



Renato
Miguel Afonso

LiDAR no domínio do espaço

Spatial domain LiDAR



universidade de aveiro
theoria poiesis praxis



Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Eletrónica e Telecomunicações, realizada sob a orientação científica do Professor Doutor Paulo Monteiro, Professor Associado do DETI, Doutor Miguel Drummond, Investigador Auxiliar no Instituto de Telecomunicações, e Doutor André Albuquerque, Optical Designer na Bosch Car Multimedia Portugal, S.A..

o júri/the jury

presidente / president

Anibal Manuel de Oliveira Duarte

Professor Catedrático da Universidade de Aveiro

arguente / examiner

Maria do Carmo Raposo de Medeiros

Professora Associada da Universidade de Coimbra

vogais / examiners committee

Paulo Miguel Nepomuceno Pereira Monteiro

Professor Associado da Universidade de Aveiro

**agradecimentos/
acknowledgements**

É com muito gosto que aproveito esta oportunidade para agradecer a toda a minha família, professor Paulo Monteiro, Miguel Drummond e Abel Lorences pela paciência e disponibilidade, ao João Prata e André Albuquerque, assim como aos meus amigos portugueses e italianos pela companhia e amizade.

Agradeço, também, aos projetos Landmark (POEI-01-0145-FEDER-031527), RETIOT (POCI-01-0145-FEDER-016432) e SOFTLI, assim como ao Instituto de Telecomunicações, Pólo de Aveiro.

Resumo

Este documento apresenta o trabalho desenvolvido no âmbito da tese de mestrado "Spatial domain LiDAR". O objetivo deste projeto é explorar uma abordagem alternativa para os atuais LiDARs baseados em "Time-of-flight". O caminho alternativo consiste em iluminar um cenário com uma matriz de pontos, estimando-se a distância a cada ponto através de triangulação.

O LIDAR investigado é validado através de uma ferramenta de simulação feita especificamente para esta tese. A precisão da estimativa é avaliada comparando-se diferentes métodos de estimação do centróide, investigando quão relevantes são os prejuízos de configuração escolhida e observando os limites de desempenho prático e teórico.

Novas soluções são propostas, ou seja, um método de super-resolução de modo a obter uma maior precisão, uma configuração com uma só câmera, e um método para desacoplar a estimativa da distância da iluminação, fornecendo, ao mesmo tempo, robustez contra a interferência.

Também é apresentada uma validação experimental preliminar, que produziu uma nuvem de pontos correta de um cenário de teste.

Este trabalho mostra que a alternativa explorada LiDAR é viável, sendo uma opção a considerar para a indústria automóvel.

Abstract

This document presents the work developed on the scope of a master thesis on "Spatial domain LiDAR". The goal of this project is to explore an alternative path to mainstream LiDARs based on time-of-flight. The alternative path consists in illuminating a scene with a dot matrix, estimating the distance to each dot via triangulation.

The investigated LiDAR is validated using a custom-made ray optics simulation tool. Estimation accuracy is assessed by comparing different centroid estimation methods, investigating how relevant are setup impairments, and looking at both practical and theoretical performance limits.

New solutions are proposed, namely a super-resolution method for increasing accuracy, a single-camera setup, and a method for decoupling estimation from illumination, at the same time providing robustness against interference.

A preliminary experimental validation is also presented, which produced a correct point cloud of a test scenario.

This work shows that the explored alternative LiDAR is viable, being an option to consider for the automotive industry.

Contents

Contents	i
List of Figures	iv
Glossary	viii
1 Introduction	1
1.1 Key sensors for autonomous vehicles	2
1.1.1 Camera	2
1.1.2 RADAR	2
1.1.3 LiDAR	3
1.1.4 Interplay between sensors	3
1.2 Roadmaps of key automotive companies for ADAS	4
1.2.1 Tesla	4
1.2.2 Audi	5
1.2.3 Toyota	6
1.2.4 The importance of LiDAR	7
1.3 Scope and motivation	7
1.4 Objectives	8
1.5 Contributions	9
2 State of the art	11
2.1 Range estimation techniques	11
2.1.1 Coincidence range finder	11
2.1.2 Parallax	12
2.1.3 Direct time-of-flight	13
2.1.4 Kinect	14
2.1.5 Intel RealSense Depth Camera	16
2.2 Automotive LiDAR solutions	16
2.2.1 Velodyne	17
2.2.2 LeddarTech	18
3 Ray tracing simulation tool	21
3.1 Theoretical background	21
3.2 Overview of the simulation tool	22
3.2.1 Validation	24
3.3 Capabilities	25

4	Triangulation LIDAR	27
4.1	Operation principle	30
4.1.1	Upgraded simulation tool	30
4.2	Centroid estimation	32
4.3	Intensity-independent centroid estimation	32
4.4	Intensity-dependent centroid estimation	33
4.5	Field estimation	34
4.6	Results	35
4.6.1	Ideal centroid estimation	35
4.6.2	Intensity-independent centroid estimation	36
4.6.3	Intensity-dependent centroid estimation	37
4.6.4	Comparison between methods	39
5	Triangulation LIDAR with super-resolution	41
5.1	Operation principle	42
5.2	Centroid estimation methods	42
5.2.1	Intensity-independent centroid estimation	42
5.2.2	Intensity-dependent centroid estimation	43
5.2.3	Results	44
5.3	Camera tilt effect on distance estimation	46
5.3.1	Tilt effect considering one spot per beam	46
5.3.2	Tilt effect considering two spots per beam	48
5.4	Single-camera LiDAR	48
5.5	Results	51
6	Decoupling LiDAR receiver from LiDAR transmitter	55
6.1	Operation principle	56
6.2	Types of interference	57
6.2.1	Overlapping interference	57
6.2.2	Non-overlapping interference	58
6.2.3	Isolated interference	59
6.3	Interfering spots: a curse or a blessing?	59
7	Experimental validation	61
7.1	Setup	61
7.2	Camera calibration	63
7.3	Estimating α and β	64
7.4	One spot per beam experiment	66
7.4.1	Centroid estimation	66
7.4.2	Distance estimation error	67
7.5	Two spots per beam experiment	67
7.5.1	Centroid estimation	67
7.5.2	Distance estimation error	68
7.6	Point cloud generation	69
8	Performance limits	71
8.1	Eye safety	71
8.2	Noise and background light	72
8.3	Number of cameras vs. angular resolution	74
8.4	Diffraction limit	74

8.5	Non-ideal spot generation	75
9	Conclusion	77
10	Future work	79
10.1	Simulation	79
10.1.1	Extension to 3D	79
10.1.2	Alternative approaches	79
10.2	Calibration	80
10.2.1	Stereo matching	80
10.2.2	Tilt correction	80
10.3	Extensive experimental validation	80
10.4	Development of a real-time prototype	80
10.5	Devising suitable/adaptable illumination schemes	80
10.6	Automatic calibration	81
10.7	Object ID via intensity analysis	81
	Bibliography	83

List of Figures

1.1	Several levels of autonomy according to the needs a driver must have [2]. . .	1
1.2	ToF working principle, valid for RADAR or LiDAR [6].	2
1.3	Comparison between point-clouds obtained by a LiDAR and a high resolution RADAR [10].	4
1.4	Tesla’s sensors description [12].	5
1.5	Tesla’s front view cameras [12].	5
1.6	Audi’s “Traffic Jam Pilot” computer board interface [14].	6
1.7	Toyota’s sensors for autonomous driving [15].	7
2.1	Soldiers using a coincidence rangefinder in the 1940s [17].	11
2.2	Coincidence rangefinder operation principle [17].	12
2.3	Illustration of stellar parallax, used for estimating the distance to a star [19].	13
2.4	Illustrating scheme of the Time-of-flight concept [22].	13
2.5	Digital rangefinder [23].	14
2.6	First generation’s Microsoft Kinect webcam [25].	14
2.7	Comparative table between Azure and the previous Kinect version (Kinect for Windows V2) [26].	15
2.8	Intel RealSense Depth Camera D435 model [28].	16
2.9	Technical specifications of Velodyne’s LiDARs [30].	18
2.10	LeddarTech Vu8: 8-segment Solid-State Light Detection and Ranging (LiDAR) sensor module [32].	19
2.11	Technical specifications of Light Emitting Diode Detection And Ranging (LEDDAR) Vu8 [33].	20
3.1	Scheme illustrating used variables to perform ray tracing for (a) an object close to the lens (point source) and (b) an object far from the lens (collimated beam).	22
3.2	Layout of the simulation tool.	23
3.3	Validation example comprising two collimated beams and its focal point at $d=70$ mm for an aperture spot of 20 mm and focal length of 50 mm.	24
3.4	Validation example comprising two collimated beams, a divergent and converging lens with a focal point at $d=132$ mm.	24
3.5	(a) Ray tracing, (b) representation of the two points coming from the intersection of the marginal rays with the sensor and spot’s diagram at the sensor.	25
4.1	Basic triangulation scheme: z can be estimated if α , β and d are known. . .	27

4.2	Distance estimation error, $\partial z/z$, in percentage (a) considering a variation on α and $\partial\beta = 0$, (b) considering an equal variation on both α and β and (c) considering a variation on d , according to the spot's distance, z , considering $d = 1.2\text{m}$ and a field of $= 5^\circ$	29
4.3	Layout considering each camera's Field of View (FoV) and the setup's dark zone.	30
4.4	New simulation tool layout for simulating a system with two cameras. . . .	31
4.5	General scheme of the setup from the spot hitting an object until estimating the distance, z	31
4.6	Spot diagram for intensity-independent centroid estimation and distances z , for a field of 0°	32
4.7	Spot diagram at the sensor for intensity-dependent scale centroid estimation and several spot distances.	34
4.8	Distance estimation error as a function of the spot's distance for several fields considering an ideal centroid estimation.	35
4.9	(a) Distance estimation error, (b) maximum distance estimation error and (c) maximum distance estimation error for the short range according to the spot's distance for intensity-independent scale and different fields.	36
4.10	Distance estimation error due to an one pixel error on the sensor compared with a no error scenario, both for an ideal centroid estimation and a field of 5°	37
4.11	(a) Distance estimation error, (b) maximum distance estimation error and (c) maximum distance estimation error for the short range according to the spot's distance considering intensity-dependent scale for several fields. . . .	38
4.12	Obtained gain from using an intensity-independent (II) scale compared with an intensity-dependent (ID) one for the centroid's estimation.	39
5.1	General scheme of the setup considering two spots per beam.	42
5.2	Spot diagram at the sensor considering two spots per beam and an intensity-independent centroid estimation for several spot distances, z and a field of 0°	42
5.3	(a) Distance estimation error, (b) maximum distance estimation error and (c) maximum distance estimation error for the short range according to the spot's distance, z , for intensity-independent centroid estimation and several fields.	44
5.4	(a) Distance estimation error, (b) maximum distance estimation error and (c) maximum distance estimation error for the short range according to the spot's distance, z , for intensity-independent scale and two spots per beam for several fields.	45
5.5	Gain from one spot per beam compared with two spots per beam method considering the maximum obtained error and an intensity-dependent scale for the centroid's estimation.	46
5.6	Scheme that illustrates the chosen referential for tilt's definition.	46
5.7	Tilt effect on the distance estimation error according distance for one spot per beam and intensity-dependent estimation.	47
5.8	Tilt effect on the distance estimation error according distance for two spots per beam and intensity-dependent estimation.	48
5.9	Possible combinations of cameras and spot(s) for centroid's estimation. . . .	49

5.10	Distance between the geometrical centroids according the spot's distance for several fields.	49
5.11	Distance between centroids according to the spot's distance for an intensity-dependent scale for the left (a) and right (b) sensors, for several fields. . . .	50
5.12	Distance estimation error according to the spot's distance based on the distance between centroids for a field of 0° and intensity-dependent scale. . . .	50
5.13	Distance estimation error according spatial position of the spot(s) for intensity-dependent centroid estimation, being the common area between the orange and blue lines the viewable area, corresponding each one to each camera's FoV.	51
5.14	Estimated distance according real distance, z for intensity dependent (a) and independent (b) scales considering one and two spots per beam.	52
6.1	Types of interference: (a) interferer (dashed) between a pair of spots, (b) interferer that overlaps with one spot of the pair and (c) single interferer. . .	55
6.2	Diagram illustration of the illumination independent algorithm for processing detected spots.	56
6.3	Spot and pixel intensity diagrams for a pair of spots with an overlapping interferer on both sensors for a field of 5° and $z = 50$ m.	57
6.4	Pair of spots with a non-overlapping interferer on both sensors for a field of 5° and $z = 80$ m.	58
6.5	Single interferer spot on both sensors for a field 5° and $z = 80$ m.	59
7.1	(a) Side and (b) front view of the cameras used in the experimental setup. .	62
7.2	Front view of both cameras on the slider, spaced by 1.2 m.	62
7.3	Laser used to produce a spot [42].	62
7.4	(a) Rear view of the laser containing a Printed Circuit Board (PCB) with a variable resistor that allows to vary the laser power and (b) front view of the laser including the collimator lens and respective cage.	63
7.5	Rear view of both lasers side by side needed to produce a pair of spots spaced by 10 cm.	63
7.6	Calibration procedure: chessboard in several positions on the camera's FoV. .	64
7.7	Scheme illustrating α and β and the auxiliary angle Θ obtained by calculating the angle between two pixels.	65
7.8	Angle between two pixels (vectors) in a given frame [44].	65
7.9	Obtained frames in a dark room containing one spot per beam for $z = 2$ m. .	66
7.10	Distance estimation error in percentage for single spot estimation with a baseline distance of 1.2 m.	67
7.11	Obtained frames in a dark room containing a pair of spots for $z = 3$ m. . . .	68
7.12	Distance estimation error for two spots per beam estimation with a baseline distance of 1.2 m.	68
7.13	Real scenario used for obtaining a point cloud.	69
7.14	Obtained point cloud.	70
8.1	Scale of a laser's power according to how dangerous that would be to a human eye [48].	71
8.2	Laser's retinal hazard regions according to wavelength [7].	72
8.3	Received power according spot's distance (a) considering a continuous emission and considering (b) bursts for different duty-cycles.	74

10.1	Illustration of types of reflection that can occur: specular and difuse (lambertian) reflection [63].	82
------	---	----

Glossary

ADAS	Advanced Driver Assistance Systems
ADC	Analog-to-Digital Converter
AMCW	Amplitude-Modulated Continuous Wave
APD	Avalanche Photodiode
CMOS	Complementary Metal Oxide Semiconductor
DSP	Digital Signal Processing
DOE	Diffracting Optical Element
FoV	Field of View
Fps	Frames per second
HDL	High-Definition LiDAR
IR	Infrared
LEDDAR	Light Emitting Diode Detection And Ranging
LiDAR	Light Detection and Ranging
MPE	Maximum Permissible Exposure
Open CV	Open Computer Vision
OSLO	Optical Software for Layout and Optimization
PCB	Printed Circuit Board
RADAR	Radio Detection and Ranging
RGB	Red Green Blue
ROS	Robot Operating System
SDK	Software Development Kit
SUV	Sport Utility Vehicle
ToF	Time-of-Flight
VGA	Video Graphics Array

Chapter 1

Introduction

According to the “Association for safe international travel”, approximately 1.25 million people die in car accidents each year, being, on average, 3287 deaths a day, and 20-50 million people get injured or disabled [1]. In order to reduce these numbers, self-driving cars have been suggested as a possible solution for this problem considering that most accidents are due to human error. If it could be created a system that would allow a car to drive itself with all the necessary mechanisms and sensors, the number of deaths and accidents on the roads would certainly be reduced, ideally, to a point at which there would not be any accidents at all.

Considering that different cars have different self-driving capabilities, it has become necessary to classify the autonomous driving systems at different levels. Thus, according to the “Society of automotive Engineers”, 6 distinguishable levels of automation on a car were created, allowing to classify its level of automation, ranging from no automation to a fully autonomous car, referring to a system that can drive at least as well as a very skilled human in any situation, as seen on figure 1.1.

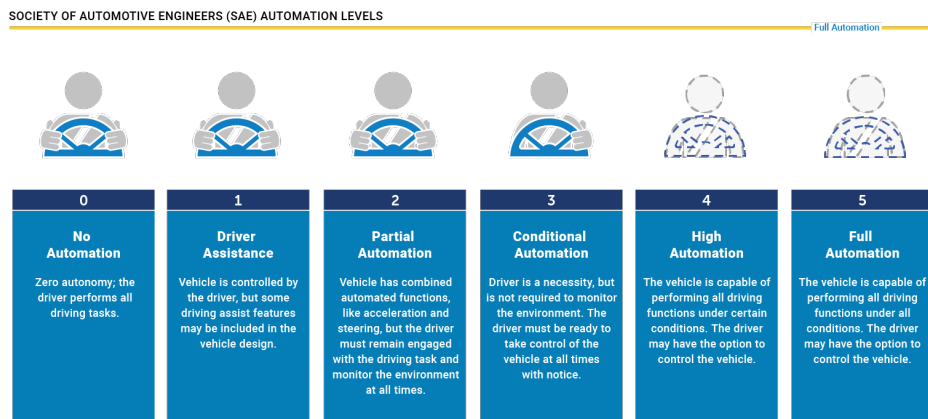


Figure 1.1: Several levels of autonomy according to the needs a driver must have [2].

