

ELEKTRONISCHE MESS- UND  
DIAGNOSETECHNIK

[ expleo ]



# Multi Sensor Ramp Detection and Localization for Autonomous Valet Parking

Master thesis

Felix Saalfrank  
368199

January 18, 2022

Supervisor:  
Lars Schürmann  
Person 2  
Gutachter:  
Prof. Dr. Gühmann  
Prof. Dr. Orgelmeister

Technische Universität Berlin  
Faculty IV - Electrical Engineering and Computer Science  
Department of Energy and Automation Technology  
Chair of Electronic Measurement and Diagnostic Technology

Berlin, 3. Januar 2022

Masterarbeit für Herrn Felix Saalfrank, Matrikel-Nr. 368199

## Multi-Sensor Ramp Detection and Localization for Autonomous Valet Parking

Prof. Dr.-Ing. Clemens Gühmann

**Problem:** Autonomous Valet Parking (AVP) will make parking easier in the future, by allowing the driver to exit the car in a drop off zone in front of a parking garage, and the car will find a parking spot on its own. When the driver calls the car again, it will also autonomously find its way to the driver. For this to work, a map of the parking garage and precise localization of the car is necessary. A challenging part is the necessary change of levels during the procedure because the ramps in parking garages are usually very narrow and require precise localization and control of the car. Therefore, information about whether or not the car is driving onto a ramp is necessary. This allows the controller of the car to adjust for the changing road conditions, e.g. increasing or decreasing the motor output power when driving up or down respectively. Also, because the maps used for the localization of the car are usually stored separately for each parking level, the loading of the new map should be initiated while the car is on a ramp.

Sekretariat EN 13 Raum EN 538  
Einsteinufer 17  
10587 Berlin

Telefon +49 (0)30 314-29393  
Telefax +49 (0)30 314-22120  
clemens.guehmann @ tu-berlin.de

Bearbeiterin  
Ewa Heinze

Telefon +49 (0)30 314-22280  
Telefax +49 (0)30 314-22120  
ewa.heinze@tu-berlin.de

**Task:** The goal of this thesis is to implement an algorithm for a car, which can detect ramps. Besides the detection, ramp properties such as the inclination angle or length should be measured. To implement this, various sensor setups will be used and compared. An Inertial Measurement Unit (IMU) will be the main sensor and will be responsible for the exact measurement of the ramp's properties, in conjunction with a wheel odometer. Additionally, a Light Detection and Ranging (LiDAR) sensor will be used to allow for the detection of the ramp, before entering it. The data of the LiDAR could also be fused with the IMU data to prevent false detections by the IMU. A camera will be tested as well for the early detection of a ramp and compared to the LiDAR. Test drives in one specific parking garage and test car will be performed. A camera will be used to validate if the detection was at the right time and the estimated ramp properties will be compared to manual measurements.

### Research steps:

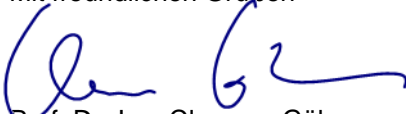
- Research of current methods to determine road grade angle using IMU, LiDAR or camera
- Comparison and selection of the most appropriate method for each sensor
- Implementation of a ramp detection algorithm
  - using an IMU
  - using a LiDAR sensor
  - using a camera
- Testing and optimizing of the methods
- Comparison and evaluation of the different methods used

- Documentation and presentation of the results and thesis

### Organization

This thesis will be written in collaboration between TU-Berlin and Expleo Germany GmbH. Felix Saalfrank's supervisor is Lars Schürmann. All necessary documents and resources will be provided by Expleo Germany GmbH. The presentation of the researched literature, the analytical work and the experiments will be carried out according to the rules of best scientific practice. The results will be presented publicly in the seminar of the Electronic Measurement and Diagnostic Technology chair.

Mit freundlichen Grüßen



Prof. Dr.-Ing. Clemens Gühmann

# Kurzfassung

100-200 Wörter Kurzfassung (deutsch)

# Abstract

100-200 word abstract (english)

# Declaration

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

# Contents

<b>Acronyms</b>	<b>x</b>
<b>List of Symbols</b>	<b>xi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Outline . . . . .	1
<b>2 State of the art</b>	<b>2</b>
2.1 IMU . . . . .	2
2.2 LiDAR . . . . .	3
2.3 Camera . . . . .	3
2.4 Bla . . . . .	3
<b>3 Background</b>	<b>4</b>
3.1 Mathematical . . . . .	4
3.2 Sensors . . . . .	4
3.2.1 IMU . . . . .	4
3.2.2 LiDAR . . . . .	6
3.2.3 Odometer? . . . . .	8
3.2.4 Camera? . . . . .	8
3.3 ROS . . . . .	8
3.4 Sensor Fusion stuff? . . . . .	8
3.5 Signal processing . . . . .	8
<b>4 Experimental Setup</b>	<b>9</b>
4.1 Sensors . . . . .	9
4.1.1 IMU . . . . .	9
4.1.2 LiDAR . . . . .	9
4.2 Sensor placement . . . . .	10
4.3 Car . . . . .	11
4.4 Garage . . . . .	11
<b>5 Methods</b>	<b>12</b>
5.1 IMU only . . . . .	13
5.1.1 Calibration . . . . .	13
5.1.2 Linear acceleration only . . . . .	14
5.1.3 Angular velocity only . . . . .	14
5.1.4 Complementary filter . . . . .	14
5.2 LiDAR only . . . . .	14
5.2.1 Calibration . . . . .	14
5.2.2 Algorithm . . . . .	15

---

5.3	Camera only . . . . .	18
5.4	Sensor fusion . . . . .	18
5.4.1	IMU and Odometer . . . . .	18
5.4.2	IMU and odometer and LiDAR . . . . .	19
<b>6</b>	<b>Results</b>	<b>20</b>
6.1	Evaluation concept . . . . .	20
6.2	Ramp metering? (IMU) . . . . .	20
6.3	Ramp detection (LiDAR and camera) . . . . .	20
<b>7</b>	<b>Conclusion</b>	<b>21</b>
<b>8</b>	<b>Appendix</b>	<b>24</b>



# List of Figures

3.1	Micro structure of a Microelectromechanical Systems (MEMS) accelerometer . . . . .	5
3.2	Setup of a mechanical spinning Light Detection And Ranging (LiDAR) [6] .	7
4.1	Mounting of the LiDAR . . . . .	10
5.1	Car driving on a ramp. Due to forward acceleration the car tilts back. . . .	12
5.2	Coordinate frames problem . . . . .	12
5.3	Frame transformation from the Intertial Measurement Unit (IMU) frame $\mathcal{I}$ , over the intermediate frame $\mathcal{B}$ to the car frame $\mathcal{C}$ . . . . .	13
5.4	Algo for LiDAR alignment . . . . .	16
5.5	Algo for ramp detection . . . . .	17
5.6	Gravity measured by IMU (in car frame). Show that $g_z$ etc . . . . .	19

# List of Tables

4.1	Comparison of the two used LiDARs [14][13]	10
5.1	Used parameters for lidar algo	18

# Acronyms

**AHRS** Attitude Heading Reference System

**AVP** Automated Valet Parking

**FIR** Finite Impulse Response

**FOV** Field Of View

**GPS** Global Positioning System

**IIR** Infinite Impulse Response

**IMU** Intertial Measurement Unit

**LiDAR** Light Detection And Ranging

**MEMS** Microelectromechanical Systems

**RADAR** Radio Detection And Ranging

**RANSAC** Random Sample Consensus

**SLAM** Simultaneous Localization and Mapping

**SONAR** Sound Navigation And Ranging

# List of Symbols

${}^{\mathcal{A}}_{\mathcal{B}}\mathbf{q}$  Quaternion to transform from a to b

${}^{\mathcal{A}}_{\mathcal{B}}\mathbf{M}$  Rotation matrix to transform from a to b

