Kamerabasierte Fahrbahnerkennung zur automatisierten Fahrbahnführung eines Modellauto

Bahri Enis Demirtel

Master Thesis – 02. November 2017

Betreuer: Dr. -Ing. Eric Lenz

TECHNISCHE UNIVERSITÄT DARMSTADT

REGELUNGSTECHNIK UND MECHATRONIK rtm

Aufgabenstellung

Für schriftliche Arbeiten (Pro-/Projektseminar, Studien-, Bachelor-, Master-, Diplomarbeiten, etc.) soll Studierenden ein LaTeX-Dokument zur Verfügung gestellt werden, das die Vorgaben aus den *Richtlinien zur Anfertigung von Studien- und Diplomarbeiten* [**?**] umsetzt. Die Dokumentation soll die Funktionen des Dokumentes beschreiben und Hinweise zu ihrer Anwendung geben.

Grundlage ist die `tudreport`-Klasse. Die damit erstellten Arbeiten müssen sowohl zum Ausdrucken geeignet sein als auch für die Bildschirmdarstellung und die elektronische Archivierung als PDF-Datei.

Beginn:	02. May 2017
Ende:	02. November 2017
Seminar:	15. November 2017

Prof. Dr.-Ing. Ulrich Konigorski

Dr. -Ing. Eric Lenz

Technische Universität Darmstadt
Institut für Automatisierungstechnik und Mechatronik
Fachgebiet Regelungstechnik und Mechatronik
Prof. Dr.-Ing. Ulrich Konigorski

Landgraf-Georg-Straße 4
64283 Darmstadt
Telefon 06151/16-25200
www.rtm.tu-darmstadt.de

Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt habe. Alle Stellen, die aus den Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 02. November 2017

Bahri Enis Demirtel

Kurzfassung

Das LaTeX-Dokument `sada_tudreport` ist eine Vorlage für schriftliche Arbeiten (Proseminar-, Projektseminar-, Studien-, Bachelor-, Master- und Diplomarbeiten, etc.) am Institut für Automatisierungstechnik der TU Darmstadt. Das Layout ist an die *Richtlinien zur Anfertigung von Studien- und Diplomarbeiten* [**?**] angepasst und durch Modifikation der Klasse `tudreport` realisiert, so dass in der Arbeit die gewohnten LaTeX-Befehle benutzt werden können. Die vorliegende Anleitung beschreibt die Klasse und gibt grundlegende Hinweise zum Verfassen wissenschaftlicher Arbeiten. Sie ist außerdem ein Beispiel für den Aufbau einer Studien-, Bachelor-, Master- bzw. Diplomarbeit.

Schlüsselwörter: Studienarbeit, Bachelorarbeit, Masterarbeit, Diplomarbeit, Vorlage, LaTeX-Klasse

Abstract

The LaTeX document `sada_tudreport` provides a template for student's research reports and diploma theses (" Proseminar-, Projektseminar-, Studien-, Bachelor-, Master- und Diplomarbeiten") at the Institute of Automatic Control, Technische Universität Darmstadt. The layout is adapted to the *"Richtlinien zur Anfertigung von Studien- und Diplomarbeiten"* [**?**] and is implemented by modification of the standard `tudreport` class, so that common LaTeX commands can be used in the text. This manual describes the class and dwells on general considerations on how to write scientific reports. Additionally, it is an example for the structure of a thesis.

Keywords: Research reports, diploma theses, template, LaTeX class

Inhaltsverzeichnis

Symbole und Abkürzungen

Lateinische Symbole und Formelzeichen

Symbol	Beschreibung	Einheit
I	Strom	A
R	Widerstand	Ω
U	Spannung	V

Griechische Symbole und Formelzeichen

Symbol	Beschreibung	Einheit
$\boldsymbol{\Psi}$	Datenmatrix	
σ	Standardabweichung	
ω	Kreisfrequenz	s^{-1}

Abkürzungen

Kürzel	vollständige Bezeichnung
Dgl.	Differentialgleichung
LS	Kleinste Quadrate (*Least Squares*)
PRBS	Pseudo-Rausch-Binär-Signal (*Pseudo Random Binary Signal*)
ZVF	Zustandsvariablenfilter