In order to achieve this goal of having a vehicle that drives itself with no need of human intervention, its system first needs, like a human, eyes, that is, it needs to know what is its surroundings and to be able to discern if what is in front of the car is a human, another car, a wall or, for instance, a traffic signal.

1.1 Key sensors for autonomous vehicles

At the current market, there are three main sensing options for automotive: camera, RADAR and LiDAR. The following sections describe the used sensors on cars for autonomous driving.

1.1.1 Camera

The use of cameras as sensors in cars is already very common, being mainly used for parking assistance. Since 2018, all new vehicles in the US are required to fit rear-cameras as standard, as well as any car with a lane departure warning system will use a front-facing camera to detect painted markings on the road [3]. Autonomous vehicles are no different: almost all prototype vehicles today feature some sort of visible light camera with many others featuring multiple cameras for building a 360° FoV.

Cameras allow a very effective detection and recognition of objects, so, the image data they produce can be fed to AI-based algorithms in order to perform object classification. If there is good illumination and enough resolution, using cameras allows to obtain a very sharp visualization of what is in the FoV, which allows to perform a very accurate object recognition.

Some companies, such as Mobileye [4], rely on cameras for almost all of their sensing. However, just like a person's eyes, visible light cameras have limited capabilities in conditions of poor visibility. Additionally, in case there are used multiple cameras, it generates a lot of video data to process, which requires substantial computing power. Beyond visible light cameras, there are also infrared cameras, which offer superior performance in darkness and additional sensing capabilities like measuring an object's temperature.

1.1.2 RADAR

RADAR is a detection system that uses radio waves to determine the range, angle, or velocity of an object. It can be used to detect air-crafts, ships, spacecrafts, guided missiles, weather formations and terrains. A Radio Detection and Ranging (RADAR) system consists of a transmitter producing electromagnetic waves in the radio or microwave domains, transmitting and receiving antennas (often the same antenna is used for transmitting and receiving), a receiver and a processor. Radio waves (pulsed or continuous) from the transmitter reflect off the object and return to the receiver, giving information about the object's location and speed [5].

One way to obtain a distance measurement using RADAR is based on Time-of-Flight (ToF). It consists on transmitting a short pulse and measure the time it takes for the reflection to return, τ . The distance is one-half of the round trip time multiplied by the propagation velocity, i.e., the speed of light. The factor of one-half comes from the fact that the signal has to travel to the object and back again, as illustrated on figure 1.2.

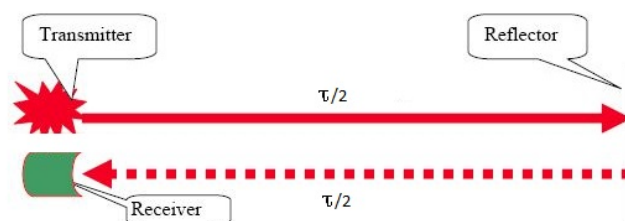


Figure 1.2: ToF working principle, valid for RADAR or LiDAR [6].

1.1.3 LiDAR

LiDAR stands for Light Detection and Ranging and it is a remote sensing technology based on laser light to collect measurements, also using ToF to estimate an obstacle's distance.

LiDAR systems for automotive must emit laser beams at eye-safe levels [7]: laser sharp beams hit objects in the environment and bounce back to a photo-detector, originating a three-dimensional image of the environment [3], allowing the creation of a very sharp point cloud. A dense point cloud provides highly valuable information as it allows the vehicle to sense everything in its environment with a very high precision, being it vehicles, buildings, pedestrians or animals. Hence why so many development vehicles feature a large 360° rotating LiDAR sensor on the roof, providing a complete view of their surroundings. While LiDAR is a powerful sensor, it is also the most expensive one in use. Some of the high-end sensors run into thousands of dollars per unit, but there are also companies, like LeddarTech, that have LiDARs at 100\$ but with a very low resolution (four points in a FoV of 15°), meaning that there is always a compromise between obtaining a detailed point cloud and the price of the sensor.

1.1.4 Interplay between sensors

Comparing with LiDAR systems, cameras are much more mature and less expensive, which brings down costs of self-driving cars, especially for end-consumers. They are also easier to incorporate (for example, Tesla's cars have eight "hidden" cameras around the car) since video cameras are already on the market with several sizes and formats. In general, whatever a human can see, so can a camera-based system. Cameras can also see the world in the same way as humans do and can, in theory, read street signs, interpret colors, and other features, by using computer vision algorithms. Finally, cameras can easily be incorporated into the design of the car, making it more appealing for consumers.

Because visible light cameras use background light as illumination, they can see until a large range of distances during daytime as long as they have a narrow FoV, being limited by the sensor's noise. At night, they must use light from an artificial source, like, for instance, the car headlights. Just as cameras "see" colour, LIDARs just see a gray-scale in the infrared spectrum. Also, nowadays, even cheap cameras with very high resolution are available: where a LIDAR might see 64 lines, a camera can see, for instance, 1200 (corresponds to the number of pixels at vertical on the frame): because of this high resolution and the capacity of discerning colour, camera systems are able to resolve details about the scene that cannot be resolved from a lower-resolution LIDAR.

On the other hand, high-end LiDAR sensors can identify details with a precision of few centimeters at more than 100 m. For example, Waymo's LiDAR [8] system not only detects pedestrians but it can also tell which direction they are facing. Thus, the autonomous vehicle can accurately predict where the pedestrian will walk. The high-level of accuracy also allows it to see details such as a cyclist waving to let someone pass, two football fields away, while driving at full speed. Waymo has also managed to cut the price of LiDAR sensors by almost 90% in the recent years. A single unit with a price tag of \$75,000 a few years ago will now cost just \$4000, making this technology more affordable [9].

LiDARs are, in general, much more expensive because of optics: lens, collimators, prisms, arrays of lasers and photo-diodes, and other optical elements that are typically much more expensive than non-optics RADAR counterparts. In a RADAR, emission and detection typically require a single antenna, which is much less expensive. Cost is one of the reasons why Tesla has chosen RADAR over LiDAR, working just as well in all weather conditions such as fog, rain, snow and dust. Still, due to its lower resolution compared with LiDAR, objects that are placed very close to each other may become indistinguishable, or, at least, less perceptible. For example, it may consider two small cars in the vicinity as one large vehicle, a case that would not happen if using a LiDAR. That difference in resolution is illustrated on figure 1.3, where it is observed that, with LiDAR, it can be obtained a higher level of detail comparing to RADAR.

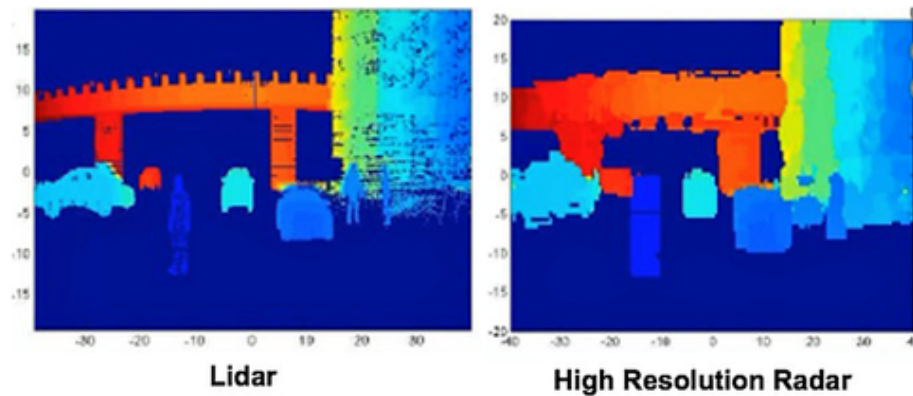


Figure 1.3: Comparison between point-clouds obtained by a LiDAR and a high resolution RADAR [10].

Concluding, LiDAR offers much superior sharpness than RADAR, being proven that it can also achieve long ranges, although, due to the large amount of optical elements used, its key disadvantage is its high price compared to RADARs or cameras.

1.2 Roadmaps of key automotive companies for ADAS

Several major car brands have already implemented in their vehicles different autonomous driving capabilities, continuously investing in this area, being Tesla the most famous one. Still, several other manufactures like Audi or Toyota illustrate the general outlook of the industry, as will be described next.

1.2.1 Tesla

When Tesla’s Autopilot first debuted on October of 2014, drivers could leave their hands off the steering wheel for some time, although government regulations have limited that capability and drivers, now, must make contact with the steering-wheel, every minute or two. Still, when the driver’s hands are off the steering-wheel, Autopilot allows the car to slow-down, speed-up, change lanes and handle most corners and curves, even when road lines are not very reliable [11]. Tesla’s Autopilot hardware suite includes 8 cameras, 12 ultrasonic sensors and a front-facing RADAR [12], as seen on figure 1.4, stating that LiDARs are not necessary for a reliable autonomous driving. Also, the most recent upgrade made to the Tesla’s sensor’s suite is the triple front-facing camera system (which does not use triangulation): the “Main Forward Camera” with a range of 150 m and a FoV of 50°, a

“Narrow Forward Camera” with a range of 250 m and a FoV of 35° , and a “Wide Forward Camera” with a range of 60 m and a FoV of 150° , being these front-facing cameras housed in the rear-view mirror cut-out, as seen on figure 1.5.

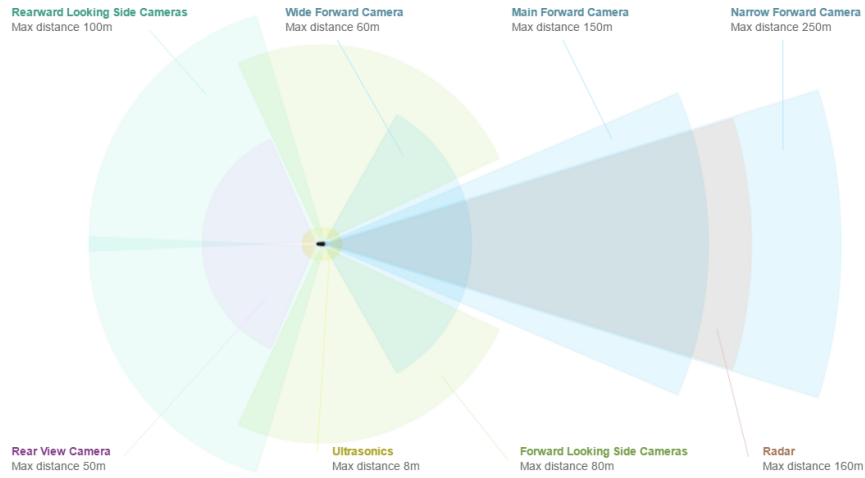


Figure 1.4: Tesla’s sensors description [12].

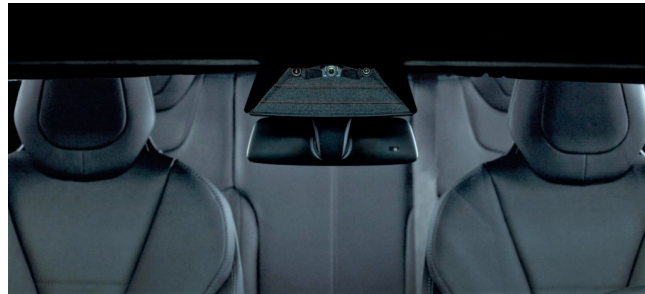


Figure 1.5: Tesla’s front view cameras [12].

1.2.2 Audi

Another company that is investing a lot on autonomous driving is Audi, with the model A8: its “Traffic Jam Pilot” is not a fully autonomous or automated technology that is designed to be used in all cases, but it is a system that can take over in relatively simple situations, being functional below 60 km h^{-1} .

A series of sensors (ultrasonic and radar), several cameras and LiDAR (Valeo-Scala [13]) are combined to create the first series production car to meet Level 3 automation standards. The system can steer, accelerate, brake and even come to a complete stop and start up again so the driver does not have to constantly handle the brakes, as well as make minor adjustments to the steering wheel, as illustrated on figure 1.6.

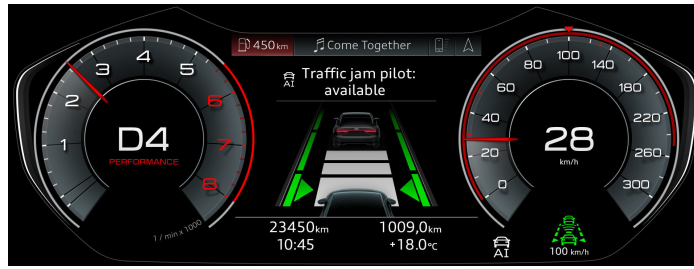


Figure 1.6: Audi’s “Traffic Jam Pilot” computer board interface [14].

The driver activates the “Traffic Jam Pilot” with the “AI button” on the center console. On freeways and highways, where a physical barrier separates the two carriageways, the system takes over the driving task in traffic jam and slow-moving traffic, as well as any other situation, being the main limitation the referred speed limit.

1.2.3 Toyota

Another manufacturer that is betting on autonomous driving is Toyota, with its Cruise AV model [15], that has 5 LiDARs, 16 cameras and 21 RADARs that are strategically placed to ensure that it can scan on a FoV of 360°.

The large number of used sensors is for the required redundancy for both long and short range views necessary for a Level 4 autonomy, as illustrated on figure 1.7. The LiDAR provides precise data on both fixed and moving objects around the vehicle and the RADAR adds data on objects with low light reflectivity. Both LiDAR and RADAR inputs are used for measuring the speed of moving objects. The cameras are used to measure the light intensity reflected off or emitted from objects for rich detail. The LiDAR and camera data are combined for classifying and tracking objects to make a faster high-confidence recognition. A complementary set of long-range sensors tracks high-speed objects, such as incoming vehicles, and the short-range sensors provide detail about moving objects near the vehicle such as pedestrians or bicycles. From their data, a three-dimensional model and map of the surroundings of the vehicle are built.

The Cruise AV has two main computer systems (a primary and a backup) that operate independently and simultaneously for self-driving decision-making. However, if a malfunction occurs in both, a diagnostics system determines whether the appropriate response is a fail-operational state or a fail-safe state, transitioning the vehicle to the corresponding safe state.

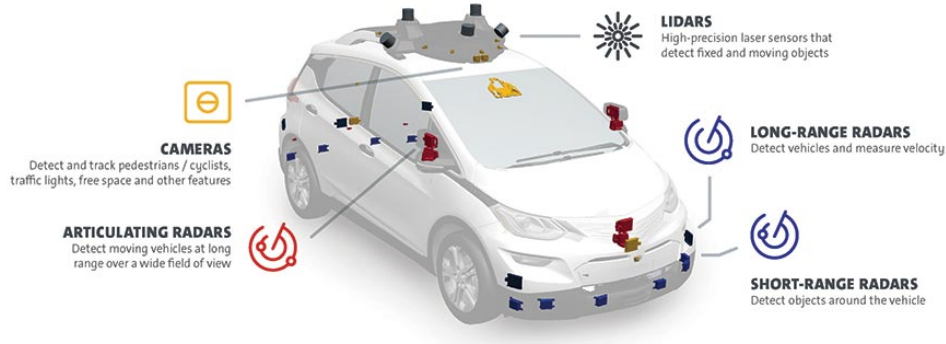


Figure 1.7: Toyota's sensors for autonomous driving [15].

In conclusion, several brands are investing a lot in these technologies, with Tesla being the main protagonist, using only RADAR and discarding completely the use of LiDAR saying that RADAR and cameras are sufficient by themselves.

1.2.4 The importance of LiDAR

One of the main advantages of LiDAR is its temporal and spacial accuracy and precision, as well as that it allows to provide self-driving cars a three-dimensional map to work with, that is, the point cloud. A further advantage in relation, for example, to cameras is that LiDARs have their own illumination, not depending on external light sources. It is also very reliable at detecting objects of several sizes and as different ranges, that is, determining its position.

As referred, the main disadvantage of LiDAR is its cost. Google's system originally ran upwards of \$75,000. Today's startups are bringing the costs of LiDAR units down to below \$4,000, but that is still relatively expensive considering one might need several units around the car. Until LiDAR costs come down to more reasonable levels, self-driving cars based on these systems may remain too expensive.

Interference is another potential issue with LiDAR: if there is a large number of vehicles generating laser pulses at the same time, it could cause interference and potentially put LiDARs out of order. Manufacturers will need to develop methods to prevent this interference.

1.3 Scope and motivation

On the current market, major companies like Velodyne or LeddarTech rely on ToF sensors. Still, using this type of sensors has some disadvantages such as guaranteeing the synchronism between the transmitter and the receiver so that a large precision can be obtained. For example, a precision of 1 cm requires a clock resolution of 30 ps, that is, a sample-rate of ~ 30 GHz. Also, Velodyne uses pulsed lasers that require complex drivers and resorts to physical rotation for scanning purposes. Still, the main problem of

the products offered by these companies is their very high price, which makes self-driving cars still very expensive, as well as the high power consumption of these systems in a car. As referred, the high price is due to the complex and numerous optical elements of these products. Ramping up the production will certainly lower prices. However, unless a disruptive idea is introduced, LIDARs will still be prohibitively expensive. This, in turn, will make autonomous cars too expensive for the average consumer, at least for the short term.

For these reasons, the proposed approach offers a simpler, much less expensive, still functional, way of estimating an object's distance and, consequently, obtaining a point-cloud of a scene, simply by using a setup with two cameras and two lasers. Instead of using ToF, this new approach consists on using geometry concepts to perform triangulation, allowing, this way, to determinate each point's distance.

