

## **3. DESIGN OF THE LANE FOLLOWING AUTONOMOUS CAR**

The design part of the car is divided into two parts. These are the movement of the software and the hardware of the vehicle. The software field is algorithm design and coding.

### **3.1. Software**

The designed code has a layered architecture. There is a python file with a main structure named Main. Methods in other Python files are imported and run by calling the methods respectively. The lane lines must be determined in the algorithm of the car following the lane. For this, lane detection algorithm is used. Image processing techniques are used in the lane detection algorithm. These image processing techniques are described in the below.

Some libraries used for image processing are: Numpy, OpenCV, Math, RPI.GPIO. The Numpy library was used for sequencing. This array operations; dividing the video video taken from the camera into frames and keeping them in a sequence, keeping the curves of the lane lines in the frames in one sequence, keeping the inclination angles of the road in the forward path in a sequence. The Math library is used to perform mathematical operations in the algorithm. The OpenCV library focuses on image processing, video capture and analysis, including features such as face detection and object detection. This library was used to detect objects and lanes in the image. The RPI.GPIO library is used to manage the pins on the raspberry pi and to locate the wheels, sensors and batteries that we have connected in the algorithm.

Frames with original RGB color space were converted to frames with HSV color space and color saturation was gained. These frames have been rendered grayscale image processing so that an image with saturated color characteristics are converted to shades of gray more clearly. If this was

done without converting from the RGB color space to the HSV color space, the frame converted to grayscale would receive shades of gray with less transition and the details of the clear shapes on the frame would not be visible.

The Canny edge detection method was used on the grayscale frame and the lines including the detailed lines in the frame were reached. After these image processing methods, the lanes in the original path obtained were detected. The lanes were determined by Hough Transform and Sobel filter methods [11,12].

For the car to go in the middle of the lanes, there is a variable called center lane in the code, and the centerline is a line that centers the two lanes [13]. Here you must first find the right and left lanes, and the mainline of code written for the detection of the left lane is as follows.

```
left_fit_average = np.average (left_fit, axis = 0)
left_line = make_coordinates (image, left_fit_average)
```

The following mainline of code is written for Centerline.

```
lines = cv2.HoughLinesP (frame, 2, np.pi / 180, 100, np.array ([]), minLength = 40,
maxLineGap = 5)
```

For the centerline, the following line of code is written.

```
center_lines = average_slope_intercept (frame, lines)
```

The algorithm written for ROI detection is called here and the calculation of the ROI is done in this algorithm. The ROI-finding algorithm maintains the intersecting and polygon definition region on the image. This is the area between the lane and the horizon.