1 Introduction

1.1 Introduction

As in all industries, technology in the automotive industry is continuing to develop day by day. For example, the number of sensors, and their corresponding features, is increasing exponentially. One such sensor is the color camera. To begin with, in the automotive industry, cameras were used only to assist drivers in parking and reversing. Nowadays, however, one of the main functions of color cameras is lane detection, in both autonomous cars and in cars equipped with a lane departure warning system. In this master's thesis, the lanes will be detected and then formulated mathematically.

The results of this master's thesis will be utilized and expanded upon by the students who will participated in the Echtzeitsysteme Projektseminar at the Technical University of Darmstadt. One of the aims of this seminar is to attend the Carolo-Cup organized annually by the Technical University of Braunschweig. Because of that, the width, the curvature, and the changes of the curvature of the track used in this master's thesis are the same as those belonging to the track used in the Carolo-Cup. In a real-life situation, there are of course oftentimes more factors that can hinder lane detection, including shadows cast by trees, buildings, and other structures; sunlight directly entering the lens of the camera and similarly less-than-ideal lighting conditions; dirt and debris on the road surface; and so on.

Therefore, the lanes of the track must be detected in a sufficiently short amount of time and there should be no dead time between lane detection and mathematical formulation. Lane detection must also be sufficiently robust, so that it should not be disrupted by less-than-ideal lighting conditions.

1.2 Problem Statement and Objective Target

Autonomous driving is a topic currently being actively researched. Research on autonomous driving can be conducted in two fundamental areas: lane detection and lane guidance. With regard to lane detection, there are different scientific techniques that can be utilized, according to the literature, all with their own advantages and disadvantages under different conditions. For example, some techniques are suitable for straight lines, but not for curves. Others are suitable for curves as well but do not function well under certain light conditions. Others still are quite robust and suitable for curves, yet are computationally intensive (resulting in a video feed with significant gaps). In this master's thesis, my aim is to research and implement the most appropriate and effective method for use in the Carola-Cup.

1.3 Structure of Thesis

In Chapter 2, the fundamentals of lane detection are explained. All methods utilized in this thesis, along with their respective justifications, are also explained in this chapter. Some methods are also compared with regard to their advantages and disadvantages.

In Chapter 3, the steps of implementation are explained. The components can be divided broadly into the properties of the track, the hardware of the model car, and the software libraries and programs to be utilized. In this chapter, the program flow will also be explained in detail.

In Chapter 4, the results of used methods will be compared. The computing time of all phases in this thesis will be presented and discussed. Also, all used parameters and their effects to this thesis will be also presented and discussed. In this chapter, the answer of the question 'how can the computing time be reduced optimized.

In Chapter 5, the condition of the work will be described. Here the other possible solutions for lane detection will be announced and their advantages and disadvantages will be compared with this master thesis.

In Chapter 6, all results of this master thesis will be presented and also here, it will be announced the improvements and increments for this master thesis.

2 Fundamentals

2.1 Properties of Truck at Carola-Cup

The Carolo-Cup is an annual competition at the Technical University of Braunschweig which are attended by student. Every year the truck and some properties of the competition is changing. For example, in the competitions until 2017 there was no traffic sign but from 2017 there are also some traffic signs, speed limit zones, blocked areas and crosswalks with pedestrian. Because of this reason, in the competitions until 2017, there was only one way to understand who has the right of way. If there is a stop line in the way which in front of intersection, it means, the car has to wait until the intersection is free. In the competitions from 2017, the intersections are in different parts: They are ’Intersections with stop lines’, ’Intersections with give-way lines’, ’Intersections with priority to right’, ’Enforced crossing direction - give-way condition’, ’Enforced crossing direction - right of way condition’. Except ’Intersections with priority to right’, they all have traffic signs, which signs who has priority. If there is a no traffic sign, it means, right side always have priority.