The simplicity and consequent substantial low cost of this approach make it a very interesting solution considering the competing products on the market.

1.4 Objectives

The main motivation of the developed work throughout this thesis was to investigate an alternative LiDAR technique based on triangulation and structured light illumination.

Several objectives were pursued, as subsequently listed:

1. Develop a ray-tracing simulation tool in order to visualize the image produced on a camera's sensor by a point source at a chosen distance and field;
2. Numerically validate the triangulation concept considering ideal parameters;
3. Perform a realistic simulation considering intensity independent or dependent scales and a discrete number of pixels on the sensor;
4. Find the best method to estimate the centroid of a spot and obtain the distance estimation accuracy as a function of the spot's distance and field;
5. Investigate the influence of considering pairs of spots instead of a single spot for improving the accuracy;
6. Investigate a way of using triangulation using a single camera and pairs of spots;
7. Study the influence of interfering spots from other LiDARs and devise methods for dealing with interference;
8. Experimentally validate the proposed concept considering a single spot or pairs of spots and compare the obtained results with numerical ones;
9. Produce a point-cloud.

1.5 Contributions

The main contributions of the described work in this thesis are:

1. Simple yet comprehensive ray optics simulation tool that allows investigating realistic setups;
2. Numerical demonstration of a super-resolution concept based on illuminating an object with a pair of spots;
3. Method for decoupling the receiver from the illumination pattern (transmitter), thus enabling illumination-independent range estimation;
4. Algorithm that takes advantage of interfering spots.

Chapter 2

State of the art

For a very long time humans have tried to find ways of measuring distances. This chapter describes some relevant distance measuring techniques.

2.1 Range estimation techniques

2.1.1 Coincidence range finder

One of the first methods to estimate distances was the coincidence rangefinder, created for war purposes, as illustrated on figure 2.1. A coincidence rangefinder is a type of rangefinder that uses mechanical and optical principles to allow an operator to estimate the distance of an object [16].



Figure 2.1: Soldiers using a coincidence rangefinder in the 1940s [17].

This device consists of a long tube with two lenses facing forwards at each end and an operator eyepiece in the center. Light from the target enters the range finder through two windows located at either ends of the instrument, as illustrated on figure 2.2. At either side the incident beam is reflected to the center of the optical bar. This reflected beam first passes through an objective lens and is, then, merged with the beam of the opposing side with an ocular prism sub-assembly to form two images of the target, which are viewed by

the observer through the eyepiece. Since either beam enters the instrument at a slightly different angle, the resulting image, if unaltered, will appear blurry.

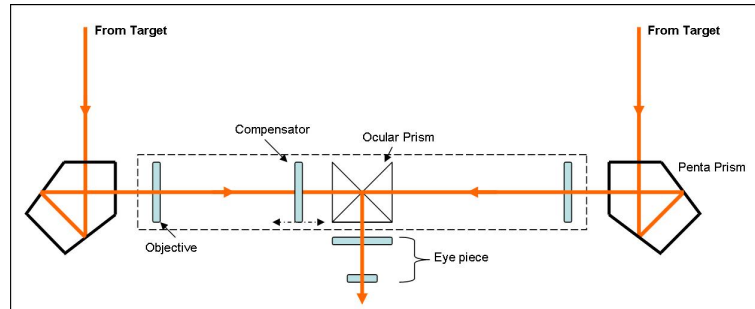


Figure 2.2: Coincidence rangefinder operation principle [17].

Therefore, in one arm of the instrument, a compensator wedge [18] is adjusted by the operator, deviating the beam so that both images exactly coincide in the observer's field. This procedure is, actually, a way of measuring the disparity between two images, which, in this case, results in a greater or lesser degree of blurriness. When there is no blur the images are said to be in coincidence. The degree of rotation of the compensator determines the range to the target by simple triangulation.

Even though functional, the fact that it is a relatively large structure, as well as that this method does not allow very precise measurements. To the best of the author's knowledge, this was the first instrument for estimating range based on disparity, which was transduced to "blurriness". This method has become obsolete by the advent of digital cameras and digital signal processing, namely stereo cameras.

2.1.2 Parallax

Another possible way to estimate the distance of a certain object is by using parallax: parallax consists on the apparent displacement of an object due to a change in the observer's point of view. The displacement that this same point suffers on both observations is related with the object's distance. In this method, each star works as a reference point source, instead of an entire image, allowing to obtain a greater accuracy on the measurements.

This method has already proven to be quite effective, also, because this concept was originally used in astronomy to determine the distance of the planets towards the Earth: as the Earth orbits the Sun, a nearby star will appear to move against the more distant background stars. Astronomers can measure a star's position once and, 6 months later, calculating the apparent change in position. The star's apparent motion is called "stellar parallax", as illustrated on figure 2.3.

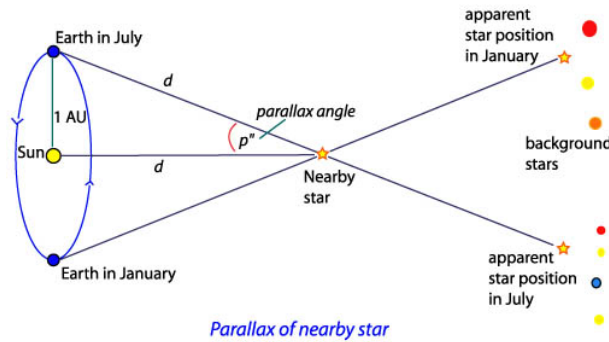


Figure 2.3: Illustration of stellar parallax, used for estimating the distance to a star [19].

2.1.3 Direct time-of-flight

As seen, the coincidence rangefinder can give an idea of an object's distance, still, it is a very slow way to do it, having led to the need of finding a more precise and faster way for estimating a distance.

The pioneer in optical ToF measurements was Galileo Galilei, who tried to prove this concept by making a simple experiment, described in one of his “Discorsi” written in the early 17th century [20]. At night, he placed two people onto two neighboring hills about 1 km away from each other. Each of them carried a lantern, covered by their hands, so that one could not see the other's lantern light, initially. The experiment started with person A uncovering his lantern and waiting for B to uncover his lantern as well, as soon as he saw the first lantern's light. That way, Galileo hoped to measure the propagation time that the light needed to travel from person A to person B and back to A, a hopeless attempt from today's point of view since it is known that light only needs 6.67 μ s to travel a distance of 2 km.

In 1968, Koechner [21] was a pioneer to introduce an optical ToF ranging system. The direct ToF technique consists on a laser that emits uniform short pulses (same waveform, duration, amplitude and period). The elapsed time between the sending and receiving events, t_D , is estimated and is proportional to the distance to the reflection point, as illustrated on figure 2.4.

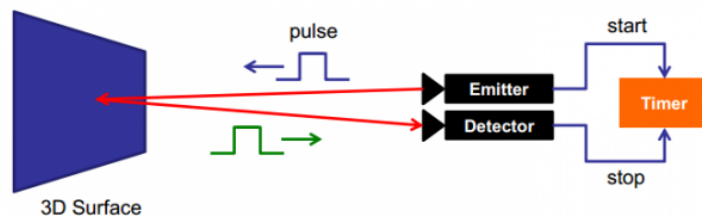


Figure 2.4: Illustrating scheme of the Time-of-flight concept [22].

Since the laser and the detector are practically at the same location, the distance can be directly determined by multiplying the speed of light by the time the impulse took to go back and forth and divide that for two, considering that the distance of the object is the one relative to half of the time of the trip,

$$d = \frac{c \cdot \text{ToF}}{2}. \quad (2.1)$$

The use of ToF technology is useful because the accuracy is distance independent (for pulsed LiDAR), depending only on the timing accuracy, being these devices mainly used for topography. Also, the ToF devices are usually very compact, light-weight and simple to use. On the other hand, the problem of establishing a ToF-ranging system is the realization of a high accuracy time-measurement mechanism: for example, in order to obtain a resolution of 1 cm, an accuracy of approximately 70 ps is required.

Nowadays, rangefinders can fit a person's hands and are very cheap, with prices starting from 20\$, reaching ranges of 100 m with a precision of ± 2 mm, like the one illustrated on figure 2.5.



Figure 2.5: Digital rangefinder [23].

2.1.4 Kinect

Another very popular device whose function is estimating distances and depth maps is the Kinect (was the first product of this type to reach the masses), which is a line of motion sensing input devices produced by Microsoft [24].

The first generation of Kinect was developed as a gaming accessory for Xbox 360, Xbox One video game consoles and Microsoft Windows PCs. Based around a webcam-style, it enables users to control and interact with their console/computer, without the need for a game controller, through a natural user interface using gestures and spoken commands. The Kinect sensor is a horizontal bar supported by a small base, as illustrated on figure 2.6, being designed to be positioned horizontally above or below the video display.



Figure 2.6: First generation's Microsoft Kinect webcam [25].

The first model was called “Kinect for Xbox 360” (2010), which was a combination of Microsoft's built software and hardware: the hardware included an infrared dot projector, a camera and a dot grid generator from which the 3D location of a nearby object can be

estimated. The Kinect contains three vital pieces that work together to detect motion and create an image on the screen: a video camera, a depth sensor and a microphone array which provide full-body 3D motion capture, facial and voice recognition capabilities.

The camera has a RGB sensor with a default pixel resolution of 640×480 pixels but the hardware is capable of resolutions up to 1280×1024 pixels (at a lower frame rate) and other colour formats such as UYVY. It runs on a frame rate of approximately 9 Hz to 30 Hz, depending on the chosen resolution.

The depth sensor contains a monochrome Complementary Metal Oxide Semiconductor (CMOS) sensor with a Video Graphics Array (VGA) resolution (640×480 pixels) and 11-bit depth. It also has an infrared dot projector that helps to create the 3D imagery throughout the room, measuring the distance of each point of the player's body by transmitting invisible near-infrared light and the geometric distortion of the dot's pattern. The sensor has a range of 1.2-3.5 m when used with the Xbox software, with an angular FoV of $57^\circ \times 43^\circ$, being also capable of tilting the sensor up to 27° either up or down. All the referred components come together to detect and track 48 different points on each player's body and repeat it at 30 Frames per second (fps).

The Kinect's second version was released in 2012, having a new hardware version of the device called "Kinect for Windows". The Software Development Kit (SDK) included Windows 7 compatible PC drivers for Kinect devices, providing Kinect capabilities so that developers could build new applications and included features like raw sensor streams, that is, access to low-level streams from the depth sensor, color camera sensor and four-element microphone array.

In 2013, it was released another version called "Kinect for Xbox One". It used wide-angle ToF camera and process over 2 Gbit/s of data. The new Kinect had greater accuracy, with three times the precision over its predecessor, as well as it could track without visible light by using an IR camera, being able to detect several different point of the player's body.

The most recent version of the Kinect is called "Azure Kinect", introduced in 2019, and features a series of major upgrades compared to the previous version, as seen on figure 2.7.

FEATURE		AZURE KINECT DK	KINECT FOR WINDOWS V2
Audio	Details	7-mic circular array	4-mic linear phased array
Motion sensor	Details	3-axis accelerometer + 3-axis gyro	3-axis accelerometer
RGB Camera	Details	3840 x 2160 px @30 fps	1920 x 1080 px @30 fps
Depth Camera	Method	Time-of-Flight	Time-of-Flight
	Resolution/FOV	640 x 576 px @30 fps	512 x 424 px @ 30 fps
Connectivity		512 x 512 px @30 fps	
		1024x1024 px @15 fps	
	Data	USB3.1 gen 1 with Type-C connector	USB 3.1 gen 1
	Power	External PSU or USB-C	External PSU
	Synchronization	RGB & Depth and IMU internal, external device-to-device	RGB & Depth internal only
Mechanical	Dimensions	103 x 39 x 126 mm	249 x 66 x 67 mm
	Mass	440 g	970 g
	Mounting	One ¼-20 UNC Four internal screw points	One ¼-20 UNC

Figure 2.7: Comparative table between Azure and the previous Kinect version (Kinect for Windows V2) [26].

2.1.5 Intel RealSense Depth Camera

Some of the most recent products in the depth camera’s market segment are the “Intel RealSense Depth Cameras”, launched in 2018.

Intel RealSense Technology provides a suite of depth and tracking technologies designed to give machines and devices depth perception capabilities that will enable them to “see” and understand the world. There are many uses for these computer vision capabilities including autonomous drones, robots, smart home devices, amongst many other applications. RealSense technology is made of Vision Processors, Depth and Tracking Modules and Depth Cameras, supported by an open source cross-platform called “librealsense”, that simplifies supporting cameras for third party software developers [27].

The Intel RealSense Depth Camera D435, as seen on figure 2.8, captures stereo depth by illuminating the scene with a laser pattern, having a variety of applications that help perceive the world in 3D. The camera includes the “Intel RealSense Vision Processor D4” featuring high depth resolution up to 1280x720 at 30 frames per second, a maximum range of 10 m, depending on calibration (can be calibrated for different light scenarios), the scene and lighting conditions, global shutter technology and a wide FoV. With the global image shutter and wide FoV ($91.2^\circ \times 65.5^\circ \times 100.6^\circ$ (Horizontal, Vertical, Diagonal)), the Intel RealSense Depth Cameras offer accurate depth perception when the object is moving or the device is in motion and it covers a wide FoV.

It should be noted that both Kinect Azure and the Intel Realsense cameras can also be used for other applications like robotics, to place in drones, domotics and others, reinforcing the wide applicability of these products.

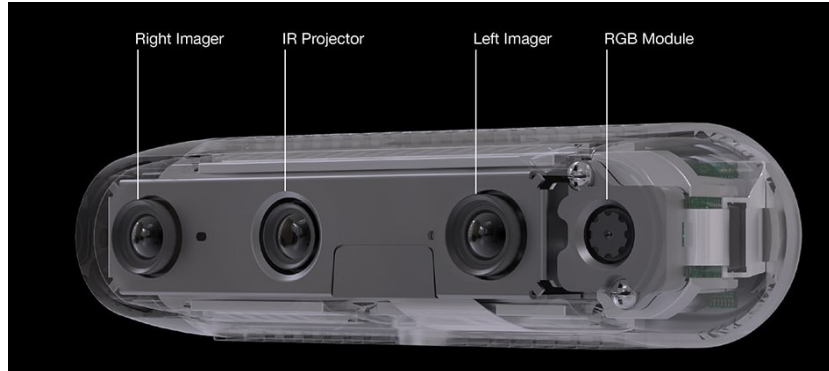


Figure 2.8: Intel RealSense Depth Camera D435 model [28].

2.2 Automotive LiDAR solutions

A new generation of LiDARs has been recently developed thanks to the massive investment in Advanced Driver Assistance Systems (ADAS). In 2018, the LiDAR market was worth 1.3 billion and analysts expect that it will value 6 billion dollars by 2024 [29], with 70% of the total market dedicated to automotive applications.

The previously referred depth estimation products are mainly for gesture recognition, near range and for closed environments. For that reason, companies had to make a significant investment in products targeted for the automotive industry, that is, products that could reach large ranges and process much more information, such as LiDARs.

At the present global market there are several solutions about what concerns LiDAR. Velodyne and LeddarTech are the most prominent at the moment, offering a wide array

of products, but there are also other companies like Quanergy System, Valeo, Continental AG and several start-ups that are investing in this market, being most of their products based on ToF mechanisms. Most of these companies are located in North America, but are also present in European countries like Germany or United Kingdom.

2.2.1 Velodyne

Starting by Velodyne, headquartered in Silicon Valley, this company is commonly referred as one of the pioneers and main competitors on this sector since 2005. Velodyne has an impressive patent portfolio that includes the rotating scanning mechanism (being this one of the major Velodyne's contributions in order to generate point clouds), in which the housing is rotated through a rotary motor to cover a complete 360° horizontal FoV, while the vertical FoV is covered by several discrete and fixed laser-detector channels vertically aligned with co-linear directions, being each point acquired by ToF.

By using several channels, each vertical slice is acquired at once for each angle of the rotating head, resulting in high frame rates and point densities. Each pixel is measured through a single laser pulse and, in order to overcome losses associated with bad weather conditions, two measuring modes are available: single mode, the strongest (default) or the last return is reported while in dual mode, both returns are recorded, which is beneficial when the same pulse is reflected at different points. The detection in each channel is performed via an Avalanche Photodiode (APD) and, after amplification, the signal is sampled using a high-speed Analog Digital Converter (ADC). The ToF is then extracted using digital signal processing algorithms.

The products developed by Velodyne are divided in two main families: High Definition LiDAR (HDL) and VLP, being illustrated a comparison between them on figure 2.9. The VLP (also called Puck due to its smaller size) is characterized by a more compact design, making it suitable for more specific applications. The second group of sensors was designed to satisfy the demands of autonomous vehicle navigation, providing accurate and dense point clouds. As seen on figure 2.9, the HDL-64 offers a range of around 100 m, like most of Velodyne products so far, with an accuracy of 2 cm. Its vertical resolution is approximately 0.4° , better than its pairs that have lower resolutions of 1.3° and 2° . Since all of them are made over rotating mechanisms, the horizontal FoV is 360° , being their horizontal resolution different according the frequency of operation. Still, the main problem inherent to Velodyne and its products is its price: all models are still very expensive, although it is expected the prices to drop as soon as mass production starts. The high price is due to fact that each channel has a dedicated transmitter and APD detector, which puts the price to several thousand dollars. For this reason, the Pucks are the cheapest products, simply because they use less channels.