$\mathbf{v}$  Vector

${}_{\mathcal{A}}\mathbf{v}$  Vector in coordinate frame  $\mathcal{A}$

$\|v\|$  Norm of vector

$\hat{\mathbf{v}}$  Unit vector

# Todo list

2do	iii
2do	iv
Just copied this from the expose, needs improvement	
maybe some more sections about the specific topics or about goals	1
2do	1
Not really IMU, maybe in prev section instead	2
2do	2
2do	3
2do	3
2do	3
2do	4
reference?	5
check what type of deg/h there are, daytoday vs in-run	6
Sth sth lorenz (short because not used)	6
2do	6
fact check, might have understood it wrong	7
Sort citations and maybe add some more	7
2do	8
Probably not necessary, as everyone knows what a camera is	8
2do	8
2do	8
filtering also often introduces a delay	8
How much about dsp? E.g. also noise, aa etc or only type of filters?	8
Figure: Picture of myAHRS+ or/and ZED2i	9
Also add ZED 2i IMU and compare (probably table similar to LiDAR)	9
Figure: Picture of robosense or/and velodyne	9
IMU placement not really important, should just be static	10
Make a good sketch again with labels etc and check if its easy understandable, only	
then continue in tikz and latex	11
LiDAR pitch angle at which most point hit area before ramp can be calculated,	
explanation here using sketch 4.1	11
Figure: Picture of eGolf	11
2do	11
Figure: Picture of ramps and/or figure of ramps showing angles	11
Think of a good way to measure the true angle of the ramps	11
2do	12
point at the front of the car, centered laterally (same y-coordinates as base_link, but	
z at ground level and x further in front)	15
FLOW chart for calibration can probably removed, flowchart for algo should be	
enough (very similar)	15
no idea if wheel chair ramps are common in parking garages?	17
Add down ramp detection	17

Brief explanation what sensor fusion is and why useful . . . . .	18
Explain model . . . . .	18
I can't explain it well . . . . .	18
2do . . . . .	19
2do . . . . .	19
2do . . . . .	20
2do . . . . .	20
2do . . . . .	20

# Chapter 1

## Introduction

### 1.1 Motivation

Parking is one of the most challenging driving tasks and the cause of almost half of the car accidents [18]. Current cars are already able to fully automated park on their own in parallel or perpendicular parking spaces. But due to the very limited space in cites, parking garages are often used in central areas [1]. Automated Valet Parking (AVP) allows for a fully automated parking experience. The car is left in a drop-off zone and finds a parking spot on its own. Afterwards the driver can give a command and the car leaves the parking spot again and picks up the driver. AVP saves time, the hassle of remembering the parking level and spot and furthermore allows to use the available space more efficiently and also minimizes the risk of collisions. For this to work an exact mapping of the environment and localization of the car in the garage is necessary.

This can be done either by Simultaneous Localization and Mapping (SLAM) or the area can be mapped beforehand (e.g. using LiDAR sensors) in which case only a localization of the car is necessary. The mapping can be done in 2D or 3D. 2D maps only show information of the current level. Hence if the car is driven up or down a ramp, the new map of the corresponding floor has to be loaded. Because the localization usually only works in a 2D-plane, a change of levels would not be detected. To solve this problem, a ramp detection has to be implemented.

Just copied this from the expose, needs improvement  
maybe some more sections about the specific topics or about goals

### 1.2 Outline

Brief overview over structure of thesis

## Chapter 2

# State of the art

### 2.1 IMU

In [4] different methods to estimate the road grade angle are discussed.

Not really IMU, maybe in prev section instead

There exist methods without Inertial Sensors relying on a model describing the longitudinal movement of the vehicle and the topology of the road. Both models are fused using a Kalman filter to improve the accuracy of the estimation [17]. A Kalman filter is also used in [16], where vehicle sensor data and Global Positioning System (GPS) data are fused. Besides the road grade, the vehicle mass is often also unknown and estimated as well [16, 7]. Another method using GPS data and IMUs to calculate the vertical and horizontal velocity change respectively and thereby the road grade is proposed in [15]. [25] omits the IMU and relies on a GPS sensor and a barometer.

GPS satellites broadcast information about their position and exact time to a GPS receiver, which then can calculate its position using triangulation [5]. While an accuracy of up to 1 m can be achieved when outside, the performance significantly drops when used indoors. The radio waves sent from the satellites are scattered, attenuated or blocked completely by walls and other obstacles, resulting in a very weak or even a complete loss of the signal [10].

Most methods mentioned above do not seem fit for the task, due to the use of GPS. Furthermore many internal measurements such as the engine torque, brake system usage, selected gear etc. can not easily be accessed and thus might not be available.

A method which does not use GPS, but only accelerometers and wheel odometers instead is described in [9]. The vehicle acceleration, calculated by deriving the wheel speed measurements in respect to time, is subtracted from the accelerometer signal in longitudinal direction. The remaining part is then the gravitational acceleration and can be used to calculate the road grade angle. A similar approach is used in [19].

- Maybe add a bit more to overview
- Then write more specific about methods actually used, such as:
- Acceleration method ([11])
- Complementary filter (e.g. [3])
- Other methods (see [3] or [4])



## 2.2 LiDAR

- LiDAR methods for plane detection
- If found, methods for elevation estimation

## 2.3 Camera

- Something simple
- Stereo vs mono

## 2.4 Bla

Add ausblick was genommen wird

## Chapter 3

# Background

### 3.1 Mathematical

E. g.

- Vector projection
- Quaternions
- Rotation matrix

### 3.2 Sensors

#### 3.2.1 IMU

An IMU is used to track the orientation and position of an object. Common uses are in the aerospace or automotive industry, often in combination with other sensors, to give information about the pose and position of a vehicle. More recently with the invention of MEMS and specifically MEMS-IMUs which allow for a very small form factor at a low cost, IMUs are also used in consumer electronics such as smartphones or fitness tracker. An IMU usually consists of the three following sensors. The acceleration is measured using an accelerometer and can be used to determine the velocity and the covered distance by integrating the acceleration with respect to time once respectively twice. The gyroscope gives information about the change of orientation. Often times a magnetometer is used as well, which is able to measure the earth's magnetic field and is used to correct the measurements of the gyroscope. It allows for the determination of the absolute heading, whereas the gyroscope can only measure relative change. But because it is very sensitive to other magnetic objects, it is often omitted. IMUs can be typically divided into the two following categories.

In the first type, the stable platform systems, the inertial sensors are mounted in such way, that they are always aligned with the reference frame. This is achieved using gimbals, which allow movement along all three axes. The gyroscopes on the platform measure the rotation and send them to torque motors, which rotate the gimbals to keep the platform in alignment with the reference frame. The advantage of stable platform systems is that the calculation of orientation and position is straight forward. The angles of the gimbals can be measured to get the orientation and to get the position, the accelerometer measurements have to be corrected for gravity (which is  $9.8 \text{ m/s}^2$  in upward direction) and be integrated two times. No coordinate transformation is necessary. The disadvantages are that the mechanical structure of the setup is complex, needs regular maintenance, requires a lot of space and has high costs.

The second type are strapdown systems, which are mostly used today. [As the name](#) suggests all the parts are fixed onto the device and are thus not anymore always aligned with the reference frame. Advantages are that due to the lack of gimbals and motors a significantly smaller build is possible and lower production costs can be achieved. A disadvantage is that the calculation of the orientation and position is more complex, the rate gyroscopes have to be integrated to get the orientation and can then be used to transform the accelerometer signals into the reference frame. But with the decrease of computational cost this disadvantages continues to diminish. And even though they are continually improved, the accuracy does not quite match the of stable platform systems.

reference?

There are many different types of gyroscopes and accelerometers such as mechanical, optical or solid state, but only the functionality of MEMS will be described, because those will also be used in the experiment. Information about the working principle of other systems and also much more information about IMUs in general can be found in [24].

MEMS consist of electrical and/or mechanical components in the size of 100 nm to 1 mm, allowing for a very small form factor. Other characteristics of MEMS are that they can easily be mass produced allowing for low cost and usually also need less power than traditional systems, because everything is integrated on the chip [20]. Almost all consumer grade electronics uses MEMS-IMUs nowadays, but they also find more and more use in many industry segments, as their accuracy continues to improve [12].