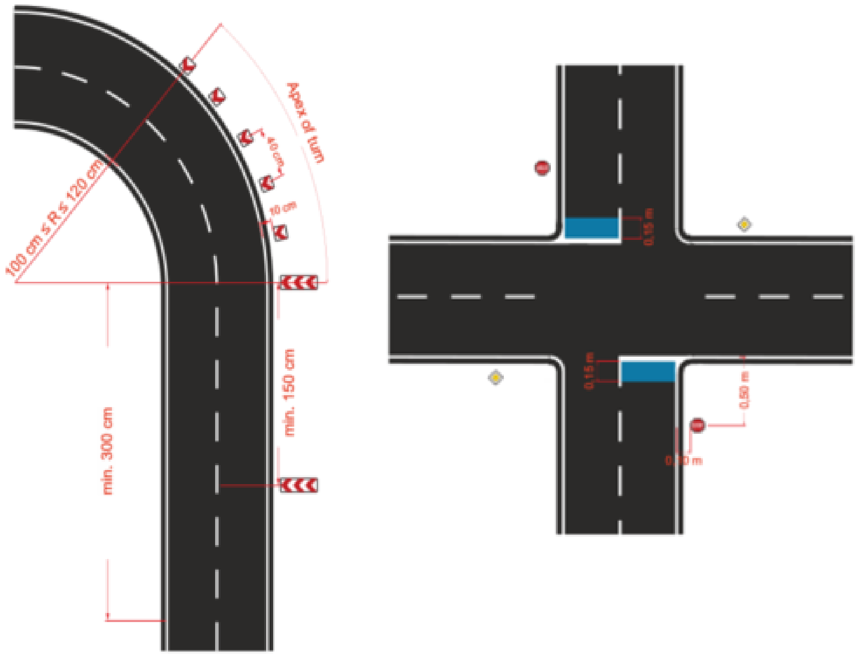


Abbildung 2.1: Left: Markings for sharp turns at Carolo-Cup. Right: Intersections with stop lines at Carolo-Cup

2.2 Inverse Perspective Mapping

Inverse Perspective Mapping(IPM) is an algorithm which obtain accurate bird's-eye view images from the sequential of forward looking cameras. At IPM algorithm, each image pixel is remapped, and a new array of pixels is created where the lines in perspective are transformed into straight lines and objects are distorted. IPM is one of the most used methods at lane detection. At lane detection, IPM ensures to show the lanes vertical and parallel to each other. On the other hand, because of re-mapping of pixels, IPM is a computationally expensive method. Because of this reason, in this master thesis, in some methods, except to remap the all pixels of images, just the pixels, which are relevant to lane and accordingly the fitted curve were remapped. Thanks to this, in some methods, so many time of computing was saved.

For using IPM method, the intrinsic and extrinsic parameters of camera are necessary to process images for coordinate transformation and calibration.

- **Intrinsic Parameters :** Instrinsic parameters include are specific to a camera. It includes information of the focal length (f_x, f_y) and optical centers (c_x, c_y). It is also called camera matrix. It depends on the camera only, so once calibrated, it can be stored for future purposes. It is expressed as a 3x3 matrix:

$$\text{camera matrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

- **Extrinsic Parameters :** Extrinsic parameters are relevant with the camera position. The parameters are H and θ. H is the distance between the camera and ground. θ is the camera tilt angle.