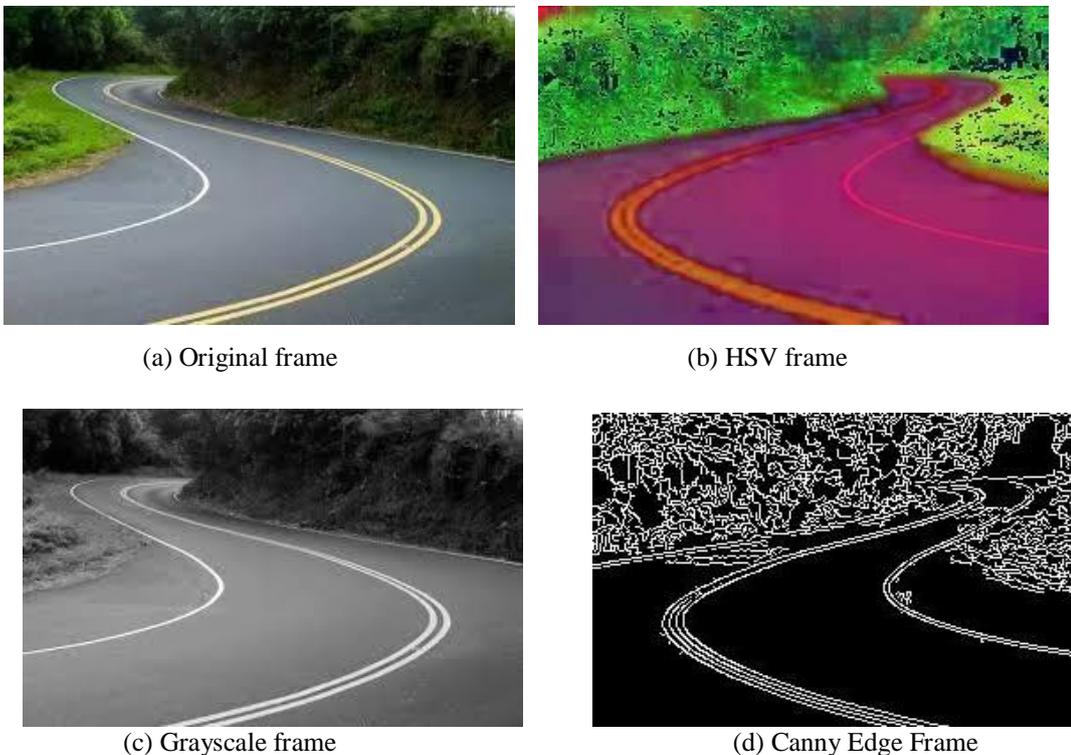


Figure 1. Steps of image processing methods on a path

In Figure 1 (a), a path with RGB color space was converted to HSV color space as in Figure 1 (b). Color saturation increases with HSV color space. Then, the image was processed as shown in Figure 1 (c) using Gaussian Blur and Gray Scale methods. Gaussian Blur allows us to adjust the degree of blurring, creating effects ranging from slightly softening the image to covering the entire image with thick fog. Gaussian Blur is also a successful filter to protect the edges in the image [14]. Figure 1 (d) is used to understand the line of the lanes on the road by using Canny edge detection and Hough Transform methods. As a result of the calculations, the lane lines are perceived as shown. In Figure 2-a, a re- converted image frame is given to the RGB image. In Figure 2-b, an image frame that is not converted to RGB color space is given.

The mainlines of code and descriptions of the steps in Figure 1 are given below. The main line of code for conversion from RGB color space to HSV color space is as follows. Cv2 in this code represents the OpenCV library. COLOR\_BGR2HSV is the command to switch from RGB color space to HSV color space.

```
hsv = cv2.cvtColor (crop, cv2.COLOR_BGR2HSV)
```

The command line below is the command to convert the frame from HSV color space to Grayscale.

```
gray = cv2.cvtColor (image, cv2. COLOR_HSV2GRAY)
```

In order to find the lines in the grayscale frame with the following lines of code, the Canny edge detection image processing method is used.

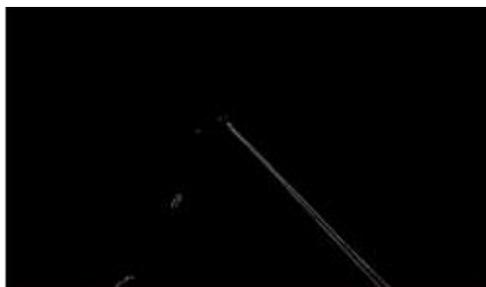
```
canny_imag A = canny (frame)  
cropped_imag A = region_of_interest (canny_imag A)
```



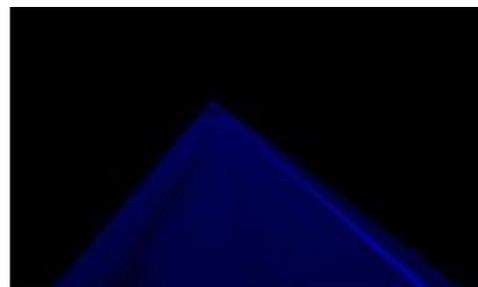
(a) Original path image



(b) Detected Lane from image



(c) Detected interested lane



(d) ROI of the path image

Figure 2. Detection of lanes in a sample path

In Figure 2 (a), the algorithm is written to detect the road where the vehicle is located and this lane is shown based on the nearest lane. In Figure 2 (b), the ROI of the lane was found. In Figure 2-c, the algorithm is written to detect the path where the vehicle is located and this lane is shown based on the nearest lane. Figure 2-d shows the ROI of the lane.