### MEMS Accelerometer

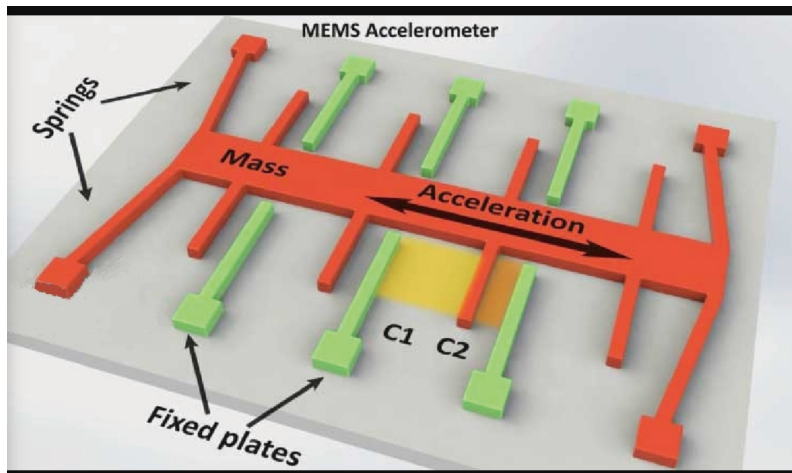


Figure 3.1: Micro structure of a MEMS accelerometer

The accelerometer is used to measure the acceleration. Besides dynamic acceleration there is the static and constant gravity acceleration on earth in upward direction. This allows for the determination of one axis of the IMU, even if it is not moving. Often times only the dynamic accelerations are of interest and to get them the acceleration data during stand still must be measured and subtracted. The micro structure of a MEMS accelerometer is shown in figure 3.1. A mass is suspended by springs along one axis and if an acceleration along this axis occurs, the mass moves in the opposite direction due to Newton's second law. The mass has little fingers perpendicular to the moving direction axis, which affect the capacity between the fixed plates. The change of capacity and thus voltage can be measured, from which the acceleration can be calculated. To be able to measure the acceleration along all three axis the same setup is used three times, perpendicular to each other.

### MEMS Gyroscopes

A gyroscope measures the angular velocity. The setup of a MEMS gyroscope is similar to that of a MEMS accelerometer. A proof mass is suspended on a frame and responds to an input force. MEMS gyroscopes make use of the Coriolis effect, which states that an rotating object with the angular velocity  $w$  of mass  $m$  and velocity  $v$  experiences a force

$$F_C = -2m(w \times v).$$

To measure the effect, a mass is vibrating along one axis, which in turn is also suspended. If the mass is oscillating along one axis and a rotation is applied, a second oscillation on the axis perpendicular to the rotation axis can be observed. E.g. if the mass oscillates along the x-axis and a rotation around the z-axis is applied, a vibration along the y-axis can be observed. By measuring the amplitude and phase of the secondary oscillation the absolute value and direction of the angular velocity can be calculated. While MEMS gyroscopes do not achieve the same accuracy as optical gyroscopes they offer many advantages such as smaller physical properties (weight and size), lower power consumption and startup time as well as a significantly lower cost. Optical gyroscopes cost in the range of \$10,000 whereas MEMS gyroscopes can cost as little as \$3 [12]. But this comes at the cost of a worse angle drift which increases from 0.01 to 0.1 deg/h for optical gyroscopes to 10 deg/h for MEMS-IMUs. MEMS gyroscopes have replaced other gyroscope types in most areas, but in areas where the highest precision possible is necessary, typically in military industry, optical gyroscopes are still used today.

ck what type of  
/h there are, day-  
ay vs in-run

### (MEMS) Magnetometer

Sth sth lorenz (short because not used)

The disadvantages are that the magnetometer is easily influenced by other ferromagnetic material and electronic devices. Therefore indoor use while getting reliable data is rarely possible.

### Typical MEMS errors

Maybe a sentence about idc about problems, because raw measurements are mostly used. The first type of errors are calibration errors, which can be eliminated. Common calibration errors are a constant bias (offset), scaling or misalignment (axes are not orthogonal to each other). The turn-on bias is different each time the IMU starts up, but can be removed as well.

- Calibration errors
- Turn-On Bias
- Bias instability
- Bias Correction methods
- VERY BRIEF
- maybe as table

### 3.2.2 LiDAR

LiDAR is a method to measure distance to objects. Similar to other systems such as Sound Navigation And Ranging (SONAR) or Radio Detection And Ranging (RADAR),

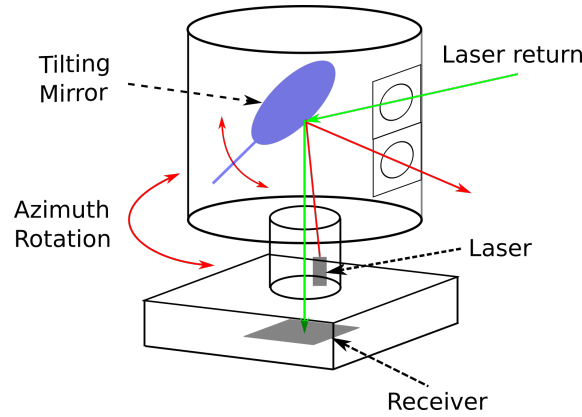


Figure 3.2: Setup of a mechanical spinning LiDAR [6]

LiDAR uses the time-of-flight principle. A short laser pulse with the velocity  $v$  is sent into the environment and the reflected light is analyzed. The duration  $\Delta t$  it took from sending to receiving can then be used to calculate the distance  $s$  with

$$s = v \frac{\Delta t}{2}.$$

The change of intensity and wavelength of the returning light are measured as well and can provide information about the reflectivity of the object (intensity) or the chemical composition of the air (wavelength). Common uses of LiDAR are the analysis of earth's atmosphere, 3D mapping of environments or in the field of autonomous driving for object detection, tracking and SLAM. Basically all applications which use RADAR can also be used with a LiDAR instead, allowing for a greater accuracy.

There are different LiDAR types but the principles are similar. A transmitter generates a signal and sends it into the environment using a scanning system and a transmission optic. As transmitter a laser with a wavelength of 850 nm to 950 nm (near-infrared) is typically used. The scanning system allows the laser to explore a large area instead of only a single point by steering the light at different azimuths and vertical angles and can be divided in mechanical spinning or solid state systems. Mechanical spinning systems is the oldest technology and is still mainly used today. A mirror which can be rotated around an axis is used, allowing for a greater vertical Field Of View (FOV). Also the whole LiDAR base on which the laser is mounted can be rotated independently from the mirror, allowing for a 360° horizontal FOV. To get a sufficient resolution the LiDAR has to spin at a high speed, but some LiDARs also use additionally a vertical array of lasers instead of only one to further increase the density of the generated point cloud. The working principle of a LiDAR using the mechanical spinning method is shown in figure 3.2. While mechanical spinning systems are very precise, they are bulky, need a lot of power and are expensive. Solid state systems and especially MEMS LiDARs try to overcome those problems. MEMS LiDAR are quasi-static, the only part that moves is the on the chip embedded mirror, but due to the small size (1 mm to 7 mm diameter) very little power has to be used to move it. They can be rotated on up to two axes, but because the laser cannot be rotated a horizontal view of 360 is not possible. But advantages compared to mechanical systems are the smaller former factor and lower cost.

After transmitting the laser signal the reflected light passes through the receiving optic and is received by photodetectors. A processing unit then generates a 3D point cloud from all the received measurements.

Sort citations and maybe add some more

fact check, might have understood it wrong

### 3.2.3 Odometer?

- Induktive Drehzahlsensoren
- Differentielle Hall-Sensoren

#### Inductive

asd

### 3.2.4 Camera?

Probably not necessary, as everyone knows what a camera is

## 3.3 ROS

Brief introduction to ROS

## 3.4 Sensor Fusion stuff?

Could also just be explained in methods

## 3.5 Signal processing

... is necessary. ... can be divided into filtering and smoothing. Filtering can be used in live applications and produces an estimate of the current value by taking the past values into account, whereas smoothing uses past and future samples and thus introduces a delay if used on live data. Because the detection should be live, only the filtering methods will be examined.

Filtering also often produces a delay

How much about dsp? E.g. also noise, aa etc or only type of filters?