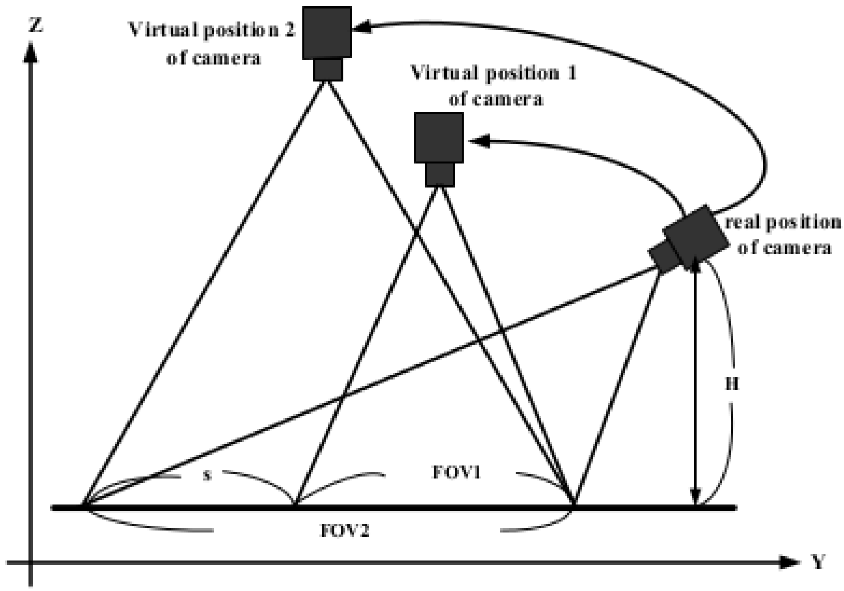


Abbildung 2.2: Related Positions of the Camera [1]

As seen at Abbildung 2.2, the camera on the car has field of view 2(FOV2) at its real position of camera but in this case, the view is not bird-eye view so if the same FOV want to be seen from bird-eye view, IPM will virtually change the position to Virtual position 2 of camera. In this case, although the camera is at its real position, it will look like that it is at the virtual position 2. For that, the image coordinates must be also changed. Below, steps of IPM calculations will be from the paper of [1] described.

In the formula, original image coordinates will be defined as (x,y), the destination image coordinates will be defined as (x^*,y^*), the distance between ground and camera will be defines as H, the focal length of camera will be defines as f and the tilt angle of camera will be defined as θ.

$$x^* = H\frac{xsin\theta + f\,cos\theta}{-ycos\theta + fsin\theta} \; ; \; y^* = H\frac{ysin\theta + f\,cos\theta}{-ycos\theta + fsin\theta}$$

In this equation , the transformed component values of x^* and y^* may be less than or equal to zero. Because of this reason, a constant d is defined as $\left| H(sin\theta + cos\theta)/(f\,sin\theta - cos\theta) \right| + 1$. This means that the coordinate point in the original source image has been mapped into the point of the destination image coordinate system. Below there is the proposed equation :

$$x^* = H\frac{x sin\theta + f cos\theta}{-y cos\theta + f sin\theta} + \mathrm{d}\ ,\ y^* = H\frac{y sin\theta + f cos\theta}{-y cos\theta + f sin\theta} + \mathrm{d}\ ,\ \text{where d} = \left|\frac{H(sin\theta + f cos\theta)}{f sin\theta - cos\theta}\right| + 1$$

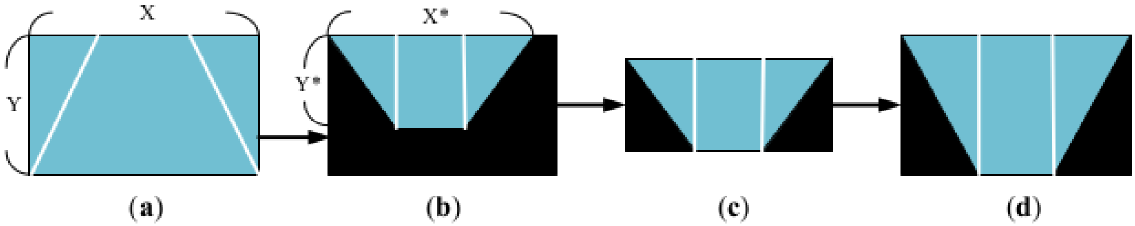


Abbildung 2.3: Procedures of IPM

2.3 Edge Detection

Edge detectors are essential parts of most of computer vision systems. Edge detectors decrease dramatically the amount of data to be processed and extract the useful part of images. They work by detecting discontinuities in brightness. In this project, the edge detector was used, the lanes to detect and prevent unnecessary information from images.There are different methods for edge detection but it can be grouped in two categories. They are :

- **Gradient method :** This method searches the maximum and minimum in the first derivative of the image and within can find the edges. For this method, first order derivative filter must be used. For example : Sobel Operator.
- **Laplacian method :** This method searches for zero crossing in the second derivative of the image and within can find the edges. For this method, second order derivative filter must be used. For example : Laplacian Filter.