The most recent product is the VLS-128, which has 128 channels in order to obtain a better vertical resolution, being mainly targeted for high-level autonomous driving, providing large fields of view and high angular resolution.

Velodyne® LiDAR



Features	HDL-64	HDL-32	VLP-16
Channels	64	32	16
Range	100-120 m	80-100 m	100 m
Accuracy	+/- 2cm	+/- 2cm	+/- 3cm
Data	Distance / Intensity	Distance / Calibrated Reflectivities	Distance / Calibrated Reflectivities
Data Rate	1.3M pts/sec	700,000 pts/sec	300,000 pts/sec
Vertical FOV	26.8°	40°	30°
Vertical Resolution	~ 0.4°	~ 1.3°	~ 2.0°
Horizontal FOV	360°	360°	360°
Horizontal Resolution	5Hz: 0.08° 10Hz: 0.17° 20Hz: 0.35°	5Hz: 0.08° 10Hz: 0.17° 20Hz: 0.35°	5Hz: 0.1° 10Hz: 0.2° 20Hz: 0.4°
Input Voltage	10-32 VDC	9-32 VDC	9-32 VDC
Power	60W	12W	8W
Environmental	IP67	IP67	IP67
Operating Temperature	-10° to 50° C	-10° to 60° C	-10° to 60° C
Size	203mm x 284mm (~8" x ~11")	86mm x 145mm (~3.6" x ~6")	104mm x 72mm (~4" x ~3")
Weight	15kg (33lbs)	1kg (2.2lbs)	0.83kg (1.8lbs)

Figure 2.9: Technical specifications of Velodyne's LiDARs [30].

2.2.2 LeddarTech

LEDDAR is a proprietary technology owned by LeddarTech [31]. It uses ToF and Digital Signal Processing (DSP) algorithms to detect, locate and measure an object's distance, the same way as Velodyne does.

The "LEDDAR technology" consists on a LiDAR that sends very short light pulses of invisible light to actively illuminate an area of interest. The sensor captures the light back-scattered from objects over its detection area and processes the signals to precisely map their location. The multi-channel sensor also provides lateral discrimination of detected objects and this feature, with 3D measurements, is the basis for object tracking, being

one of the main LeddarTech's contributions to have separated the transmitter from the receiver.

LeddarTech's main product, the Vu8, illustrated on figure 2.10, consists of a 905 nm laser emitter combined with diffracting optics, that is, a laser diode that emits to a micro-mirror that allows to perform scanning, providing a wide illumination beam that is available in various horizontal FoV configurations from 20° to 100° (each available in different vertical FoVs).

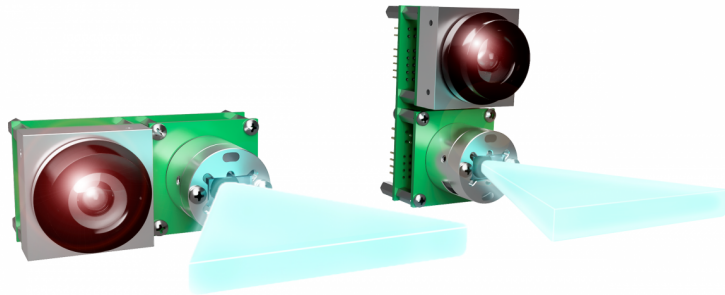


Figure 2.10: LeddarTech Vu8: 8-segment Solid-State LiDAR sensor module [32].

The most appealing feature of this design is the intended modular architecture, achieved through the division into 3 different boards: source, receiver and carrier board. This allows for multiple configurations and specifications that can be adopted by simply swapping the building blocks to afford versatility, adaptability and flexibility [33].

The receiver assembly includes 8 independent detection elements with simultaneous multi-object measurement capabilities supported by powerful LEDDAR signal processing algorithms [32], being the discrete-time signal analyzed and the ToF is directly recovered for every object in the FoV. Its key features are a long range detection, up to about 185 m, compact and lightweight frame, fast refresh rate of up to 100 Hz, proven reliability both indoors and outdoors and various beam options for optimized FoV. Figure 2.11 summarizes the Vu8 characteristics.

LiDAR Class	2D Solid-state Flash
#Lasers / #Detectors	1 / 1D array of 8 independent detectors
FOV (H×V)	Narrow FOV: 20°×0.3° / 20°×3° Medium FOV: 48°×0.3° / 48°×3° Wide FOV: 100°×0.3° / 100°×3°
Max. Range (Reflectivity)	Narrow FOV: 185m (retro-reflector) / 60m (90%) Medium FOV: 118m (retro-reflector) / 31m (90%) Wide FOV: 61m (retro-reflector) / 12m (90%)
Range accuracy/resolution	± 5 cm / 10mm
Frame Rate	Up to 100Hz
Laser Class / Wavelength	Class 1 Eye-safe per IEC 60825-1:2014 / 905nm
Pulse width / PRR	10ns / 10 kHz
Supply Voltage / Power consumption	12±0.6 VDC / 2W
Data per point	ToF distance measurement Intensity/Amplitude
Operating temperature	-40°C ... +85°C
Interfaces	SPI, USB-CAN-Serial
Dimensions (H×W×D)/ Weight	Narrow FOV: 70×35.2×67.5 mm ³ / 110.3g Medium FOV: 70×35.2×45.8 mm ³ / 107.6g Wide FOV: 73×50×65 mm ³ / 128.5g
Price	From \$450 (depends on the modules)

Figure 2.11: Technical specifications of LEDDAR Vu8 [33].

Thus, comparing Vu8 with Velodyne, more precisely, the HDL-64, the Velodyne wins concerning the maximum range, even though the Vu8 can reach a range of 185 m but with a narrower FoV. On the other hand, the HDL-64 has an effective maximum range of 120 m, but it should be taken in account that these results depend of the used setup that each company defined. Also, an advantage of Velodyne's products is their proprietary rotating mechanism, that is, it is always guaranteed a 360° horizontal FoV when LeddarTech's product has a horizontal FoV of 100°. Also, the HDL has an accuracy of ± 2 cm when the Vu8 has ± 5 cm. Concerning prices, the Vu8 costs 475\$, VLP-16 costs 4,000\$ and the VLP-128 costs 8000\$.

Chapter 3

Ray tracing simulation tool

Considering that the idealised system contains cameras, it was necessary to simulate a lens, a sensor and light rays. In order to do so, several simulators like ZEMAX [34] (there is also Code V [35], Zemax's direct competitor) or Optical Software for Layout and Optimization (OSLO) [36] were first considered. These are powerful optical design programs used for the design and analysis of imaging and illumination systems such as camera lenses, based on ray tracing analysis [37].

Ray tracing consists on modeling the ray's propagation through an optical system. ZEMAX can simulate the effect of several types of lenses, mirrors and plot spot and ray-tracing diagrams. OSLO is also a lens design program that allows advanced ray tracing, analysis and optimization methods for this type of applications. The problem was that OSLO was only available as a student's version that was very limited for the necessary purposes and ZEMAX had a very high cost. For these reasons, and also for pedagogic purposes, a decision was made of creating a simulation tool from scratch using, in this case, MATLAB, which will be explained in the following sections.

3.1 Theoretical background

In order to be able to plot the ray-tracing diagrams, there are mathematical expressions that relate variables like entrance and output angles, input and output ray positions and focal length, always considering an ideal lens, that is, a transfer matrix that represents systems of this type.

In optics, a thin (ideal) lens is a lens with a thickness (distance along the optical axis between the two surfaces of the lens) that is negligible compared to the radius of curvature of the lens surfaces. The thin lens approximation neglects optical effects (like aberrations) caused by the thickness of the lenses and simplifies ray tracing calculations [38]. On the other hand, simulating a real lens would have implied taking into account distortion and aberrations, which are not relevant the scope of this thesis. Hereupon, it was assumed, for simplicity, that all simulations would considered ideal lenses.

On figure 3.1 is represented an ideal lens and the considered referential. The variable y_{plot} represents the yy position of the obstacle in relation to the optical axis of the lens and the spot's position on the x axis will be defined as z . The angle of incidence, θ_{in} is calculated considering a triangle whose sides refer to the spot's distance, z and the yy position, y_{plot} .

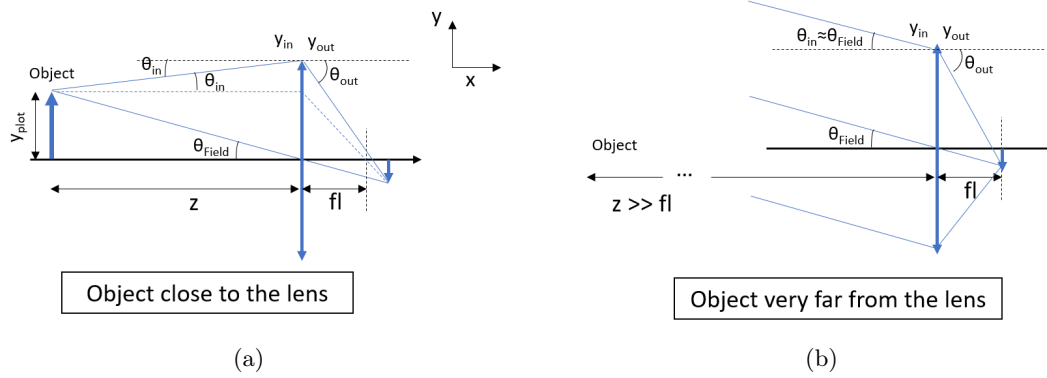


Figure 3.1: Scheme illustrating used variables to perform ray tracing for (a) an object close to the lens (point source) and (b) an object far from the lens (collimated beam).

To determine the output angle, θ_{out} , it is necessary to know the focal distance, 16 mm¹, the entrance position, y_{in} , and the angle of incidence, θ_{in} ,

$$\theta_{in} = \tan^{-1} \left(\frac{y_{in} - y_{plot}}{z} \right). \quad (3.1)$$

Hereupon, since an optical system can be described by a product of matrices, there is a transfer matrix [37] that allows to relate a lens focal length, the ray entrance position and incident angle with the output position and angle. In this case, since it is being considered a thin lens, that matrix is the one defined on (3.2) as M , being the output values given by (3.2),

$$S = M \cdot r = \begin{bmatrix} 1 & 0 \\ -\frac{1}{fl} & 1 \end{bmatrix} \cdot \begin{bmatrix} y_{in} \\ \theta_{in} \end{bmatrix} = \begin{bmatrix} y_{out} \\ \theta_{out} \end{bmatrix}. \quad (3.2)$$

This is the equation that relates the input and output angles and positions [37] considering the small angles approximation, allowing to perform all the necessary ray-tracing for every needed case.

3.2 Overview of the simulation tool

In order to simulate a lens system, a simulation tool was developed in MATLAB with the purpose of allowing the analysis of such system comprising several paraxial lens.

The simulation tool visual layout is divided in four parts: the setup, the surface data, the lens system and the spot diagram, as presented on figure 3.2. Before starting any simulation, some parameters must be defined:

1. Number of rays comprising the spot. The minimum number is two so that a focus point can exist;
2. Defines the aperture of the system;
3. The field is the angle that the central ray of the beam makes with the optical axis;

¹This value comes from the objective chosen and purchased for the experiments that has a 16 mm focal length lens.

4. Defines the spot's distance relative to the lens, in millimeters;
5. Defines the position where the sensor will be relative to the last lens;
6. Allows to choose whether the rays are collimated or not: if the rays are chosen not to be collimated, the object's distance must be defined. Collimated beams are equivalent to a point source at an infinite distance from the receiver;
7. The aperture stop position defines the distance at which the first lens will be relatively to the camera's aperture;
8. Allows the user to choose between a horizontally aligned (e.g., a wall) or circular (e.g., a rounded object) point's displacement as well as the number of points to be displayed;
9. Plots the lenses and the ray-tracing from the point source until the sensor. Every simulation gets plotted on top of the previous one so that the user can easily compare different cases;
10. Allows the user to choose the focal length and the thickness of each surface. The thickness defines the distance from one surface to another. Also, by default, the user must indicate with a 0, in the thickness position, the last surface to be represented. At the moment, the system allows a 3-lens setup, which is enough for this thesis purposes, but, if needed, more can be easily added;
11. Plots the spot's diagram at the sensor. The marginal rays (plotted at 9) hit the sensor defining the upper and lower limits of the spot. Having those two points as reference, a circle is plotted, defining the spot itself;
12. After defining all parameters, the user must press this button to run the simulation;
13. Clears all inserted values and the plotted results.

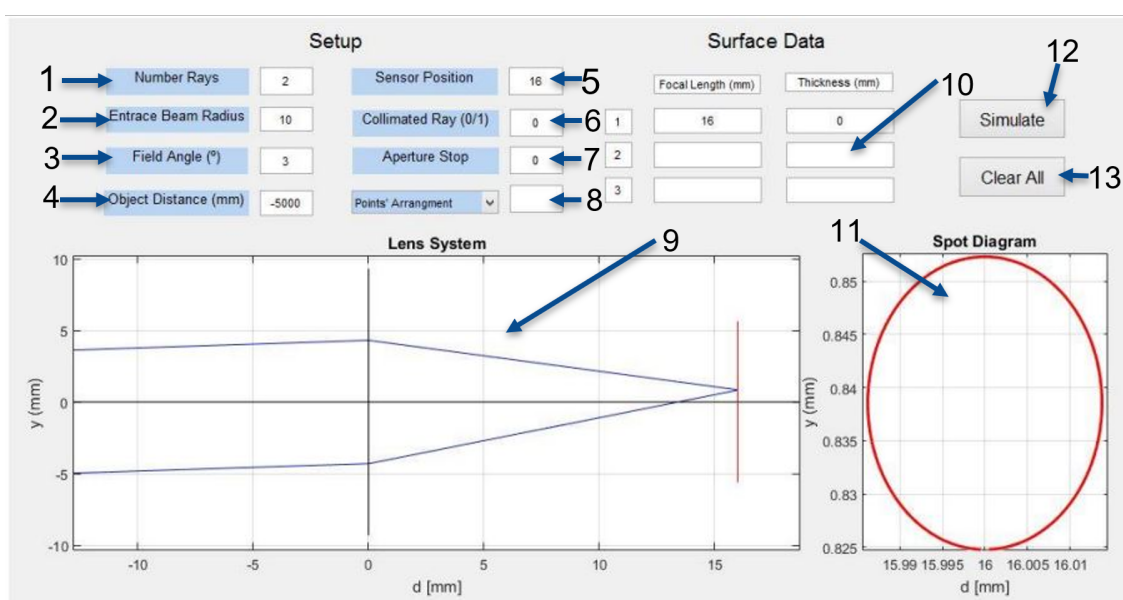


Figure 3.2: Layout of the simulation tool.

3.2.1 Validation

In order to validate the simulator, some examples were tested and compared with ones obtained with ZEMAX, kindly produced by Dr. André Albuquerque.

The first example, on figure 3.3, consists of 2 collimated beams: the first one with a field of 0° and the other with a field of 15° , 7 rays per beam, an aperture stop diameter of 30 mm and a focal length of +50 mm, that is, a convergent lens, and aperture stop position of 20 mm. Since the beams are collimated, it is expected the rays to focus 50 mm after the lens and. For the field of 0° , the beam must focus exactly at 50 mm from the lens and over the optical axis.

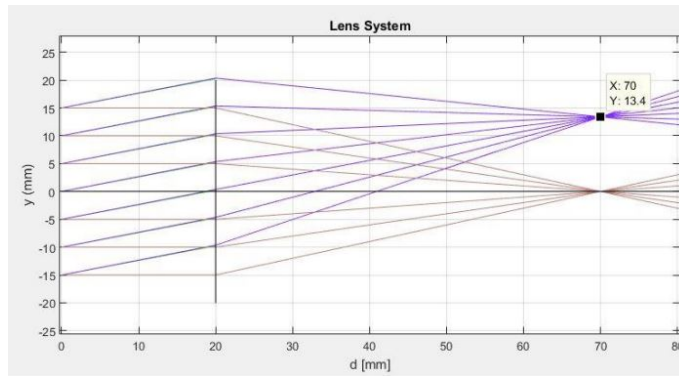


Figure 3.3: Validation example comprising two collimated beams and its focal point at $d = 70$ mm for an aperture spot of 20 mm and focal length of 50 mm.

After successfully validating the first example, the second one, on figure 3.4, consists of 2 collimated beams with fields of 0° and 15° , 7 rays per beam and an aperture stop diameter of 40 mm. In this case, there are 2 lenses: the first one with a focal length of -50 mm, that is, a divergent lens, and the second one of +40 mm, a convergent lens. The first one is at a distance of 20 mm from the camera's aperture and the second lies 40 mm from the first lens.

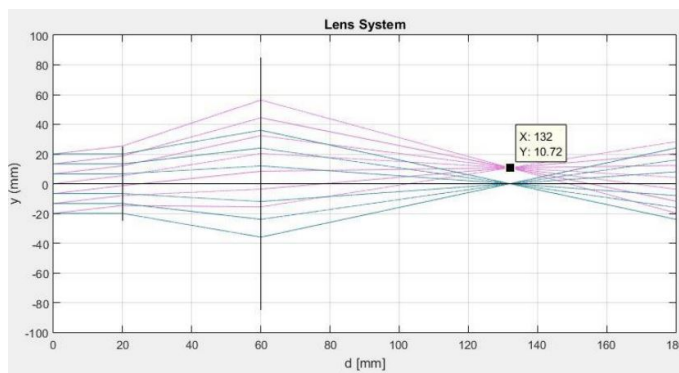


Figure 3.4: Validation example comprising two collimated beams, a divergent and converging lens with a focal point at $d = 132$ mm.