Digital filters can be generally divided into two different categories. Finite Impulse Response (FIR) filter rely on a fixed number of recent input values. An example would be the moving average filter, which takes the past  $n$  values into account. Infinite Impulse Response (IIR) filter rely on previous output as well as most recent input by summing all points with a certain weight (e.g. exponential filter). This also explains the naming of the two types, the FIR filter "forgets" past values, whereas the IIR filter uses the previous estimate and thus theoretically takes all past values into account.

Savitzky-Golay filter

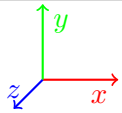
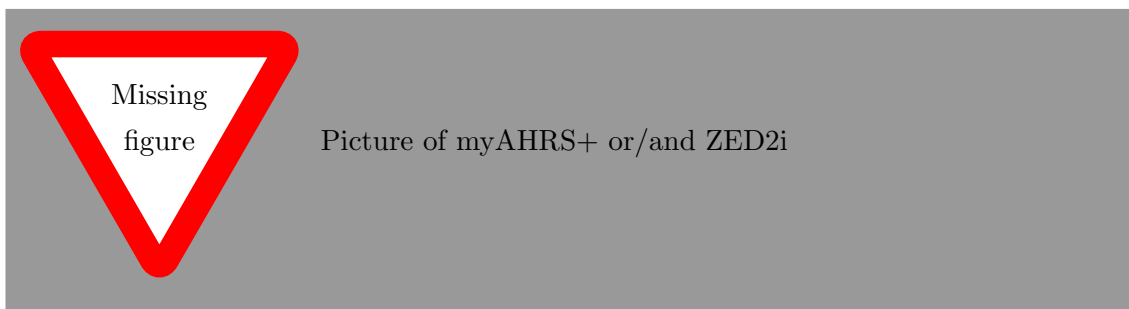
$$y = \frac{1}{h} \sum_{i=\frac{1-m}{2}}^{\frac{m-1}{2}} C \quad (3.1)$$

## Chapter 4

# Experimental Setup

### 4.1 Sensors

#### 4.1.1 IMU



For the experiments the myAHRS+, a low cost high performance Attitude Heading Reference System (AHRS) will be used. An AHRS contains an IMU and outputs the raw data but also has an integrated Kalman filter which calculates the pose in form of quaternion or euler angles. It offers an micro-USB interface and runs with up to 100 Hz. It can capture a change of  $\pm 2000$  dps (degrees per second),  $\pm 16 g$  and  $\pm 1200 \mu T$ . During the experiment only a fraction of this range is expected to be reached, hence the sensor seems suitable. Besides the hardware the unit already has an Extended Kalman Filter (EKF) on board. The EKF fuses the measurements of the three sensors and estimates a quaternion (and sth else?) from it. But this will not be used.

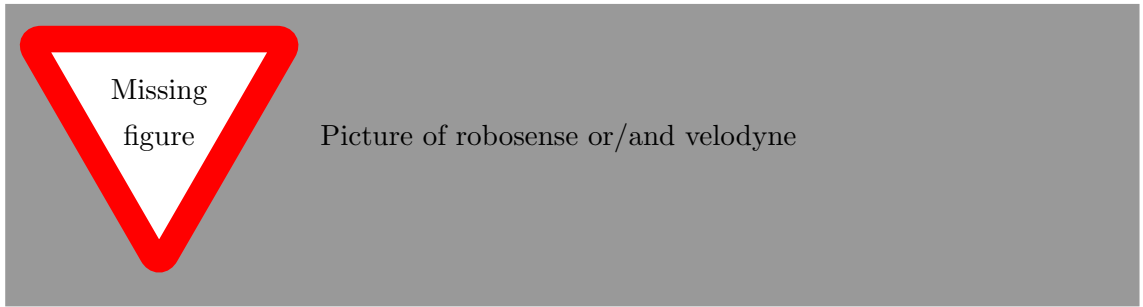
Also add ZED 2i IMU and compare (probably table similar to LiDAR)

#### 4.1.2 LiDAR

Two different LiDARs will be used during the experiment. The RS-Bpearl and the Velodyne UltraPuck. The most relevant specifications of the two LiDARs can be seen in table 4.1. Both are mechanical LiDARs and have the same number of laser channels, but the Velodyne has a significant better vertical resolution, due to the smaller vertical FOV.

Table 4.1: Comparison of the two used LiDARs [14][13]

	RS-Bpearl	Velodyne Ultra Puck
Channels	32	32
Range	100 m	200 m
Range accuracy	$\pm 3$ cm	$\pm 3$ cm
Horizontal FOV	$360^\circ$	$360^\circ$
Vertical FOV	$90^\circ$	$40^\circ$ ( $-25^\circ$ to $15^\circ$ )
Horizontal resolution	$0.2^\circ$ to $0.4^\circ$	$0.1^\circ$ to $0.4^\circ$
Vertical resolution	$2.81^\circ$	$0.33^\circ$
Frame rate	10 Hz to 20 Hz	5 Hz to 20 Hz
Laser wavelength	905 nm	903 nm
Points per second	576,000	600,000



## 4.2 Sensor placement

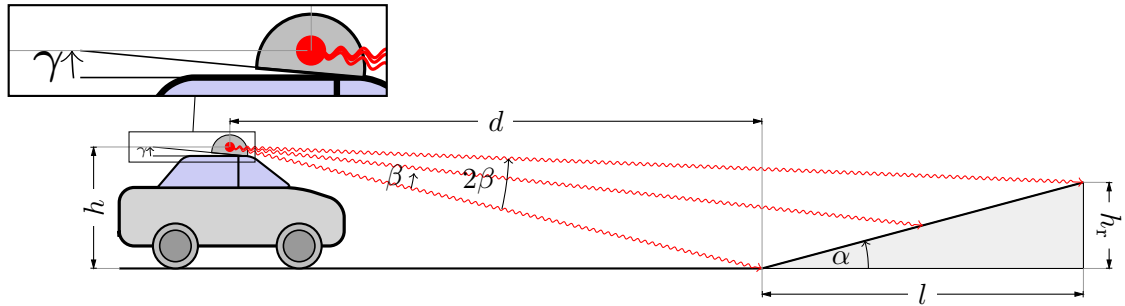


Figure 4.1: Mounting of the LiDAR

IMU placement not really important, should just be static

The LiDAR will be placed on top of the car, to get a greater FOV. The pitch angle at which the LiDAR will be mounted should be chosen so that the number of points in the area at the beginning of the ramp are maximized. This allows for an easier detection of planes with different inclination angles. The coordinates at which the lasers hit the ground and ramp depend on the height of the LiDAR  $h_L$ , the distance to the ramp  $d$ , the angle of the ramp  $\alpha$ , the pitch angle  $\gamma$  at which the LiDAR is mounted and finally on the vertical resolution and FOV of the LiDAR. The coordinates can be calculated in the following way. Assuming there is no ramp, the x-coordinate where the laser waves hit the ground can be calculated with

$$x = \frac{h}{\tan(n\beta - \gamma)} \quad (4.1)$$



with  $n$  being the "laserID" starting from the lowest opening angle and going to the highest. When the ground is not a flat anymore, the assumption from 4.1 does not hold anymore. The x distance between two laser points reduces on a ramp. ... whatever who cares

Make a good sketch again with labels etc and check if its easy understandable, only then continue in tikz and latex

$$c = h - \frac{z}{\tan \alpha}$$

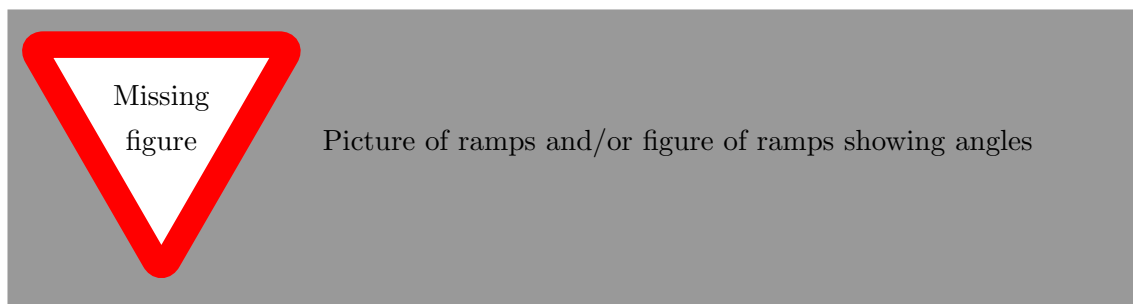
LiDAR pitch angle at which most point hit area before ramp can be calculated, explanation here using sketch 4.1