### 3.2. The Movement of the Vehicle

We want the vehicle to the center the lanes before start moving. In the code, the function that generates the virtual line is written to center. Then you need to follow the lane where it is located. For tracking of the lane, the ROI has the angle of rotation of the lane. When driving straight ahead, the vehicle is commanded to continue straight ahead. We used DC motors for the movement of the vehicle.

```
GPIO.output(Motor1A,GPIO.LOW)
```

Above left is the left motor stop command. In the same way, the right engine must be stopped for a right turn. This stop command may be slightly different, if the vehicle is not going to turn completely and rotating at a different degree than the angle of 180, the motor in the direction of rotation is decelerated and the motor in the other direction is accelerated. This is the logic of the vehicle's progress and rotation relative to the road.

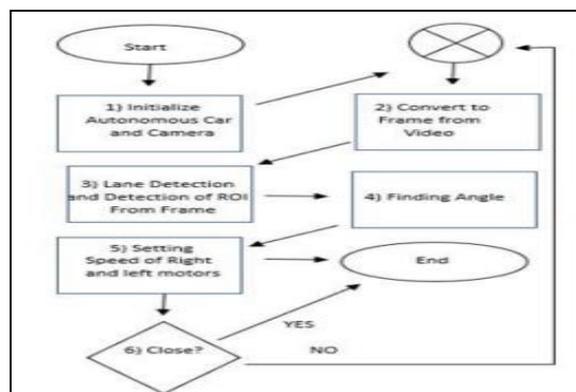


Figure 3. Image of autonomous vehicle prototype

## 4. EXPERIMENTS

A great experience has been gained in the construction of this project. Before the integration of the code to the vehicle, the code was checked by remote movement with the computer. Then the code is integrated into the vehicle. The desired result is to obtain and process the image from the camera and to find the lanes correctly. The vehicle should also be able to move smoothly with commands sent from raspberry pi without deviating from its path. The pre-integration and post-integration operation of the code was verified and successfully completed.

## 5. THE PROPOSED METHOD



```
Algorithm. Pseudocode for the proposed method
1: // autonomous car and camera setting
2: while !CloseCar() do
3:     convert to frame from video
4:     detect to lane and ROI from frame
5:     find to angle of load
6:     set to speed of motor
7: end while
```

Figure 4. The proposed method: The Flowchart and The Pseudocode.

The figure above shows the algorithm-based operation designed in the period from the start of the vehicle to the closing of the vehicle.

When the vehicle is started, the camera also operates (Scheme1) and instant frames (frames) are taken from the camera (Scheme2). Lanes are detected by image processing methods and ROI is found (Scheme3). When the ROI is found, it is understood how the vehicle should be directed to protect the lane and there is an angle for this (Scheme 4). The angle found is adjusted by the speed of the right and left motors (Figure 5). As long as the vehicle is not switched off, the camera operates and returns to Scheme2, resulting in a flow loop. If the vehicle is switched off (Scheme6 takes yes), the flow switches to Scheme7 and the autonomous vehicle and camera are switched off.

## 6. CONCLUSION

In this paper, an autonomous vehicle prototype that detect lanes via image processing techniques, which are a major part of autonomous vehicle technology is presented. Some image processing algorithms such as canny edge detection, Sobel filter, etc. are used to provide autonomous movement capability. They were implemented and tested on our prototype vehicle. The prototype vehicle detected and followed the determined lanes successfully and it reached the destination. As a future work, we are planning to use generative neural networks, deep learning, and various machine learning algorithms to detect lane and traffic signs together.

## ACKNOWLEDGEMENTS

The project in this paper is presented as a bachelor thesis and it is supported by Istanbul University-Cerrahpasa Scientific Research Projects Commission as project number 32561 and “FBA-2019-33004”. We would like to thank Istanbul University-Cerrahpasa Scientific Research Projects Unit for their support.