Both examples were successfully validated, proving the correct functioning of the simulation tool.

3.3 Capabilities

In order to simulate a point source, it was assumed that each spot is simulated considering two marginal rays. Both rays hit the limits of the first lens, thus hitting the sensor at the most extreme positions.

The following example, on figure 3.5, considering a point source with a field of 0° at a distance $z = 3$ m, represents how the spot's image at the sensor is obtained: both marginal rays hit the lens, being the two represented points at 3.5b the intersection of the referred marginal rays with the sensor. Having these two points as reference, a circumference is plotted with the purpose of representing the entire spot projected on the sensor. Such a representation is valid as long as the objective lens is symmetric both vertically and horizontally, which is typically the case.

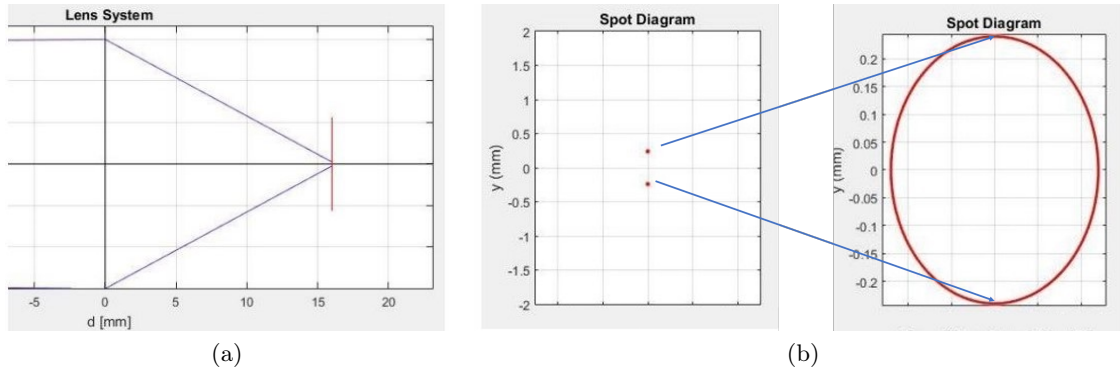


Figure 3.5: (a) Ray tracing, (b) representation of the two points coming from the intersection of the marginal rays with the sensor and spot's diagram at the sensor.

The simulation tool is able to simulate several point sources at any distances and fields, also allowing to define several parameters as previously specified, to visualize all the ray-tracing diagrams as well as the spot's diagram at the sensor. Even considering that every lens' system can be approximated as a single equivalent lens, the tool allows to consider systems with several lens in cascade, for instance, to simulate a relay lens.

The simulation tool does not allow a 3D representation of the point sources, that is, it allows to represent them at different fields and distances, however, given the circular symmetry of the system, the results for the horizontal plane are the same as shown here for the vertical plane. Also, the simulator does not take directly into account the intensity of the spot. In other words, it has been considered, for the sake of simplicity, that the photons are equally distributed throughout the spot, which, in reality, is not verified.

Chapter 4

Triangulation LIDAR

In this thesis, a geometrical approach was investigated for estimating a spot's distance considering a certain referential by using triangulation [39]. This method consists on having two “observers” with a known distance between themselves and a third point that represents the spot whose distance must be estimated. These three points form a triangle, as illustrated on figure 4.1.

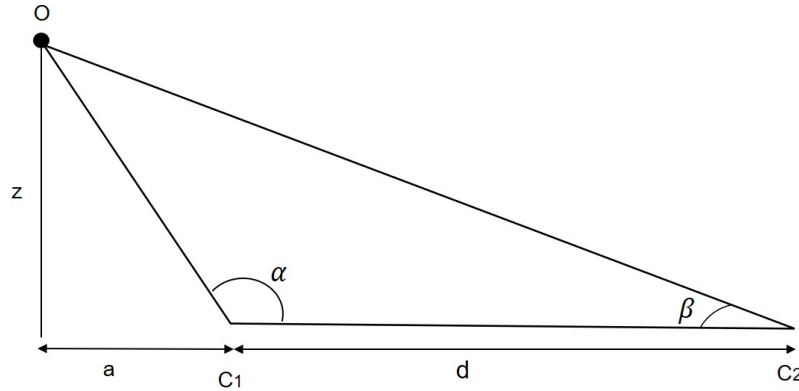


Figure 4.1: Basic triangulation scheme: z can be estimated if α , β and d are known.

Considering the referred triangle, if the two angles α and β are known, then z can be estimated by simple trigonometry. The spot (O) is the target whose distance, z , is to be estimated; C_1 and C_2 represent both cameras on the left and right, respectively; α is the angle made by $\angle C_2C_1O$ and β the angle made by $\angle C_1C_2O$; a is an auxiliary variable and d is the distance between both cameras called baseline.

By looking at figure 4.1 one can derive, by means of basic geometry, that:

$$\tan(\pi - \alpha) = \frac{z}{a}, \quad (4.1)$$

thus,

$$a = \frac{z}{-\tan(\alpha)}. \quad (4.2)$$

The same reasoning is applied considering β and $a + d$, in

$$\tan(\beta) = \frac{z}{a + d}. \quad (4.3)$$

$$z = a \cdot \tan(\alpha) + d \cdot \tan(\beta), \quad (4.4)$$

thus,

$$z = d \cdot \frac{\tan(\alpha) \cdot \tan(\beta)}{\tan(\alpha) + \tan(\beta)}. \quad (4.5)$$

Knowing that there is an error associated to the considered variables, that is, d , α and β , it is very important to study the effect that a variation on those variables has on the distance estimation error. For other words, analyse the variation on the estimated distance, z , when varying d , α or β .

Considering (4.5), there were obtained the partial derivatives of z with respect to α , β and d :

$$\frac{\partial z}{\partial \alpha} = d \cdot \frac{\sec^2(\alpha) \cdot \tan^2(\beta)}{(\tan(\alpha) + \tan(\beta))^2}, \quad (4.6)$$

$$\frac{\partial z}{\partial \beta} = d \cdot \frac{\sec^2(\beta) \cdot \tan^2(\alpha)}{(\tan(\alpha) + \tan(\beta))^2}, \quad (4.7)$$

$$\frac{\partial z}{\partial d} = \frac{\tan(\alpha) \cdot \tan(\beta)}{\tan(\alpha) + \tan(\beta)}. \quad (4.8)$$

The total error is given by:

$$\partial z = d \cdot \frac{\sec^2(\alpha) \cdot \tan^2(\beta)}{(\tan(\alpha) + \tan(\beta))^2} \cdot \partial \alpha + d \cdot \frac{\sec^2(\beta) \cdot \tan^2(\alpha)}{(\tan(\alpha) + \tan(\beta))^2} \cdot \partial \beta + \frac{\tan(\alpha) \cdot \tan(\beta)}{\tan(\alpha) + \tan(\beta)} \cdot \partial d. \quad (4.9)$$

By the previous equations, the following results for the distance estimation error were obtained on figure 4.2.

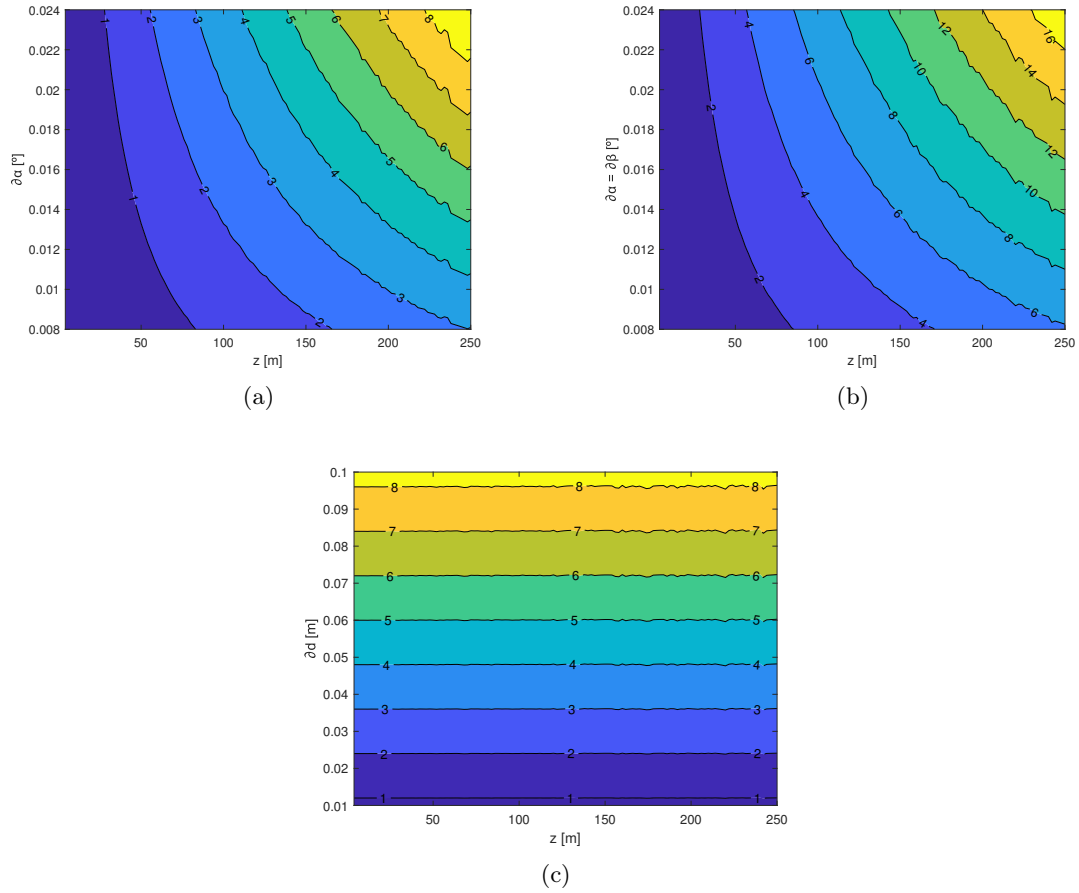


Figure 4.2: Distance estimation error, $\partial z/z$, in percentage (a) considering a variation on α and $\partial\beta = 0$, (b) considering an equal variation on both α and β and (c) considering a variation on d , according to the spot's distance, z , considering $d = 1.2$ m and a field of $= 5^\circ$.

Considering figures 4.2a and 4.2b, the values of the angle variation go up to 0.024° because it is the FoV correspondent to a pixel. This means that the error for $\partial\alpha = 0.024^\circ$ represents the theoretical estimation error resulting from the error of one pixel considering that $\partial\beta = 0$.

On figure 4.2b is represented the obtained error for several $\partial\alpha$, but, on this case, it was considered that there was an error estimation on both sensors, that is, $\partial\alpha = \partial\beta$. According to (4.9), since the error due to each component is summed, it was expected that the error obtained for this case would be twice as much as for the previous case, as verified.

Lastly, the variation of z is directly proportional to the baseline distance variation, ∂d , as observed on figure 4.2c, that is why the obtained results have the observed behavior. Since $\partial z/\partial d$ is constant and systematic, it is simple to correct.

4.1 Operation principle

In order to implement the triangulation concept, the setup on figure 4.3 was devised, consisting, in a first stage, of two cameras and a laser.

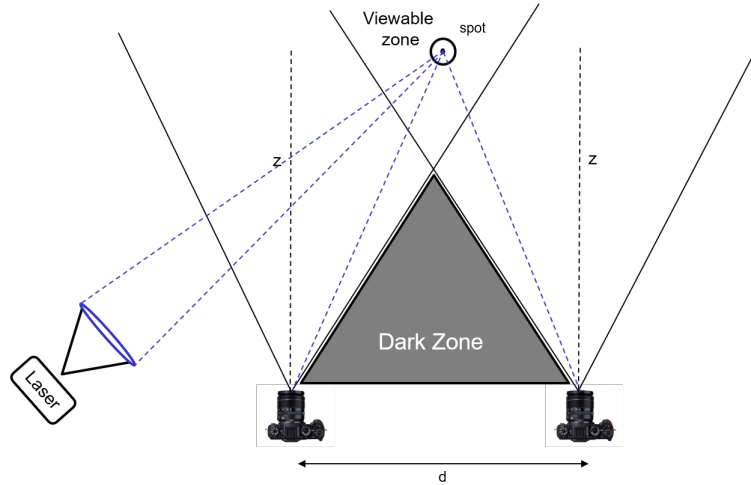


Figure 4.3: Layout considering each camera's FoV and the setup's dark zone.

Considering a spot as a point source, that is, an isotropic light source that is within the FoV of both cameras, that point will be captured by both cameras. As explained in the previous chapter, the spot in a certain obstacle is simulated as a point source and its image on the sensor is obtained based on the point source's marginal rays.

In order to simulate such scenario, some parameters must be taken in account throughout all experiments. The FoV [40], that is, the angular extent of a certain scene that can be imaged by a camera, considering the used sensor and objective, is 45° . The distance between cameras, d , was assumed to be 1.2 m throughout all simulations. This value comes from comparing the width of several cars of different segments, from small utilitarians to big Sport Utility Vehicle (SUV). The chosen reference was a Smart Fortwo, which is the smallest car on the market, being its width approximately 1.2 m. In case a final prototype has to be placed on a car, if it fits the smallest car, then it will also fit in a bigger car.

As highlighted on figure 4.3, there is a zone that will not be on any camera's FoV, called the "dark zone". The black filled lines represent the FoV limits for each camera. The zone that is seen by both camera's FoV is called the "viewable zone", being the area where triangulation can be applied. Considering that both cameras, in practice, will be precisely pointing forward relatively to the car, then, the height of the triangle corresponding to the dark zone is 1.45 m, meaning that, with this setup, the car can only estimate distances greater than 1.45 m.

4.1.1 Upgraded simulation tool

In order to simulate a system with two cameras, the simulation tool was upgraded to display two ray tracing diagrams for the beams captured by each camera, as illustrated on figure 4.4. Spot diagrams were also included in order to visualize the spot on the sensor as well as a graphical representation of its geometrical center on both right and left sensors. The simulation parameters are defined by the user the same way as illustrated on figure 3.2, being the distance between cameras pre-defined, as well as the lenses characteristics are the same for both.

Figure 4.4 represents a point source 250 m away from the receiver, being on the common FoV of the cameras. The vertical line plotted at 0 mm represents the lens and 16 mm after is the sensor that, due to the scale with which the scheme is represented, is not visible. Both marginal rays hit the lens, focusing on the sensor, originating the spot diagram. It should be noted that only one lens is used throughout all simulations and that the spot's field is defined relatively to the camera on the right (upper diagram). In this case, the field is 0° , that is why, on the upper spot diagram, the geometrical center (represented by the point in blue) is exactly at 0 mm. The obtained centroid for the left sensor is -0.096 mm, that is, a different value from the one obtained on the right sensor, corroborating the concept of triangulation.

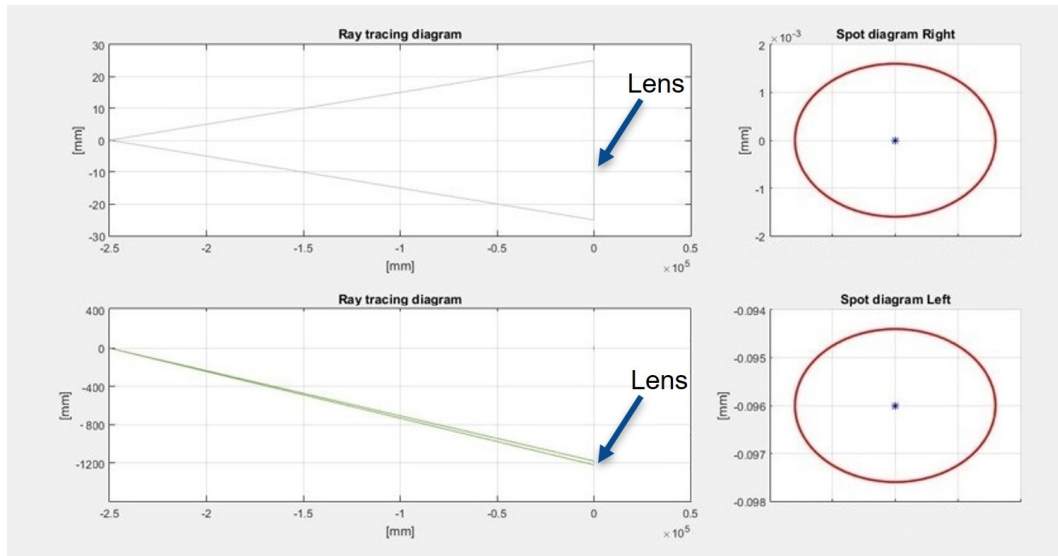


Figure 4.4: New simulation tool layout for simulating a system with two cameras.

After having the necessary basis to implement a simulation procedure, the following scheme, on figure 4.5, illustrates the followed procedures to estimate the distance, z .