### 4.3 Car



- Some stats as height, track width etc, electric car hence less vibrations
- Problem with speed limit and odom (no odom when driving ramp up)
- Connection setup (maybe own section or maybe not interesting at all)

### 4.4 Garage



Think of a good way to measure the true angle of the ramps

## Chapter 5

## Methods

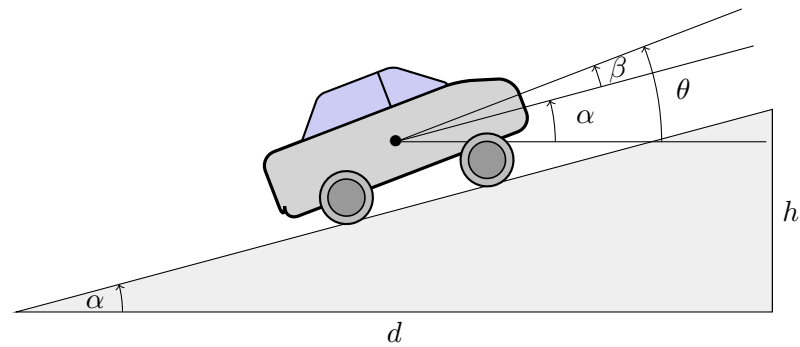


Figure 5.1: Car driving on a ramp. Due to forward acceleration the car tilts back.

- Coordinate system problem
- Problem with car tilt
- Short description of different methods

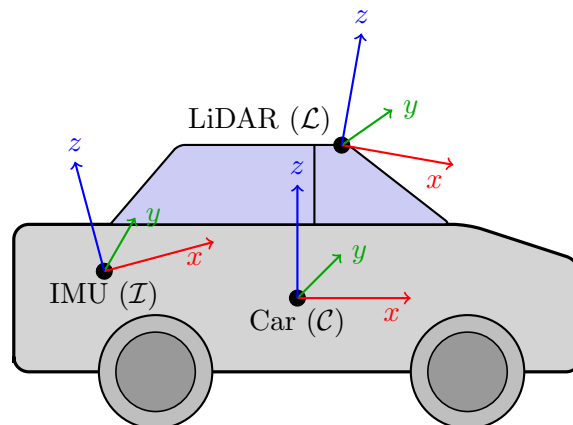


Figure 5.2: Coordinate frames problem

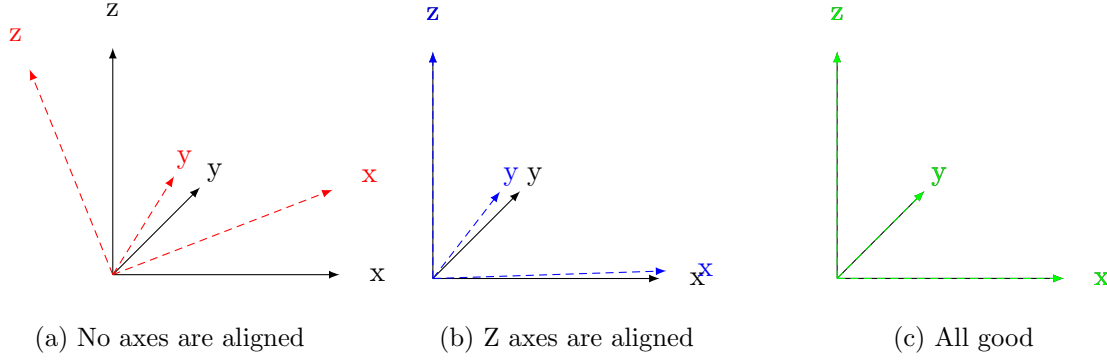


Figure 5.3: Frame transformation from the IMU frame  $\mathcal{I}$ , over the intermediate frame  $\mathcal{B}$  to the car frame  $\mathcal{C}$

## 5.1 IMU only

### 5.1.1 Calibration

Because the measurements of the IMU are measured in the device frame  $\mathcal{I}$ , which is not aligned with the car frame  $\mathcal{C}$ , they have to be transformed. This can be achieved using a rotation matrix  ${}^{\mathcal{I}}\mathbf{M} \in \mathbb{R}^{3 \times 3}$  which transforms the measurements of the linear acceleration  ${}^{\mathcal{I}}\mathbf{a}_n \in \mathbb{R}^{1 \times 3}$  and angular velocity  ${}^{\mathcal{I}}\mathbf{v}_n \in \mathbb{R}^{1 \times 3}$  into the car frame. Note that the upper index? to the left of the matrix symbol denotes the source frame, whereas the destination frame is written below it.  $n \in \mathbb{N}$  is the time step.

During standstill, the only measurable acceleration besides noise and bias is the acceleration due to gravity. Assuming the car stands on flat ground, the gravity acceleration in the car frame is measured in upwards z-direction. In a first step, the measured linear acceleration (gravity) in the IMU frame will be aligned with the z-axis of the car. According to Euler's rotation theorem, which says that any arbitrary rotation of a rigid body while holding one point (origin) fixed can be achieved by a rotation around a single fixed axis passing through the origin, there exists one rotation axis  $\mathbf{j}$  and rotation angle  $\alpha$  to achieve this.

In the first step, the quaternion  ${}^{\mathcal{I}}\mathbf{q}$  which describes the rotation from the device frame  $\mathcal{I}$  to an intermediate coordinate system  $\mathcal{B}$ , which has the z-axis up, will be found. Resulting in  ${}^{\mathcal{B}}\mathbf{z} = {}^{\mathcal{C}}\mathbf{z} = \mathbf{e}_z = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^\top$ . Note that this is not necessarily true for the other axes,  ${}^{\mathcal{B}}\mathbf{x} \neq {}^{\mathcal{C}}\mathbf{x}$  and  ${}^{\mathcal{B}}\mathbf{y} \neq {}^{\mathcal{C}}\mathbf{y}$ .

In general, the rotation to align a vector  $\mathbf{v}_1$  with a vector  $\mathbf{v}_2$  can be expressed using a quaternion. At first both vectors must be normalized, resulting in the two unit vectors  $\hat{\mathbf{v}}_1$  and  $\hat{\mathbf{v}}_2$ . The rotation axis is perpendicular to both vectors and can thus be calculated using the cross product

$$\mathbf{j} = \frac{\hat{\mathbf{v}}_1 \times \hat{\mathbf{v}}_2}{\|\hat{\mathbf{v}}_1 \times \hat{\mathbf{v}}_2\|}. \quad (5.1)$$

The angle between two vectors can be calculated with

$$\cos(\alpha) = \frac{\hat{\mathbf{v}}_1 \cdot \hat{\mathbf{v}}_2}{\|\hat{\mathbf{v}}_1\| \cdot \|\hat{\mathbf{v}}_2\|} = \hat{\mathbf{v}}_1 \cdot \hat{\mathbf{v}}_2 \implies \alpha = \arccos(\hat{\mathbf{v}}_1 \cdot \hat{\mathbf{v}}_2). \quad (5.2)$$

The denominator is equal to 1 (because the norm of an unit vector is 1) and thus omitted. The quaternion can then be calculated with

$$\mathbf{q} = \begin{bmatrix} \cos\left(\frac{\alpha}{2}\right) \\ \mathbf{j} \cdot \sin\left(\frac{\alpha}{2}\right) \end{bmatrix} \quad (5.3)$$

Using this the quaternion  ${}^{\mathcal{I}}\mathbf{q}$  to transform  ${}^{\mathcal{I}}\mathbf{a}$  onto  ${}^{\mathcal{B}}\mathbf{z}$  has the rotation axis

$$\mathbf{j} = \frac{{}^{\mathcal{I}}\hat{\mathbf{a}} \times {}^{\mathcal{B}}\hat{\mathbf{z}}}{\|{}^{\mathcal{I}}\hat{\mathbf{a}} \times {}^{\mathcal{B}}\hat{\mathbf{z}}\|} = \frac{{}^{\mathcal{I}}\hat{\mathbf{a}} \times \mathbf{e}_z}{\|{}^{\mathcal{I}}\hat{\mathbf{a}} \times \mathbf{e}_z\|} = \frac{{}^{\mathcal{I}}a_y\mathbf{e}_x - {}^{\mathcal{I}}a_x\mathbf{e}_y}{\|{}^{\mathcal{I}}a_y\mathbf{e}_x - {}^{\mathcal{I}}a_x\mathbf{e}_y\|} \quad (5.4)$$

and rotation angle

$$\alpha = \arccos({}^{\mathcal{I}}\hat{\mathbf{a}} \cdot \mathbf{e}_z). \quad (5.5)$$

with  ${}^{\mathcal{I}}\hat{\mathbf{a}} \in \mathbb{R}^{1 \times 3}$  being the measured linear acceleration in the IMU or car frame respectively.