According to [2],there are three steps at edge detection algoritm. They are :

- **Filtering :** For edge detection, it is required to use an suitable smoothing filter. The filters sharpen the edges and inhibit the unnecessary informations. It is often utilized to improve the functioning of an edge detector against noise. The more filtering is applied, however, the greater the loss of edge strength.
- **Enhancement :** To be able to better detect edges, changes in the intensity in the area surrounding a point must be determined. Pixels in which a significant change in intensity occurs are emphasized by enhancement, which is usually applied by calculating the gradient magnitude.
- **Detection :** Though many points in an image have a nonzero value for the gradient, not all of these points are actually edges. Because only points with strong edge content are desired, a method must be applied to determine which points are actual edge points. Thresholding is often utilized to do so.

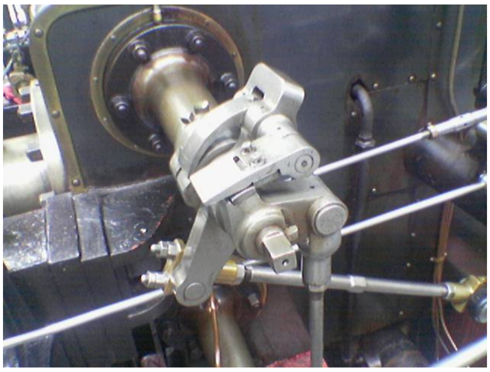
Well known smoothing filters are :

- Sobel-Operator
- Canny Edge Detector
- Laplacian-Filter
- Prewitt-Operator

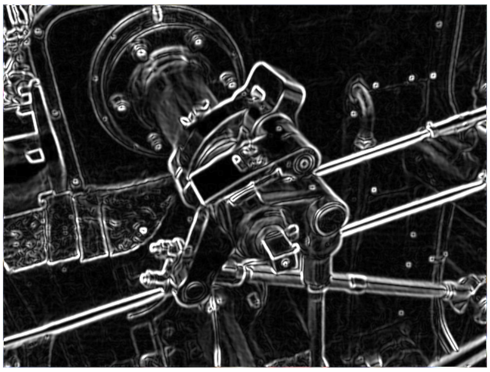
In this master thesis, Sobel Operator was used so it will be described a bit in detail.

2.3.1 Sobel Operator

One of the most used edge detectors at image processing and computer vision is Sobel Operator. The Sobel Operator uses vertical and horizantel masks. These masks are used odd-sized square matrices and they are generally 3x3 matrices.



(a) Original Image



(b) Sobel Operator applied Image

Abbildung 2.4: Sobel Operator[3]

2.4 Hough-Transformation

2.4.1 Standart Hough-Transformation

2.4.2 Probabilistic Hough-Transformation

2.5 K-Nearest Neighbours Algorithm

K-Nearest Neighbor(KNN) is an non-parametric lazy learning algorithm. Non-parametric technique means, that it doesn't make any assumptions on the underlaying data distribution. In the definition of KNN was used the term of 'lazy learning algorithm'. It means, it doesn't use the data training points to do any generalization. In other words, there is no explicit training phase or it is very minimal. It also means that the training phase is pretty fast. Most of the lazy algorithms - especially KNN - makes decision based on the entire training data set. On the other hand, KNN is one of the top 10 data mining algorithms[4].