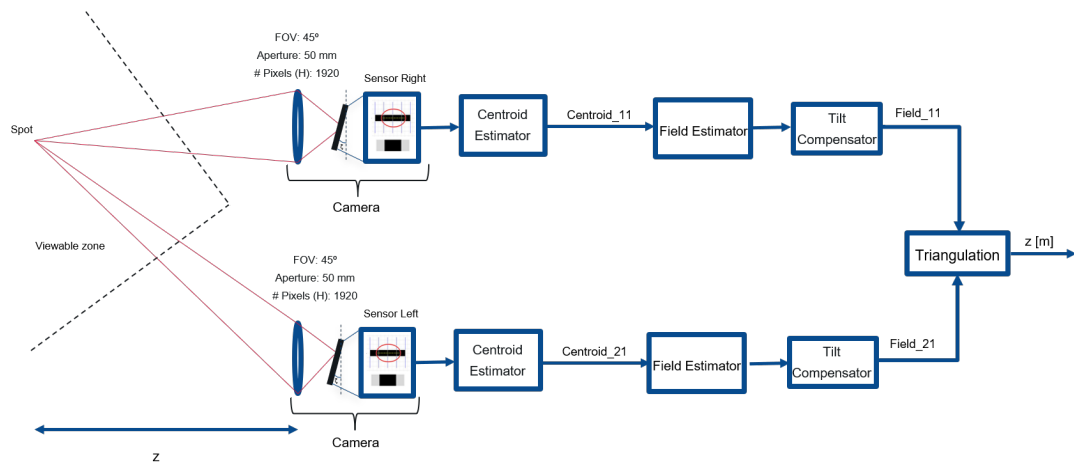


Figure 4.5: General scheme of the setup from the spot hitting an object until estimating the distance, z .

A spot on the viewable zone hits the lens, focusing on the sensor. Due to a possible physical cameras' displacement, that is, the cameras might not be perfectly aligned, a possible camera's tilt, $\Theta_{1,2}$, must be taken in account. Still, the centroid is estimated, being that value used to search on a look-up table the correspondent field, that is, the "Field Estimator". Then, a block was added to take into account the tilt's effect, that is, the "Tilt Compensator", being the tilt's effect traduced by: $\alpha_{out} = \alpha_{in} \pm \Theta_1$ and $\beta_{out} = \beta_{in} \pm \Theta_2$. After correcting a possible tilt, the distance is effectively estimated using (4.5).

4.2 Centroid estimation

Having a look-up table that relates a point's position on the sensor to the correspondent field, the goal is to estimate that point considering the spot's image on the sensor, knowing that a spot will spread over multiple pixels, depending on its distance to the receiver.

At this point, it is introduced the concept of centroid. A centroid is the geometrical center of the spot, represented as a coordinate that will be considered on the look-up table in order to find the correspondent field.

To estimate the spot's centroid, two techniques are tested and explained in the following sections: intensity-independent and intensity-dependent.

4.3 Intensity-independent centroid estimation

When obtained the spot's diagram at the sensor, it is possible to discern which pixels have been hit by the spot. On figure 4.6 are represented the spot's image at each sensor, being the red circle the spot limits.

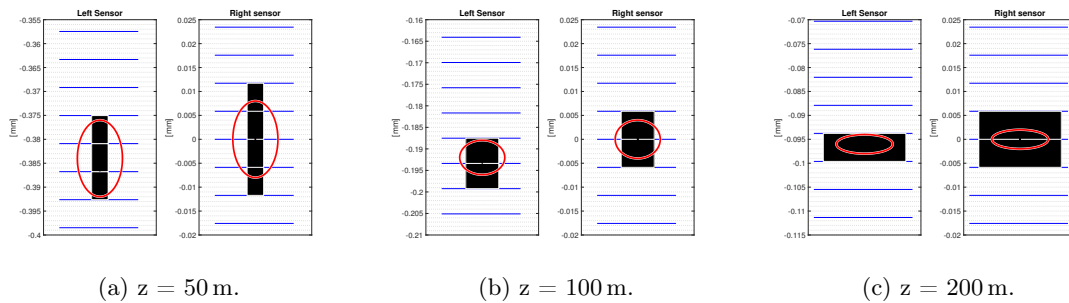


Figure 4.6: Spot diagram for intensity-independent centroid estimation and distances z , for a field of 0° .

Here, all the pixels hit by the spot are considered for the centroid's estimation, being filled in black. It should be noted that the pixel size is of $5.86 \times 5.86 \mu\text{m}^2$ and that the blue horizontal lines represent the pixel's limits at the sensor in order to have a more visual perception of the simulated scenario, being the horizontal axis merely illustrative.

In figure 4.6a is represented the spot's diagram for both sensors considering a spot 50 m away from the receiver. On the left sensor, the spot occupies 3 pixels, being the position of the centroid on the upper limit on the pixel in the middle, that is, -0.3809 mm. For the right sensor, the centroid is estimated as the mean of the 4 pixels' position, that is, the centroid will be at 0 mm. As said before, since the field is considered relatively to the

camera on the right, in this specific case, the spot is in front of the right's camera, that is why the centroid is precisely at 0 mm.

On figure 4.6b is represented the spot's diagram for a spot distance of 100 m. The spot's size at the sensor is much smaller than on the previous case, occupying, now, 2 pixels. In order to estimate the centroid, since the spot overlaps 2 pixels, then it will be the mean of those pixels position, that is, -0.1934 mm and 0 mm for the left and right sensors, respectively.

The last case, on figure 4.6c, represents the spot's diagram for an object 200 m away from the receiver. Now, the spot occupies less than a pixel's size, still, the entire pixel is considered for the estimation. On the right, the spot has obviously the same size, but, by chance, it hit 2 pixels, so those 2 will be considered for the centroid's estimation, being the obtained centroids -0.0938 mm and 0 mm.

Considering all the results above, the maximum error that can occur is of 1 pixel. In case a single ray hits a pixel, as referred, that pixel will be considered for the centroid's estimation which leads to a certain error. On the other hand, in case it happens that, for example, one single ray hits a pixel "above" and "below" the spot's limits, it won't affect the centroid's estimation. Considering the referred limitation of this method, an intensity-dependent scale method was tried.

4.4 Intensity-dependent centroid estimation

Although the intensity-independent estimation is straightforward, it is not very accurate knowing that the intensity value of each pixel is neglected.

So far, it was assumed that, on the intensity-independent estimation part, the spot's rays were equally distributed throughout the spot. Using an intensity-independent scale does not take advantage of the intensity information on each pixel, as well as the fact that, with this method, one ray hitting a pixel is enough for that pixel to be considered for the centroid estimation, meaning that a pixel with a high intensity value has the same weight as a pixel with a very low one.

For that reason, instead of using an intensity-independent scale, it was tested to use an intensity-dependent one in order to try to obtain a more precise estimation. To do so, it was considered a group of rays spread according to a Gaussian distribution throughout the spot, that is, denser in the middle and less dense the more distant from the center, due to the fact that, in practice, the light emitted by lasers follows the same distribution.

At the sensor, each pixel will be hit by a different number of rays, corresponding to different intensities on each pixel, leading to the "gray scale matrix" represented on the following picture 4.7, on the right of each spot. The gray scale matrix contains, at each position, the number of rays that hit a specific pixel, which was then converted to a gray-scale representation, being each position associated to a certain pixel.

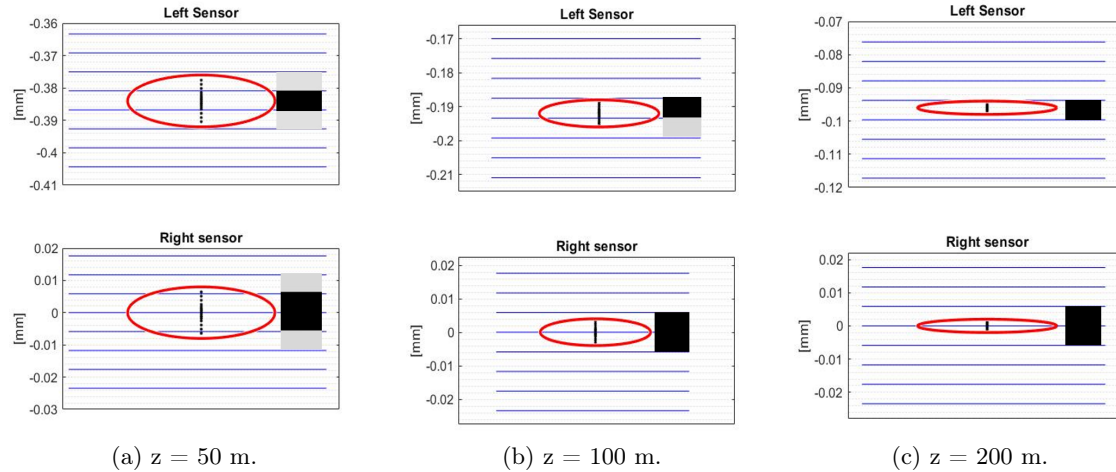


Figure 4.7: Spot diagram at the sensor for intensity-dependent scale centroid estimation and several spot distances.

On the sensor, it can be estimated how many rays hit each pixel, leading to a matrix containing, for each pixel, the number of hit rays. The pixel hit by the largest number of rays is the darkest: the smaller the number of rays on each pixel, the lighter the shade of gray. Having this information, in order to properly estimate the centroid, the system searches the most intense pixel, that is, the pixel that was hit by the largest number of rays, being that pixel's superior limit the position corresponding to the centroid. In case there are found two pixels with the same intensity, then the centroid is the given by the average of those two pixel's position.

For the first example on figure 4.7a, for $z = 50$ m, the centroid for the left sensor is -0.3809 mm and for the right sensor is always 0 mm due to the chosen field of 0° . On the second example, for $z = 100$ m, the centroid is -0.1875 mm and for $z = 200$ m, on figure 4.7c, the obtained centroid was -0.0938 mm.

Considering all the results above, for the intensity-dependent approach, the unwanted scenario that happened for the intensity-independent cases of considering, for effects of counting, a pixel that was hit, for instance, by just one ray, no longer happen. The pixel that matters is the most intense one, which defines, itself, where the centroid will be.

Another possible approach would be making a weighted average of the pixel's intensity, but, considering that from $z = 50$ m on, a spot only occupies between 3 and 4 pixels and, from $z = 100$ m on, between 1 and 2 pixels, it means that making a weighted average would have no major advantage due to the very small number of hit pixels. For that reason, it was chosen to consider the most intense pixel as reference for the centroid's estimation.

4.5 Field estimation

At this stage, the point is to find exactly the spot's position at the sensor and create a correspondence between its position and field, i.e., α and β . Considering a FoV of 45° and a sensor with 1920 horizontal pixels, dividing the FoV for the number of pixels, then it results in 0.024° as the FoV correspondent to each pixel. The angular spacing used to build the look-up table was 0.002° in order to guarantee precise results. In order to do so, it was simulated a collimated beam for the entire "viewable zone", (that is, for several fields) since a collimated beam produces an infinitesimal spot on the sensor. Then, for each field,

the point's position on the sensor was saved, building the look-up table. It was chosen to create a look-up table due to its practicality and simplicity comparing, for instance, to a mathematical way for estimating the correspondence between the spot's position on the sensor the the correspondent field.

4.6 Results

4.6.1 Ideal centroid estimation

In order to test the validity of the look-up table, it was simulated a point source at several distances and fields. The spot's distance was estimated and compared with the real value, z , providing the estimation error. The obtained results are shown in figure 4.8.

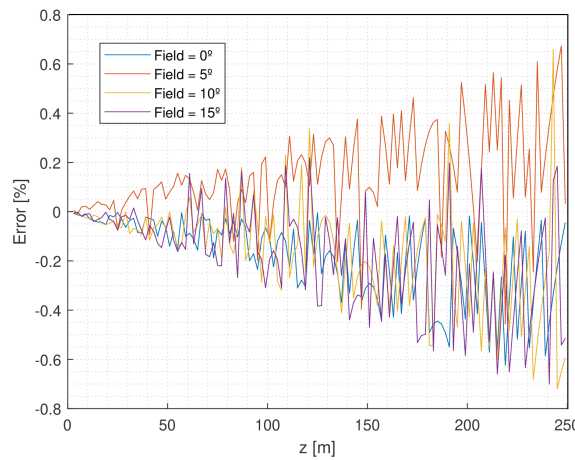


Figure 4.8: Distance estimation error as a function of the spot's distance for several fields considering an ideal centroid estimation.

In theory, considering an ideal centroid, the distance estimation error should be zero. The obtained values for α and β (and consequently, the distance, z) are obtained, as known, from the look-up table that, by itself, is discretized with a step of 0.002° . Considering that the FoV correspondent to a pixel is approximately 0.024° , the chosen step causes there to be about 12 levels of precision. In an ideal scenario, this step would be infinitely small, leading, precisely, to an error of 0% throughout the entire range of distances.

Figure 4.8 reveals that the error increases with the distance in a monotonic way, that is, increases almost linearly with the distance. It was expected that the error would not vary with the field if there was a sensor with an infinite number of pixels, but, in fact, it does vary, again, due to the chosen step for the look-up table. The cases where the error is 0% are those in which, by coincidence, the value on the look-up table was exactly the value of the ideal centroid.

4.6.2 Intensity-independent centroid estimation

On figure 4.9a is represented the distance estimation error according the spot's distance for four different fields.

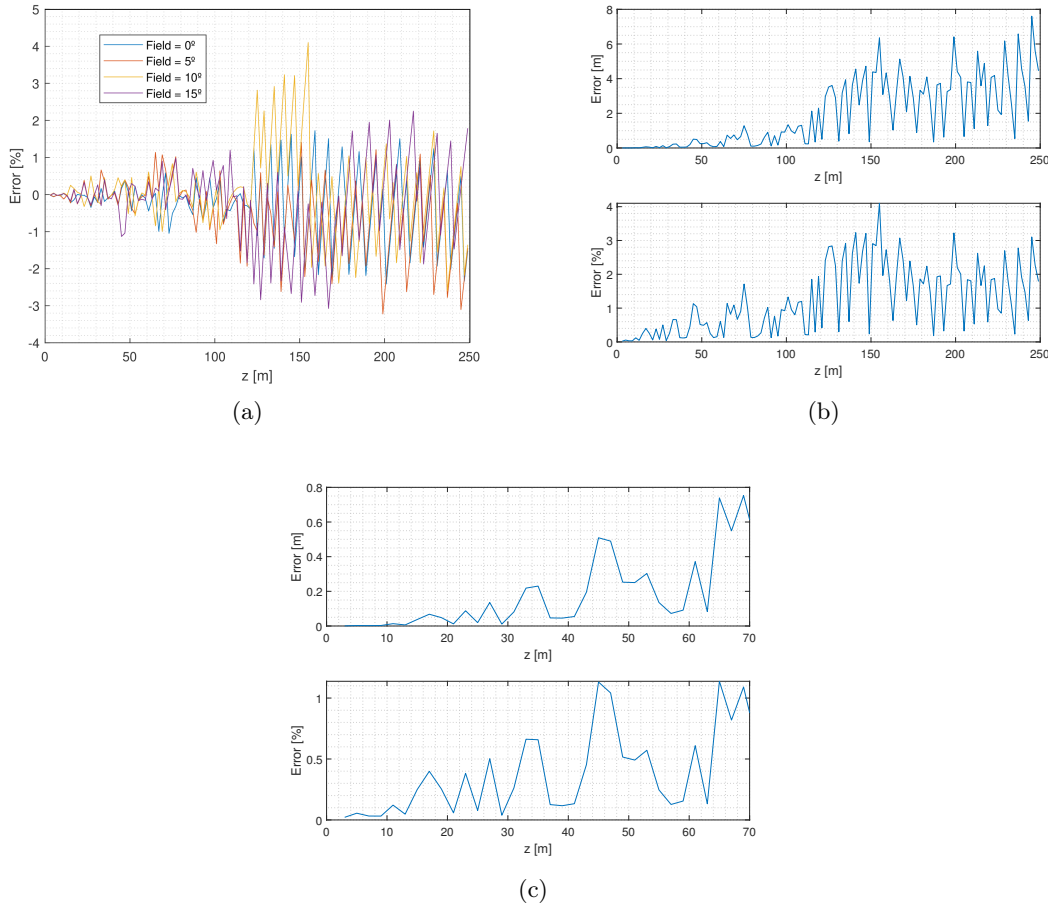


Figure 4.9: (a) Distance estimation error, (b) maximum distance estimation error and (c) maximum distance estimation error for the short range according to the spot's distance for intensity-independent scale and different fields.

Considering the intensity-independent scale, it is expected that the obtained results on 4.9a follow approximately the same tendency as the ones obtained on figure 4.8, but, due to the centroid's position discretization, with a larger error. Those results were confirmed. The error follows, in general, the same behavior, that is, increases with the distance and varies for different fields, but does not increase monotonically due to the centroid's position discretization, having, instead, a very sharp saw-tooth behavior. For the same distance and a different field, the error is different, but the general impact of the field on the error is the same. This difference is due to the fact that, even knowing that for the same distance the spot has the same size, varying the field varies the number of pixels reached by the spot, hence the variation of the error as a function of the field for the same distance.

The error's saw-tooth behavior has to do with the cases when a spot hits several pixels and one of the external pixels is hit by a very small number of rays: since it is considered an intensity-independent scale, that pixel will be taken into account, representing the worst case scenario.

In order to perform a better analysis of the error's behavior, it was studied with more detail the error for the short range, that is, until 70 m, as seen on figure 4.9c. For all the considered fields, it is represented the largest error for each distance, allowing to trace a worst case scenario for this centroid estimation method. Until 12 m, the error is lower than 0.1%, which corresponds, maximum, to an error of 1 cm. After this distance, the error starts to vary with a saw-tooth behavior, as clearly seen on figure 4.9, reaching, at 70 m, an error of almost 0.8 m.