Now that the z-axes of both frames are aligned, the x- and y-axis can be aligned by a rotation around the z-axis. To find this rotation angle  $\beta$ , the car has to be accelerated forward. This acceleration is being measured along the x- and y-axis. The z-axis is set to zero, because it only measures the gravity which is not of interest. The resulting vector is being aligned with the forward axis of the car  ${}^{\mathcal{C}}\mathbf{x} = \mathbf{e}_x = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^\top$ , in the same way as before. Resulting in the rotation angle

$$\beta = \arccos({}^{\mathcal{B}}\hat{\mathbf{a}} \cdot \mathbf{e}_x) \quad (5.6)$$

The resulting quaternion

$${}^{\mathcal{B}}_{\mathcal{C}}\mathbf{q} = \begin{bmatrix} \cos\left(\frac{\beta}{2}\right) \\ \mathbf{e}_z \cdot \sin\left(\frac{\beta}{2}\right) \end{bmatrix} \quad (5.7)$$

can then be concatenated with the previous quaternion to get the final quaternion

$${}^{\mathcal{I}}_{\mathcal{C}}\mathbf{q} = {}^{\mathcal{B}}_{\mathcal{C}}\mathbf{q} \otimes {}^{\mathcal{I}}_{\mathcal{B}}\mathbf{q}. \quad (5.8)$$

A quaternion of the form  $\mathbf{q} = \begin{bmatrix} w & x & y & z \end{bmatrix}^\top$  can be converted to a rotation matrix with

$$\mathbf{M} = \begin{bmatrix} 1 - 2y^2 - 2z^2 & 2(xy - zw) & 2(xz + yw) \\ 2(xy + zw) & 1 - 2x^2 - 2z^2 & 2(yz - xw) \\ 2(xz - yw) & 2(yz + xw) & 1 - 2x^2 - 2y^2 \end{bmatrix} \quad (5.9)$$

And finally the measurements  $\mathbf{A}$  can be transformed using

$${}^{\mathcal{C}}\mathbf{A} = {}^{\mathcal{I}}\mathbf{M} \cdot {}^{\mathcal{I}}\mathbf{A} \quad (5.10)$$

### 5.1.2 Linear acceleration only

$$\theta = \arcsin\left(\frac{\mathbf{a}_x}{g}\right) \quad (5.11)$$

### 5.1.3 Angular velocity only

### 5.1.4 Complementary filter

## 5.2 LiDAR only

### 5.2.1 Calibration

Same as for the IMU, a transformation from the LiDAR frame to the car frame is necessary. The calibration is very similar to that of the IMU. At first both z-axes will be aligned. This is achieved by detecting the ground plane in the point cloud and finding the transform, such that the normal vector of the plane aligns with the z-axis of the car (the plane gets projected onto the xy-plane of the car frame). This results in the correct pitch and roll angle and the missing yaw angle is assumed to be zero. Should it not be zero it can also

be measured by hand and given as parameter.

For the ground plane detection Random Sample Consensus (RANSAC) [2] is used.

RANSAC is a non-deterministic algorithm to remove outliers and is often used in computer vision. RANSAC can also be used for plane segmentation in 3D point clouds. Consider a point cloud with  $n$  points, where each point  $i$  has the coordinates  $x_i, y_i, z_i$ . In a first step, three random points from the point cloud are selected. Three, because this is the minimum number of points needed for a plane. Now the parameters  $a, b, c, d$  of the plane equation

$$ax + by + cz + d = 0 \quad (5.12)$$

can be calculated. Then for every other point the deviation from the proposed plane can be calculated with

$$dist = \frac{ax_i + by_i + cz_i + d}{\sqrt{a^2 + b^2 + c^2}} \quad (5.13)$$

and is then summed up. If the distance is within a certain threshold, the point counts as an inlier. After iterating through the whole point cloud, the number of inlier points and their coordinates are stored. This process is then repeated again until the maximal number of iterations are reached. The plane with the greatest number of inliers is then selected.

Then the normal vector of the plane, which can be conducted from the plane equation as follows

$$\mathbf{n} = (a \ b \ c)^\top \quad (5.14)$$

is projected onto the z-axis of the car. The necessary rotation is then applied to the detected plane. Now that a plane has been found it must be ensured, that it really is the ground plane. Typically either the ceiling, ground or a side wall gets detected with RANSAC. The greater the plane is (or the more points lie inside a plane), the more likely is the detection of the plane. Due to the mounting of the lidar, the ceiling usually does have the most points and is thus detected in the first iteration.

An accidental ceiling detection can be prevented by looking at the average z-values of the detected plane. Because the lidar is mounted on the roof of the car, the z-values of the detected plane must be negative. If they are positive, the ceiling has been detected. Furthermore it is known, that the WHICH angle to rotate the lidar to level ground should not exceed the mount angle. If that is the case, most likely a side wall has been detected. If either condition has not been fulfilled, the detected plane does get removed from the point cloud and using RANSAC a new ground plane estimation is made and validated. This process is repeated until both conditions are fulfilled. The yaw angle and the x- and y-translation from the LiDAR to car

point at the front of the car, centered laterally (same y-coordinates as base\_link, but z at ground level and x further in front)

must be entered manually, but the pitch and roll angle and the distance from the LiDAR to the ground are used from the calibration. A flow chart of the described algorithm is shown in figure 5.4.

Flow chart for calibration can probably removed, flowchart for algo should be enough (very similar)

### 5.2.2 Algorithm

Because the raw LiDAR data is too big to allow for real time processing, preprocessing is necessary. It consists of a passthrough filter to remove unwanted points (e.g. behind the car) and a voxel grid filter to downsample the point cloud. Before the passthrough filter can be applied, the point cloud must be transformed to the car frame. The in the previous