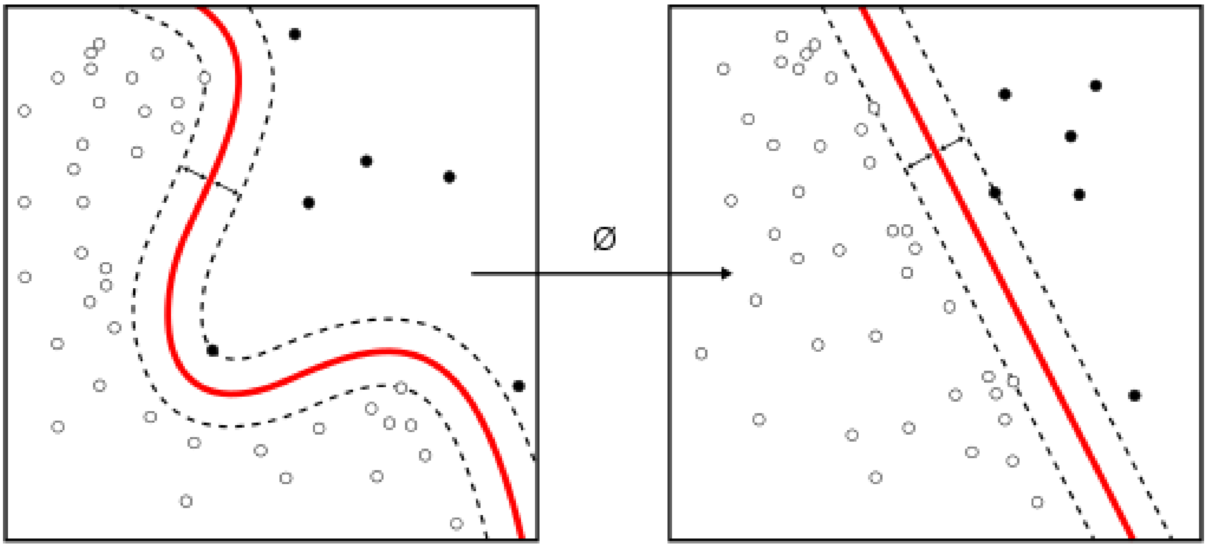


Abbildung 2.5: K-Nearest-Neighbours Algorithm[5]

2.6 Curve Fitting

3 Implementation

3.1 Test Track

As before mentioned, medium-term goal of this master thesis is attending to the Carola-Cup at Braunschweig University so the test truck was prepaid in the Carola-Cup properties by Nicolas Acero Sepulveda, who did his bachelor thesis also with this model auto. For this test truck, two black PVC floor carpets were used and on these floor carpets, the lanes of the truck were made by using white electrical tape. The straight part of the truck was made on one of these PVC floor carpet and the curved part of truck was made on second PVC floor carpet. The straigt part of the truck is approximately 2 meters long and the curve radius of the curved part of test truck is approximately 1 meter. This curve is the tightest curve at Carola-Cup so with this test truck can be tested the worst case situation. In the Carola-Cup competition, the truck is much more bigger but for testing this master thesis, we don't need to build bigger test truck.

Abbildung 3.1: Test Truck

3.2 Hardware

3.2.1 Model Auto

During the course of this master thesis, a model automobile was being used which was prepared for the Projectseminar Echtzeitsysteme at Technical University of Darmstadt. The chassis, steering mechanism, power train, and engine control were derived from the model-building of a Japanese company, Tamiya. The maximum velocity of the model automobile is approximately 1 m/s and the minimum steering radius is around 90 cm.