The increase of the error is due to the decreasing spot's size. As seen on figure 4.6, the spot's size decreases with the distance, leading to a less precise estimation of the centroid and, consequently, of both α and β , because there is not enough available resolution. For that reason, the impact that an error of one pixel has (which is the worst case scenario) gets worse as the distance increases, as shown on figure 4.10.

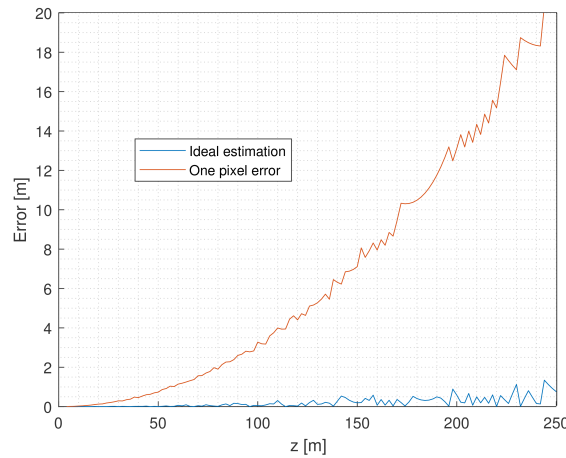


Figure 4.10: Distance estimation error due to an one pixel error on the sensor compared with a no error scenario, both for an ideal centroid estimation and a field of 5° .

Here is represented the error that a miscalculation of one pixel, in one of the sensors, has on the estimated distance compared with an ideal case, that is, these results were obtained considering an ideal centroid estimation, being corroborated by the obtained results on figure 4.2a, being, for instance, an error of approximately 20 m correspondent to the obtained 8%, for the same z , by the theoretical error analysis.

4.6.3 Intensity-dependent centroid estimation

Following the same procedure as on the previous section, it was simulated the distance estimation error according the spot's distance, z , for several fields, on figure 4.11, considering an intensity-dependent centroid estimation.

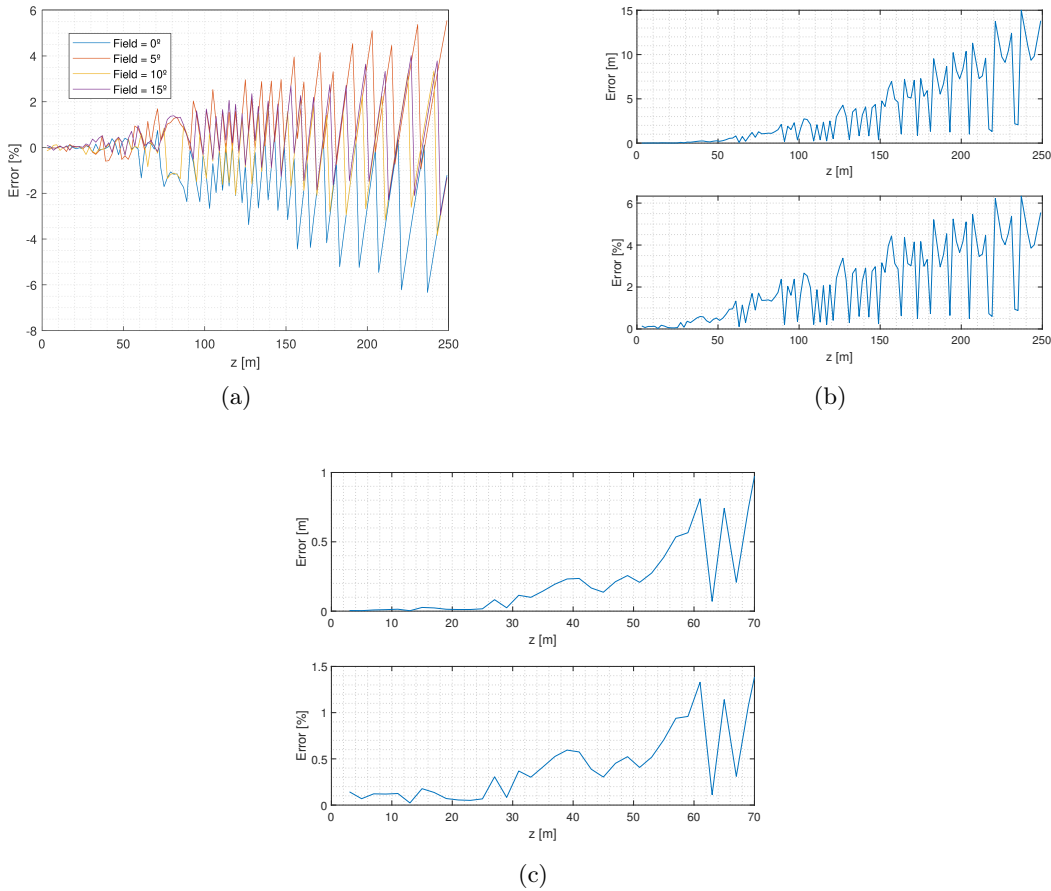


Figure 4.11: (a) Distance estimation error, (b) maximum distance estimation error and (c) maximum distance estimation error for the short range according to the spot's distance considering intensity-dependent scale for several fields.

As seen on figure 4.11a, the error follows a linear behavior with the distance. Analogously to the intensity-independent scale, the saw-tooth behavior is still verified for the same reasons. By applying this estimation method, the results are expected to be at least as good as the intensity-independent one since this approach takes advantage of the intensity value of the pixels and not simply if the pixel was hit or not by a ray.

For a better understanding of the error's behavior for the short range of distances, it was made a zoomed-in plot, as seen on figure 4.11c. The analysis of this figure shows, as done for the intensity-independent estimation, a zoomed view of the error in meters and percentage up to a distance of 70 m. Until around 40 m, the maximum error hits a maximum of 0.6%, which corresponds to an error of 24 cm. From 60 m to 70 m the error increases until 1.4%, which corresponds, in a worst case scenario, to an error of almost 1 m.

Also, it is important to point out what happens for the long range distances, that is, from 150 m on: the error increases, as happened for the intensity-independent scale, in a monotonic way, reaching almost 6%. The high error value for this range of distances is due to the number of pixels occupied by the spot, which varies between 1 and 2. Considering the very low number of pixels, different methods do not have a great impact on the error due to the resolution limit of the sensor.

4.6.4 Comparison between methods

In order to perform a better comparison of both methods, the gain from one method to the other was estimated. The gain is obtained dividing the maximum error of one method by the other: in this case, intensity-independent / intensity-dependent, as plotted on figure 4.12. In order to make a better analysis of the obtained results, this graphic was plotted using a logarithmic scale on the yy axis.

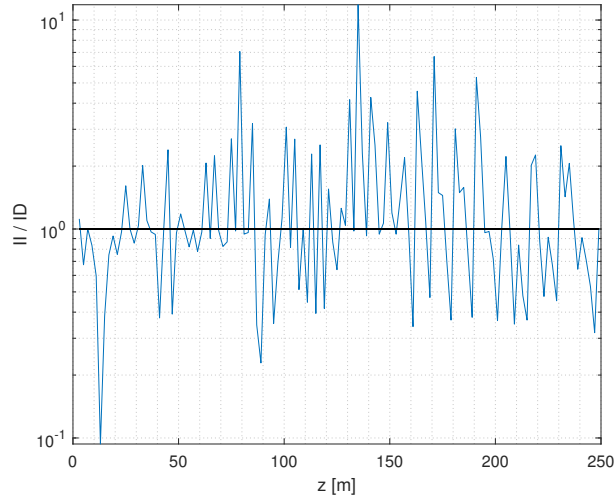


Figure 4.12: Obtained gain from using an intensity-independent (II) scale compared with an intensity-dependent (ID) one for the centroid's estimation.

As observed on the figure above, if the gain is higher than 1 (black line), the obtained error by intensity-independent scale is larger than the intensity-dependent one, meaning that, for all the distances where the gain is larger than one, the intensity-dependent method originates a smaller error. The points where the gain is equal to one mean that the maximum obtained error for that distance was the same for both methods. For the short range of distance, there was not a great improvement, that is, the results were similar for both methods (gain equal to 1). Still, for medium range of distances, that is, from 70 m to 150 m, the medium gain was 1.9. The general medium obtained gain was 1.44, proving that there is an improvement by using an intensity-dependent scale.

Chapter 5

Triangulation LIDAR with super-resolution

As seen in the previous chapter, the greater the distance, the smaller is the spot at the sensor, leading to a greater estimation error. The consequent error due to this behavior has a super-linear variation with distance. The main reason for the systematic error is low sensor resolution, being a desirable solution to increase resolution above the resolution of sensor. Solutions for addressing such a problem are referred to as geometrical super-resolution [41].

This chapter presents a super-resolution method that consists in creating bigger spots at the sensor by considering pairs of spots instead of only one spot at a time, that is, two spots per beam. Now, a pair of spots is considered as a “super spot”, making more pixels available for estimating the centroid.

In order to simulate this scenario, it is necessary to add another point source, which, in practice, simply consists of having another laser with the same power. Lasers were set 10 cm apart because, considering that the FoV of a single pixel is approximately 0.024° , for a maximum distance of 250 m the minimum spacing between a pair of spots such that these do not overlap at the the same pixel is of 10 cm. This means that effective resolution is at least doubled in comparison to a single spot per beam. Also, it is reasonable to assume that two spots spaced by 10 cm hit the same object. Sending a pair of spots is a better solution than sending a larger or highly-diverging spot when considering eye safety. Eye safety standards, explained in section 8.1, impose a Maximum Permissible Exposure (MPE), which is basically maximum power per unit of area. As a result, two spots with the same power can be launched provided that these be sufficiently spaced, whereas emitting a single spot with twice as much power usually ends up infringing the maximum MPE.

5.1 Operation principle

On figure 5.1 is represented the setup's scheme, comprising a series of stages with the goal of estimating an effective centroid and, consequently, the distance, z .

Comparing with the one spot per beam scenario, the difference is that, after estimating both centroids on both sensors, it is obtained their mean value, that is, the so called effective centroid, being that value the one used on the look-up table to obtain the correspondent field.

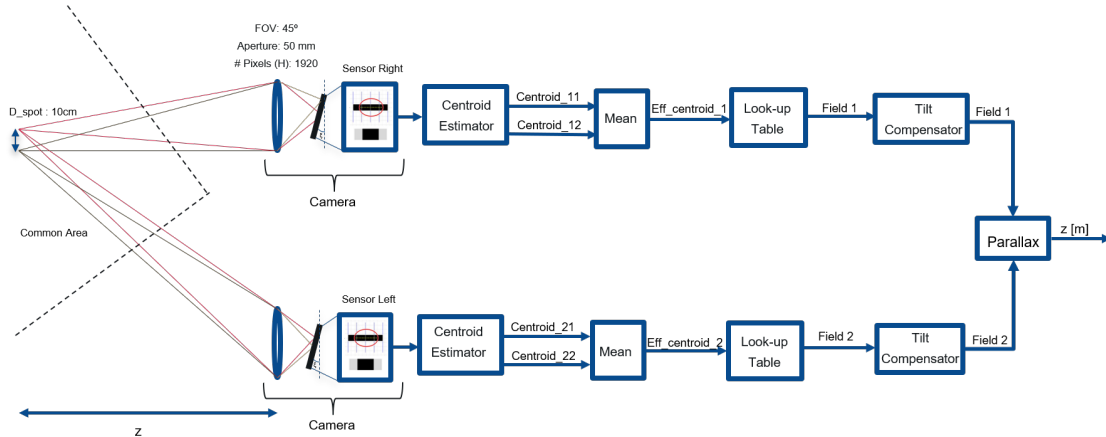


Figure 5.1: General scheme of the setup considering two spots per beam.

5.2 Centroid estimation methods

5.2.1 Intensity-independent centroid estimation

In order to estimate the effective centroid considering two spots per beam, the procedure is similar as in one spot per beam: for each spot of each pair, the centroid is estimated, in this case, using the intensity-independent scale approach. After estimating each spot's centroid, that is, centroid_11 and centroid_12 (the same for the other sensor), the mean value of both is calculated. On figure 5.2 are represented the spot's diagrams on both sensors considering two spots per beam for different distances, z .

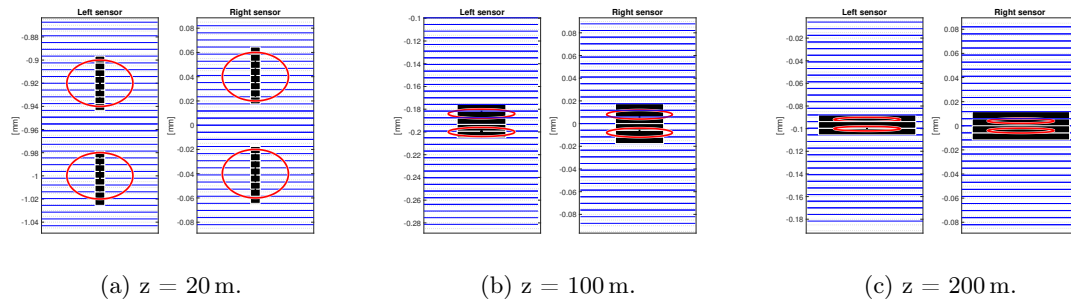


Figure 5.2: Spot diagram at the sensor considering two spots per beam and an intensity-independent centroid estimation for several spot distances, z and a field of 0° .

Comparing figures 5.2a to 5.2c, it can be observed the variation on the number of occupied pixels by the spot according to the variation of the spot's distance, z . On the

first figure, for $z = 20$ m, a spot occupies around 8 pixels and each beam, that is, the pair of spots, occupies 21 pixels (in length). This means a gain of almost 3 in the number of “available” pixels compared with the one spot per beam approach. For $z = 100$ m, on figure 5.2b, a spot occupies 2 pixels and the pair itself hits 5 pixels, that is, a gain of 2.5. Lastly, for $z = 200$ m, on figure 5.2c, the obtained gain is 3, passing from 1 hit pixel to 3.

So, in a first analysis, it is clear the achieved gain in resolution obtained by considering pairs of spots, allowing to conclude that this approach assures a minimum gain of 2 in available resolution.

5.2.2 Intensity-dependent centroid estimation

Considering an intensity-dependent centroid’s estimation, the procedure is exactly the same as in one spot per beam. After having both centroids estimated, it is made their average in order to estimate the effective centroid. Lastly, as on 5.1, the effective centroid will correspond to a certain field to be then applied on (4.5).

5.2.3 Results

Distance estimation error for intensity-independent centroid estimation

On figure 5.3 are represented the distance estimation error and the maximum error considering two spots per beam, for different fields.

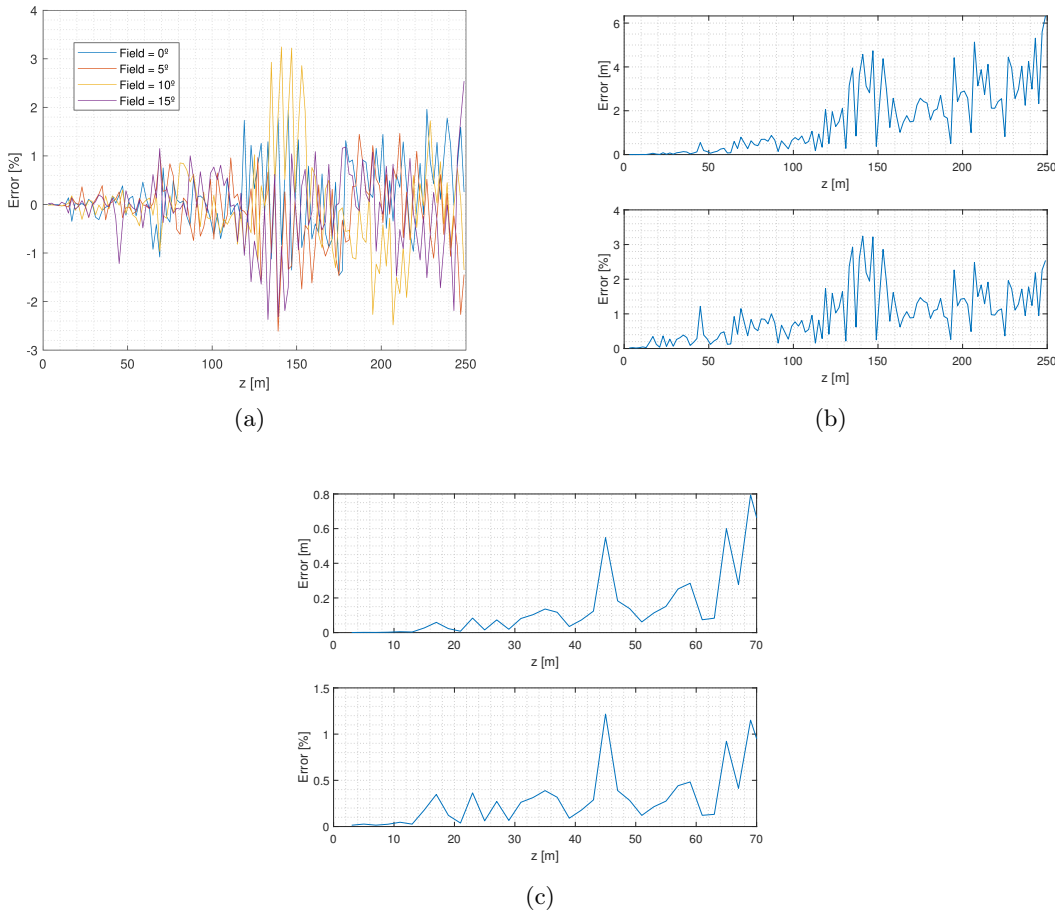


Figure 5.3: (a) Distance estimation error, (b) maximum distance estimation error and (c) maximum distance estimation error for the short range according to the spot's distance, z , for intensity-independent centroid estimation and several fields.