## REFERENCES

- [1] David C. Andrade, Felipe Bueno, Felipe Franco, Rodrigo A. Silva, Student Member, IEEE, João H. Neme, Erick Margraf, Student Member, IEEE, William Omoto, Felipe Farinelli, Angelo M. Tusset, Sergio Okida, Max M. Santos, Senior Member, IEEE, “vel Strategy for Road Lane Detection and Tracking based on Vehicle’s Forward Monocular Camera”, IEEE Transactions on Intelligent Transportation Systems, September 2018.
- [2] Aharon bar hillel, Ronen Lerner, Dan Levi, Guy Raz, “Recent progress in road and lane detection: A survey”, Machine Vision and Applications 2011.
- [3] Gurjashan Singh Pannu, Mohammad Dawud Ansari, Pritha Gupta,” Design and Implementation of Autonomous Car using Raspberry Pi”, International Journal of Computer Applicationa, 9 March 2015.

- [4] Peerawat Mongkonyong, Chaiwat Nuthong, Supakorn Siddhichai, Masaki Yamakita, "Lane detection using Randomized Hough Transform", 8th TSME-International Conference on Mechanical Engineering, January 2018.
- [5] A.A.M. Assidiq, Othman O. Khalifa, Md. Rafiqul Islam, Sheroz Khan, "Real time lane detection for autonomous vehicles", IEEE Xplore, 2008.
- [6] Anjali Goel, "Lane Detection Techniques - A Review", International Journal of Computer Science and Mobile Computing, February 2014.
- [7] Ze Wang , Weiqiang Ren "LaneNet: Real-Time Lane Detection Networks for Autonomous Driving", ArXIV, 4 July 2018
- [8] CHANG YUAN, HUI CHEN , JU LIU , DI ZHU , AND YANYAN XU, "Robust Lane Detection for Complicated Road Environment Based on Normal Map", IEEE, July 27th 2018.
- [9] Sertap Kamçı, Aylin Yazıcı, Bilge Kamberoğlu, Doğukan Aksu, Muhammed Ali Aydın, "A Survey of Lane Detection and Traffic Signs Recognition for Autonomous Vehicles", Akademik Bilişim'2019, Ordu Üniversitesi Bilim ve Teknoloji Dergisi, 2019.
- [10] Qingquan Li, Jian Zhou, Bijun Li, Yuan Guo, Jinsheng Xiao, "Robust Lane Detection Method for Low-Speed Environments", MDPI, Dec 2018.
- [11] Thittaporn Ganokratanaa , Mahasak Ketcham, Sasipa Sathienpong, "Real-Time Lane Detection for Driving System Using Image Processing Based on Edge Detection and Hough Transform", International Conference on Digital Information and Communication Technology and its Applications, 2013
- [12] Nur Shazwani Aminuddin, Masrullizam Mat Ibrahim, Nursabillilah Mohd Ali, Syafeeza Ahmad Radzi, Wira Hidayat Mohd Saad & Abdul Majid Darsono, "A New Approach to Highway Lane Detection by Using Hough Transform Technique", 2 (Dec) 2017, Journal of IC
- [13] C.Y. Kuo, Y.R. Lu and S.M. Yang , "On the Image Sensor Processing for Lane Detection and Control in Vehicle Lane Keeping Systems", US National Library of Medicine National Institutes of HealthNCBI, April 2019
- [14] Gurveen Kaur, Dinesh Kumar, "Lane Detection Techniques: A Review", International Journal of Computer Applications, IJCA Journal, 2015.

#### AUTHORS

**Sertap Kamçı** completed her primary school education in Incirli Bahçe Primary School. She completed her secondary school education at the Siir Mektebi Primary School and his high school education at Gungören Anatolian High School. She graduated Istanbul University-Cerrahpasa Faculty of Engineering Computer Engineering at 2019.



**Doğukan Aksu** received his B.S. and M.Sc. degrees in Computer Engineering at Istanbul University in 2015 and 2018 respectively. He is a PhD student in Computer Engineering. His research interests are information and network security, cryptography, artificial intelligence, machine learning and image processing.



**Muhammed Ali Aydın** completed his B.S. in 1997-2001 in Computer Engineering at Istanbul University, M.Sc. in 2001-2005 in Computer Engineering at Istanbul Technical University and Ph.D. in 2005-2009 Computer Engineering at Istanbul University. His research interests are communication network protocols, network architecture, cryptography, information security and network security.