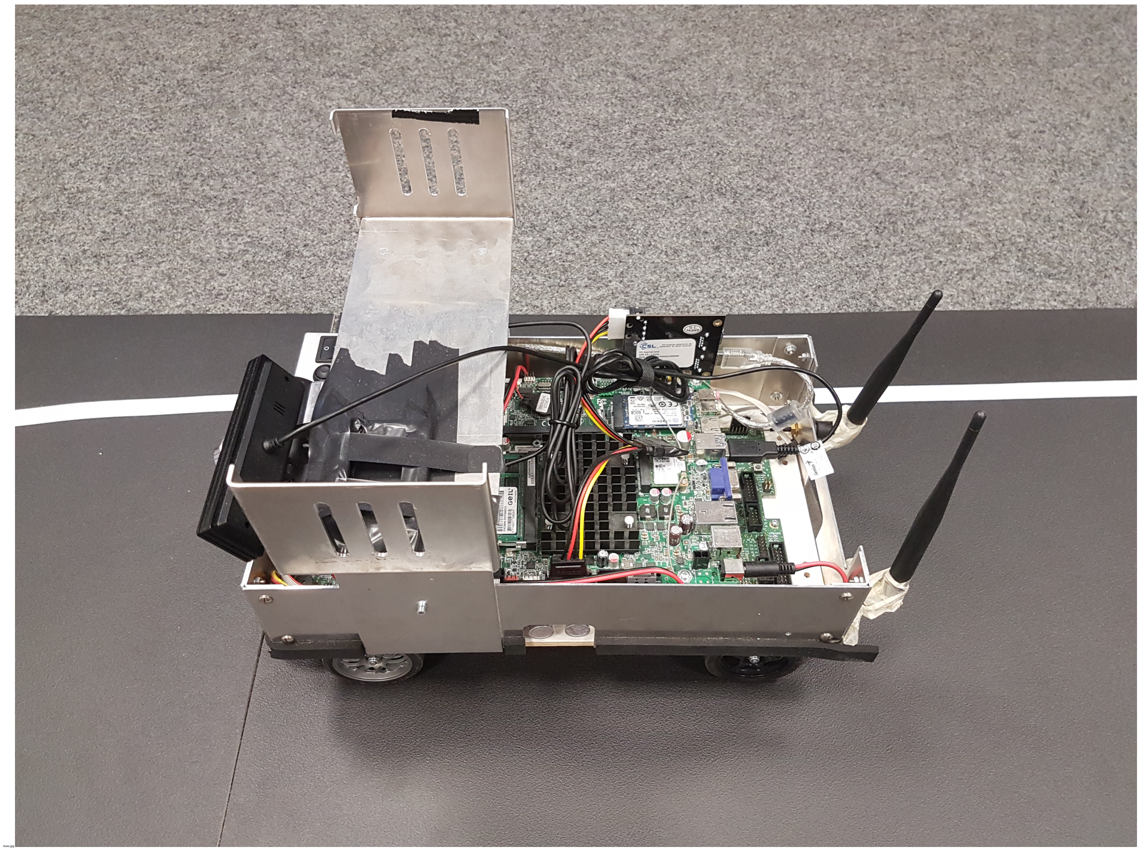


Abbildung 3.2: Model Auto

3.2.2 Microcontroller and Main Board

In this model auto, there is a microcontroller and a main board. The microcontroller is used for controlling steering and receiving the measurements from ultrasonic sensors and hall effect sensors. The 16-bit microcontroller is from MB96300 series from Fujitsu company.

The main board on the model car is from PD10BI-MT ThinMini-ITX series from MiTAC company. This main board communicate with the microcontroller over via UART interface through USB connection. On this main board, there is an Intel Quadcore-Processor and an Intel HD Graphics card. Furthermore, there is a 8 GB DDR3-1600 RAM and 1Gbit/s Ethernet, VGA, HDMI, USB 2.0/3.0, SATA ports and an Intel Dual Band Wireless AC 7260 Network adapter, which is connected to two external WLAN antennas. A 60 GB Kingston SSD-Harddisk is connected over an integrated PCI-Express Port. A 3200 mAh Li-Fe battery is used as power supply.

3.2.3 Camera

The camera is one of the main components of lane detection and accordingly, autonomous driving. For this thesis, I had to research the most suitable camera because all cameras have different properties.