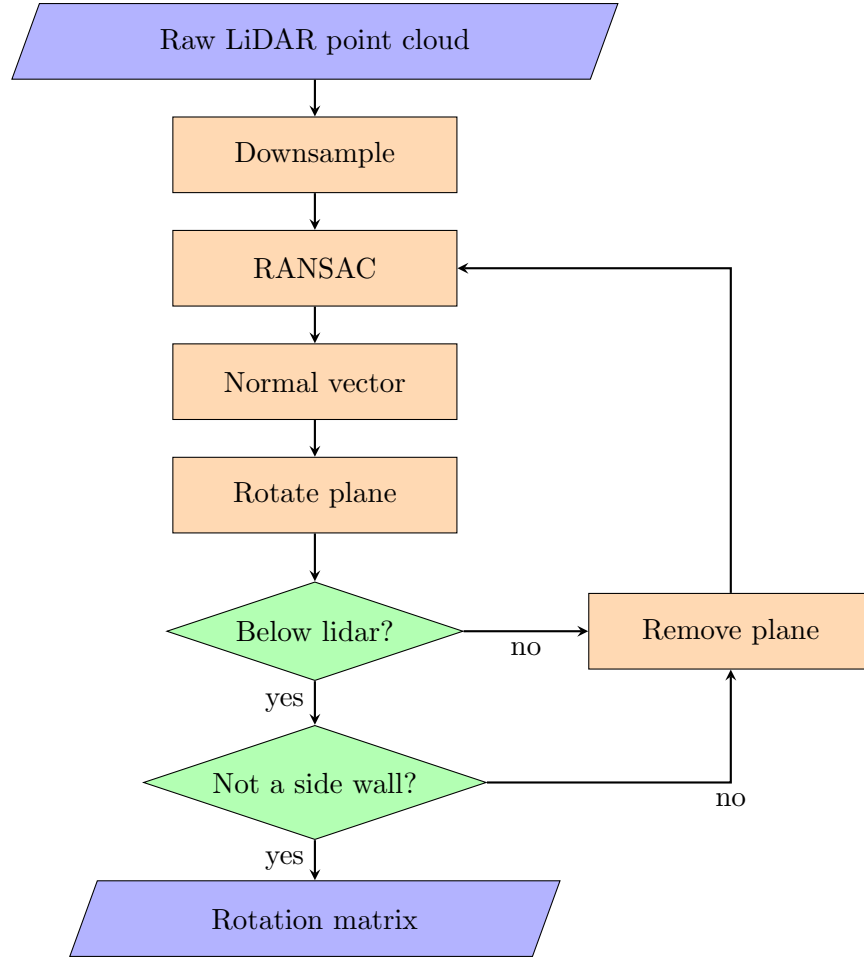


Figure 5.4: Algo for LiDAR alignment

section described calibration algorithm is performed once at the start and its returned rotation is then applied to every new measurement.

The passthrough filter then removes all the points which lie outside the specified x, y and z limits. Because the car drives forward, only points in front of the car are of interest. Furthermore the points further away than a certain threshold are neglected, because the resolution and accuracy of the measurements of the LiDAR decrease with increasing distance. The ceiling points are removed by limiting in z-direction. The exact values used for the passthrough filter can be seen along the other parameters in table 5.1.

The next step in reducing the point cloud size is the voxel grid filter [22]. The point cloud is converted into a 3D grid consisting of small cubes called voxels. Each cube can contain multiple points or none, the size of the voxels (also known as leaf size) determines the resolution. All the points inside a cube are then reduced to their most centroid point. If the cube does not contain any points, it is neglected.

Now that the point cloud size is reduced greatly the actual ramp detection can be performed with sufficient performance. The RANSAC algorithm usually detects the following types of planes: ceiling, ground, side wall or the desired ramp. RANSAC is applied iteratively until a plane of type ramp has been found, each time a different type of plane has been found it gets removed. To prevent an infinite loop the algorithm will exit after either a certain number of iterations has been performed, or if after the removal of a plane not enough points are left in the point cloud.

The accidental detection of the ceiling was already prevented during the passthrough filter step, where the ceiling points have been removed from the point cloud. By limiting

the maximal angle between the ground plane and the detected plane the side walls are ignored. Similarly, the detected plane gets classified as ground plane, if the angle between the ground plane and the detected plane is near zero (or below the specified minimum angle). Beside the angle, the width of the plane (y-range) is being calculated and compared to the parameters, to make sure that the plane is indeed a drivable ramp for cars and not e.g. a small ramp for wheel chairs . If a ramp has been detected, the estimated angle and distance from the car front to the beginning of the ramp are returned.

no idea if wheel ch  
ramps are common  
parking garages?

Add down ramp detection

An visual representation of the algorithm is depicted in fig. 5.5.

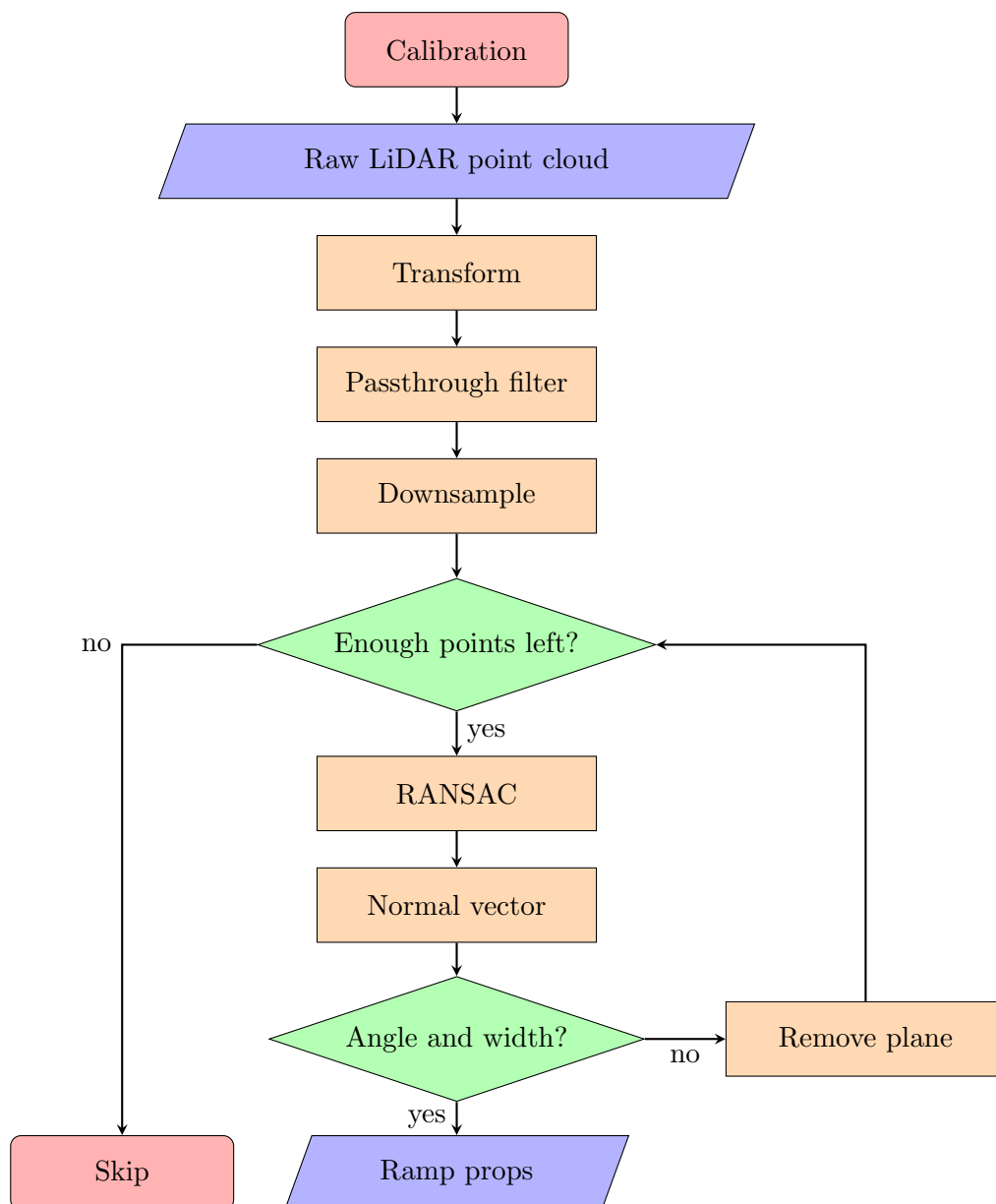


Figure 5.5: Algo for ramp detection

Table 5.1: Used parameters for lidar algo

parameter	value
<b>Passthrough filter</b>	
x	0 m to 30 m
y	−2 m to 2 m
z	−1 m to 2 m
<b>Voxel filter</b>	
leaf_size	0.1 m
<b>RANSAC</b>	
max_iter	100
distance_threshold	0.11 m
normal_distance_w	0.01
<b>rd</b>	
angle	3° to 9°
width	2 m to 6 m
<i>o</i>	4

### 5.3 Camera only

### 5.4 Sensor fusion

Brief explanation what sensor fusion is and why useful

#### 5.4.1 IMU and Odometer

##### Car acceleration from odometer data

Because the odometer only delivers the speed of each wheel, the car velocity has to be calculated first. During turns the left and right wheels travel at different speeds, the wheel on the inner side of the turn travels slower, than the outer wheel. E.g. during a left turn, the left wheel moves slower than the right wheel. A simple yet sufficiently accurate model to calculate the car velocity from the wheel speeds is the linear single track model ("Einspurmodell") [8]. In this model both wheels on one axis are replaced with one wheel in the middle.