At the beginning of the Projectseminar Echtzeitsysteme, the Logitech C270 HD Webcam was being used. The resolution of the camera is 1280x960 pixels and the Frame per Second (FPS) value is 30 Hertz (Hz) at a 640x480 pixel resolution. The field of View (FOV) is just 60 degrees. The problem with this camera is that if there is a curve, the camera cannot see all of the lanes, and thus is not very suitable for lane detection. When I started my master thesis, there was a Kinect v2 camera on the model car. The Kinect v2 camera was developed by Microsoft and released in 2013. This camera has a depth sensor with a resolution of 512x424 pixels and its FOV is 70x60 degrees. The FPS value is 30 Hz at a 512x424 pixel resolution. This camera also has a color camera with resolution of 1920x1080 pixels and a FOV of 84.1x53.8 degrees. The FPS value is 30 Hz at a 1920x1080 pixel resolution. This camera had two main disadvantages for this master thesis. The first disadvantage is the FOV value of camera. This value is better than the value of Logitech C270 camera but it is still not enough for curve lane detection. The second main disadvantage is the location of the color camera. The color camera of this camera is not in the middle of camera, but rather, on the right. This is a disadvantage for us because when there are curves going left as opposed to right, the camera is unable to see the left and even perhaps the middle lane of the truck. Thus, this is problematic for lane detection.

Due to these reasons, I had to choose a camera which has a sufficiently high FOV value. After doing research, I decided that the Genius Widecam F100 camera is the best choice for this master thesis because this camera has a FOV value of 120 degrees and it can also be used with the Linux Operating System. The resolution of this camera is 1920x1080 pixels and the FOV value is 120 degrees. The FPS is 30 Hz at a 1920x1080 pixel resolution. With this camera, it is possible to detect most if not all lanes, including when there are curves.

Abbildung 3.3: Genius 120-degree Ultra Wide Angle Full HD Conference Webcam(WideCam F100)

3.3 Software

In this chapter, the software algorithms will be focused, which are defined in this master thesis. Through program flow charts and explaination of all steps of program flow charts, it will be tried to explain algorithms better. For finding the best solution, 5 different source codes versions(?variants/?methods) were generated. For all these source codes,the computing times were calculated and compared which solution can detect the lanes better. In next pages, there are detailed explanations of used versions(?variants/?methods). The development environment and the used softwares will be also described , which are used in master thesis.

3.3.1 Development Environment and Related Softwares

As also mentioned at subsection 3.2.2, in this project the introduced main board was used. One of the compact and fast version of Linux 16.04 operating system, *Lubuntu* was installed in this main board.

The version *Kinetic* of ROS was used for implementation of this master thesis. ROS is the abbreviation of **R**obotic **O**perating **S**ystem, which is a robotics middleware (i.e. collection of software frameworks for robot software development). At Ros wiki page[6], ROS is defined that, ROS is an open-source, meta-operating system for your robot. It provides the services you would expect from an operating system, including hardware abstraction, low-level device control, implementation of commonly-used functionality, message-passing between processes, and package management. It also provides tools and libraries for obtaining, building, writing, and running code across multiple computers.

For using prepaid image processing functions, an open source computer vision and machine learning software library was used, which is called as OpenCV. According to OpenCV website[?], there are more than 2500 optimized algorithms in OpenCV library and OpenCV has more than 47 thousand people of user community.

ROS can be programmed by programming languages Python, C++ or Lisp and OpenCV can be programmed by programming languages Python or C++. In this master thesis, C++ was used.

3.3.2 Method 1

4 Evaulation and Discussion

5 Related Works

6 Conclusion

Literaturverzeichnis

[1] Lin, Chien-Chuan und Ming-Shi Wang: *A Vision Based Top-View Transformation Model for a Vehicle Parking Assistant*. Sensors, Department of Engineering Science, National Cheng Kung University Taiwan, No.1, University Road,Tainan City 701, Taiwan, 2012.

[2] Ramesh Jain, Rangachar Kasturi, Brian G. Schunck: *Machine Vision*. McGraw-Hill, 1995.

[3] Wikipedia: *Sobel operator*, October 2017.

[4] Xindong Wu, Vipin Kumar, J. Ross Quinlan Joydeep Ghosh Qiang Yang Hiroshi Motoda Geoffrey J. McLachlan Angus Ng Bing Liu Philip S. Yu Zhi-Hua Zhou Michael Steinbach David J. Hand Dan Steinberg: *Top 10 algorithms in data mining*. Springer-Verlag London Limited 2007, 2007.

[5] Wikipedia: *k-nearest neighbors algorithm*, August 2017.

[6] Dirk, Thomas: *What is ROS*, May 2014.