Explain model

The linear assumption holds true for low lateral accelerations (up to  $4 \text{ m s}^{-1}$ ), which will not be surpassed in the parking garage scenario. Using the assumptions from above, the car velocity  $v_{\text{car}}(t)$  can be calculated with

$$\alpha(t) = \frac{v_{\text{rl}}(t) - v_{\text{rr}}(t)}{d} \quad (5.15)$$

$$\gamma(t) = \frac{\alpha(t)}{f_{\text{odom}}} \quad (5.16)$$

$$v_{\text{car}}(t) = \frac{v_{\text{rl}}(t) + v_{\text{rr}}(t)}{2} \cdot \cos(\gamma) \quad (5.17)$$

with  $v_{\text{rl}}$  and  $v_{\text{rr}}$  being the wheel speeds of the rear right and rear left wheel respectively.  $\alpha$  is a helper variable,  $d$  the track width! and  $\gamma$  is the yaw angle of the car.

can't explain it well



The car's acceleration can be derived from the velocity using

$$a_{\text{car}}(t) = \frac{d}{dt}v_{\text{car}}. \quad (5.18)$$

But because all measurements are discrete, numerical differentiation e.g. forward difference must be used

$$a_{\text{car}}(h) = \frac{v_{\text{car}}(x+h) - v_{\text{car}}(x)}{h} \quad (5.19)$$

with  $h$  being the step size, which depends on the rate of the sensor.

### Gravity method

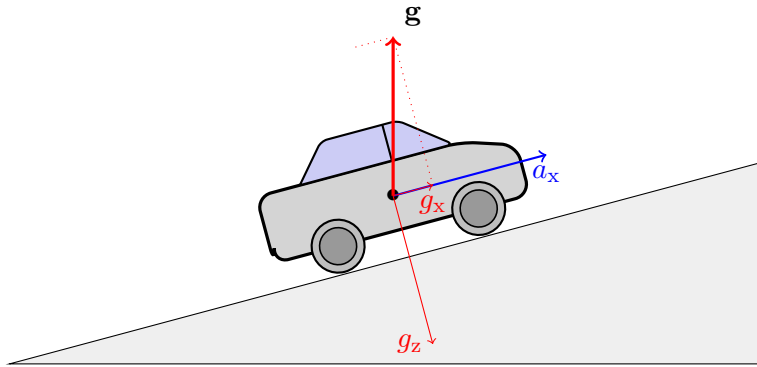


Figure 5.6: Gravity measured by IMU (in car frame). Show that  $g_z$  etc

$$\alpha = \arcsin\left(\frac{a_x - a_{at}}{g}\right) = \arcsin\left(\frac{a_x - \frac{dv}{dt}}{g}\right) = \arctan 2(a_x, a_z) \quad (5.20)$$

- Algo description
- Problems

#### 5.4.2 IMU and odometer and LiDAR

- Algo description

# Chapter 6

## Results

### 6.1 Evaluation concept

- Explanation of scores etc
- Taking average for multiple ramps but also one single ramp

### 6.2 Ramp metering? (IMU)

- How well do different IMU methods work...
- ... at ramp detection
- ... at ramp distance measuring
- ... at angle estimation

### 6.3 Ramp detection (LiDAR and camera)

- Confusion matrix or similar (false negatives could be hard (e.g. curve))
- Estimated angle and distance compare to ground truth
- Do downwards ramp work?
- Camera LiDAR projection for nice visualization
- Something about camera

## Chapter 7

# Conclusion

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

# Bibliography

- [1] Holger Banzhaf, Dennis Nienhüser, Steffen Knoop, and J Marius Zöllner. The future of parking: A survey on automated valet parking with an outlook on high density parking. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1827–1834, 2017.
- [2] Martin A Fischler and Robert C Bolles. Random Sample Paradigm for Model Consensus: A Application to Image Fitting with Analysis and Automated Cartography. *Graphics and Image Processing*, 24(6):381–395, 1981.
- [3] Wenpei He and Junqiang Xi. A Quaternion Unscented Kalman Filter for Road Grade Estimation. *IEEE Intelligent Vehicles Symposium, Proceedings*, (Iv):1635–1640, 2020.
- [4] Jens Jauch, Johannes Masino, Tim Staiger, and Frank Gauterin. Road grade estimation with vehicle-based inertial measurement unit and orientation filter. *IEEE Sensors Journal*, 18(2):781–789, 2018.
- [5] Hakan Koyuncu and Shuang Hua Yang. A Survey of Indoor Positioning and Object Locating Systems. *International Journal of Computer Science and Network Security (IJCSNS '10)*, 10(5):121–128, 2010.
- [6] You Li and Javier Ibanez-Guzman. Lidar for Autonomous Driving: The Principles, Challenges, and Trends for Automotive Lidar and Perception Systems. *IEEE Signal Processing Magazine*, 37(4):50–61, 2020.
- [7] Khalil Maleej, Souso Kelouwani, Yves Dubé, and Kodjo Agbossou. Event-based electric vehicle mass and grade estimation. *2014 IEEE Vehicle Power and Propulsion Conference, VPPC 2014*, (i), 2015.
- [8] Manfred Mitschke and Henning Wallentowitz. *Dynamik der Kraftfahrzeuge*. Springer Fachmedien Wiesbaden, Wiesbaden, 2014.
- [9] Mattias Nilsson and Erik Öhlund. Estimation of road inclination, 2012.
- [10] Busra Ozdenizci, Vedat Coskun, and Kerem Ok. NFC internal: An indoor navigation system. *Sensors (Switzerland)*, 15(4):7571–7595, 2015.
- [11] N. Palella, L. Colombo, F. Pisoni, G. Avellone, and J. Philippe. Sensor fusion for land vehicle slope estimation. In *2016 DGON Inertial Sensors and Systems (ISS)*, pages 1–20. IEEE, sep 2016.
- [12] Michael Perlmutter and Stephen Breit. The future of the MEMS inertial sensor performance, design and manufacturing. *2016 DGON Inertial Sensors and Systems, ISS 2016 - Proceedings*, pages 1–12, 2016.
- [13] C Rev. VLP-32C User Manual.
- [14] RoboSense. RS-Bpearl User Manual. (v.2.0.1), 2020.

- [15] Jihan Ryu and J. Christian Gerdes. Integrating inertial sensors with Global Positioning System (GPS) for vehicle dynamics control. *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, 126(2):243–254, 2004.
- [16] Per Sahlholm and K. Henrik Johansson Karl. Road grade estimation for look-ahead vehicle control using multiple measurement runs. *Control Engineering Practice*, 18(11):1328–1341, 2010.
- [17] Per Sahlholm, Henrik Jansson, Ermin Kozica, and Karl Henrik Johansson. A SENSOR AND DATA FUSION ALGORITHM FOR ROAD GRADE ESTIMATION. *IFAC Proceedings Volumes*, 40(10):55–62, 2007.
- [18] Allianz SE. A sudden bang when parking, 2015.
- [19] Chouki Sentouh, Saïd Mammar, and Sébastien Glaser. Simultaneous vehicle state and road attributes estimation using unknown input Proportional-Integral observer. *IEEE Intelligent Vehicles Symposium, Proceedings*, pages 690–696, 2008.
- [20] Derek K. Shaeffer. MEMS inertial sensors: A tutorial overview. *IEEE Communications Magazine*, 51(4):100–109, 2013.
- [21] G. Vaughan. *Laser Remote Sensing*, volume 21. 2006.
- [22] George Vosselman, Ben Gorte, George Sithole, and Tahir Rabbani. Recognising structure in laser scanner point clouds. *International archives of photogrammetry, remote sensing and spatial information sciences*, 46(8):33–38, 2004.
- [23] Dingkan Wang, Connor Watkins, and Huikai Xie. MEMS mirrors for LiDAR: A review. *Micromachines*, 11(5), 2020.
- [24] Oliver J Woodman. An introduction to inertial navigation. Research report 696, University of Cambridge, aug 2007.
- [25] Behdad Yazdani Boroujeni and H. Christopher Frey. Road grade quantification based on global positioning system data obtained from real-world vehicle fuel use and emissions measurements. *Atmospheric Environment*, 85:179–186, 2014.

## Chapter 8

# Appendix

Code extracts and extra plots etc