As in the previous simulations, the error maintains a saw-tooth behavior due to the already explained reasons. The worst case scenario is for the range of 200 to 250 m, where the error almost reaches 6 m. Still, for a better analysis of the short range distances, it was plotted a zoomed version of the maximum error according distance, on figure 5.3c: until a distance of 35 m, the maximum error is 0.4%, which corresponds to an error of 14 cm, but, for instance, at around 40 m the error gets as low as approximately 2 cm.

Distance estimation error for intensity-dependent scale estimation

On figure 5.4 are represented the distance estimation error and the maximum error considering two spots per beam and an intensity-dependent centroid estimation, for different fields.

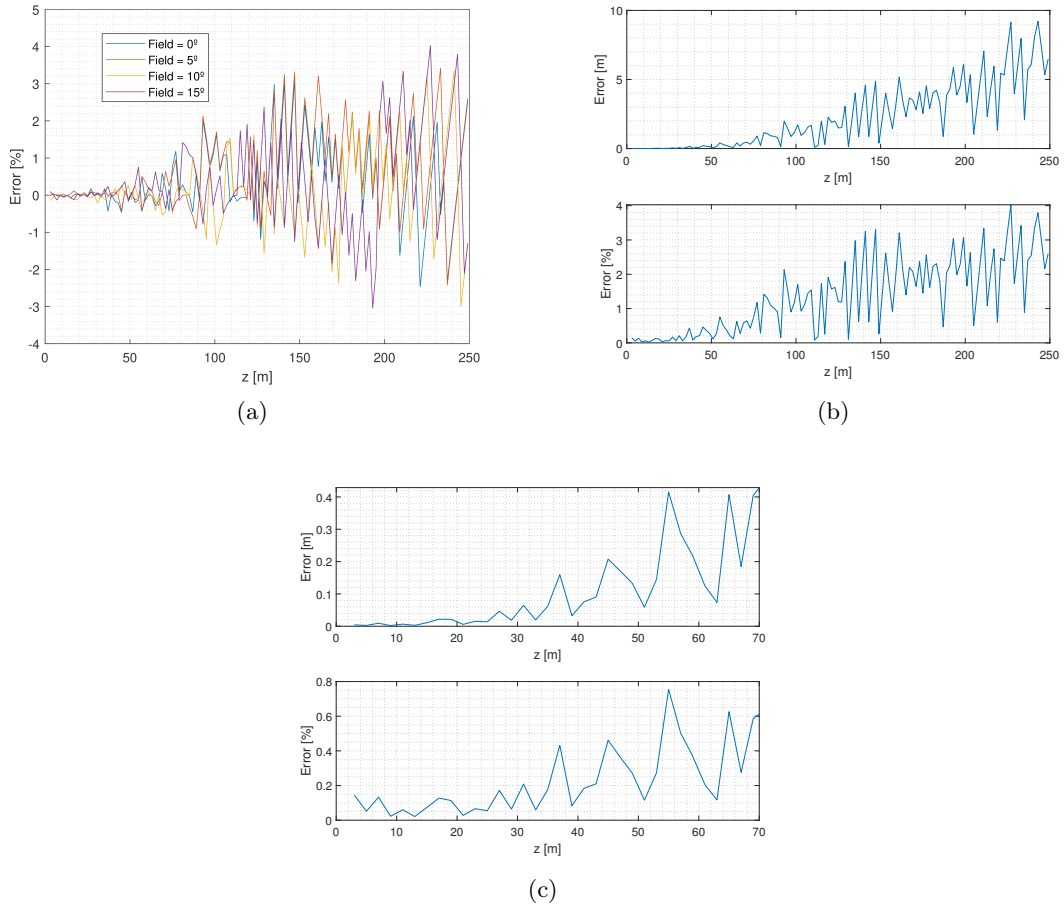


Figure 5.4: (a) Distance estimation error, (b) maximum distance estimation error and (c) maximum distance estimation error for the short range according to the spot's distance, z , for intensity-independent scale and two spots per beam for several fields.

As observed, the distance estimation error varies with the field, although it is not a considerable variation. By observing 5.4b, until around 80 m, the error, for all cases, is less than 1%. After this distance, it starts to raise until a maximum of 4%, corresponding, for instance, at 230 m, to a 9 m error. On figure 5.4c it can be seen the maximum distance estimation error according distance for the short range of distances, showing that, until 70 m, the error is always less than 1%, that is, a maximum error of 0.4 m until 70 m.

In order to perform a better comparison of this method in relation to the one spot per beam case, it was obtained the gain of one method in relation to the other, as plotted on figure 5.5.

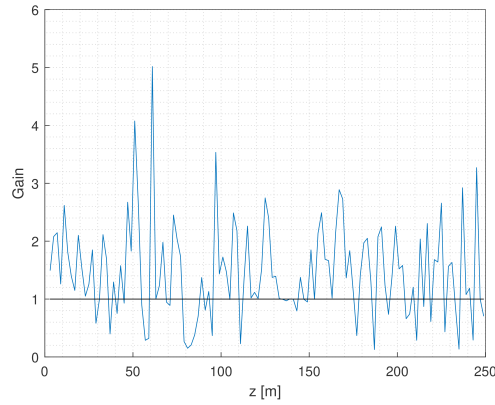


Figure 5.5: Gain from one spot per beam compared with two spots per beam method considering the maximum obtained error and an intensity-dependent scale for the centroid's estimation.

By analysing the figure above, when the gain is greater than 1 (represented by the black line), it means that the maximum error obtained by the one spot per beam method is greater than the obtained on the two spots per beam approach, meaning that the second one has better results than the one spot per beam case. The gain is greater than one on approximately 80% of the distance's range, being the average gain of ~ 1.5 , which proves the expected improvement achieved by super-resolution, that is, the validation of the concept.

5.3 Camera tilt effect on distance estimation

So far, all simulations were done considering, as referred, a camera's perfect alignment, that is, the camera's optical axis is considered to be perfectly perpendicular to the camera's baseline. In practical terms, this will never occur, therefore, it cannot be assumed in the first place. For that reason, it was simulated the camera's tilt effect on the distance estimation error, varying the tilt on both cameras, being the first referred tilt on the right camera and the second on the left one.

5.3.1 Tilt effect considering one spot per beam

Initially, it was studied the effect on the error for the one spot per beam case. The positive tilts are referred to the clockwise sense for the left camera and anti clock-wise for the right camera, being the first referred tilt concerning the right camera and the second one the left's camera, as illustrated on figure 5.6.



Figure 5.6: Scheme that illustrates the chosen referential for tilt's definition.

In figures 5.7a to 5.7d are represented the obtained results for an intensity-dependent centroid's estimation considering different tilts on both cameras.

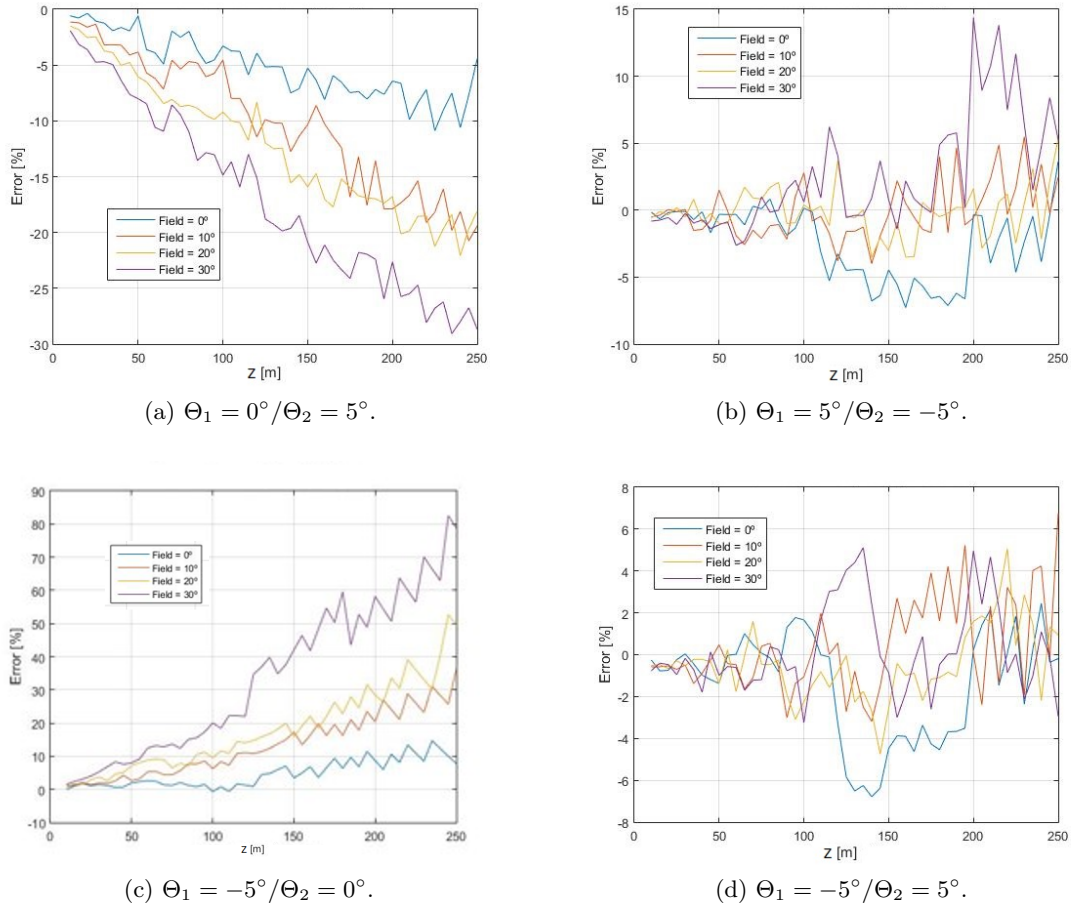


Figure 5.7: Tilt effect on the distance estimation error according distance for one spot per beam and intensity-dependent estimation.

For the case shown on figure 5.7a, it was induced a tilt of 5° on the left camera, meaning that all estimated fields considering the left camera will be offset by -5° , that is, the estimated distance will be smaller than the real distance, that is why the error is negative. Also, for example, for a field of 20° and a distance of 200m, the error reaches -20%, meaning that the camera's tilt, even if relatively small, can have a very detrimental consequence on the distance estimation.

On the other hand, on figure 5.7b, having induced a positive tilt on the right camera and a negative tilt on the left camera, it partially compensates the system's tilt. Still, at some distances, the error escalates until 15%. This means that there are cases where, even if the cameras have a considerable tilt, it can happen that these two compensate themselves, minimizing the tilt's effect. Still, 15% is a huge error, meaning that the tilt must be nevertheless compensated.

Considering figure 5.7c, the produced effect is simply the opposite of the one of figure 5.7a. Finally, on figure 5.7d is occurs the same effect as on figure 5.7b.

5.3.2 Tilt effect considering two spots per beam

The same simulations were made considering two spots per beam, being the general trends, as expected, the same as in the single spot cases, as illustrated on figure 5.8. The major difference is on the $5^\circ / 5^\circ$ case, where, for the two spots per beam case, the error, in general, is smaller than the one on the one spot per beam case; still, considerable when compared to the non-tilted cases.

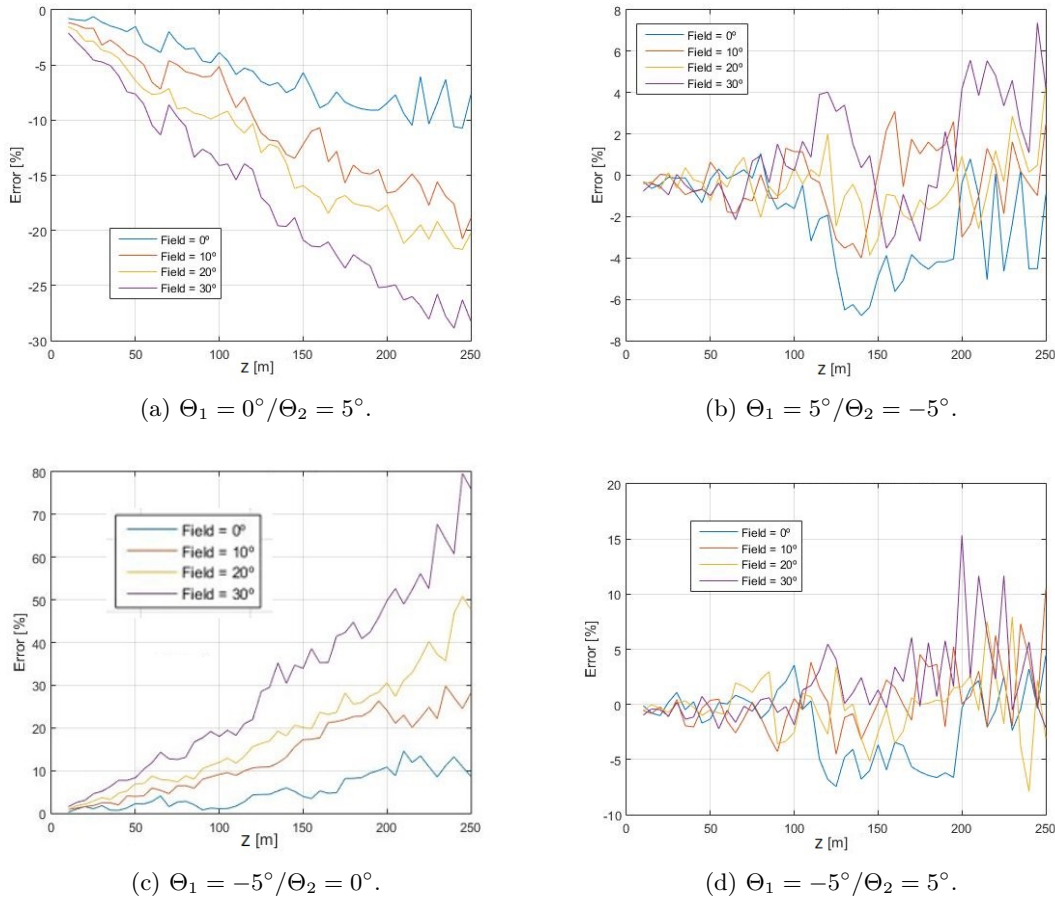


Figure 5.8: Tilt effect on the distance estimation error according distance for two spots per beam and intensity-dependent estimation.

After simulating all the possible centroid estimations for different tilt combinations, it can be concluded that the tilt has a very detrimental effect on the distance error estimation and that a tilt compensation stage for each camera is required.

5.4 Single-camera LiDAR

So far, it has been considered the use of one or two spots per beam in order to estimate the centroid and determinate a spot's distance. Considering that there are two cameras and two spots, there are three possible combinations that can be made, as illustrated on figure 5.9.

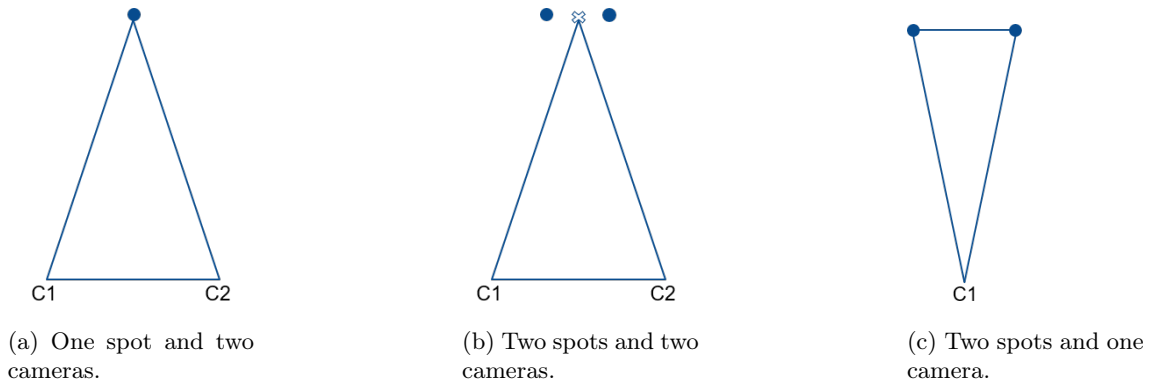


Figure 5.9: Possible combinations of cameras and spot(s) for centroid's estimation.

The first illustration, 5.9a, represents the estimation case for one spot per beam. The second illustration, on 5.9b, represents the two spots per beam case, being the cross in the middle of both spots the effective centroid. Lastly, one can try to estimate the distance, z , by using only one camera and two spots.

As verified throughout the simulations, when two spots per beam, the distance between the pair seen at the sensor gets smaller as a function of distance, z . Having verified this, it was studied the ideal distance between the centroids, that is, considering the geometric centroid for each spot. The results are depicted on figure 5.10.

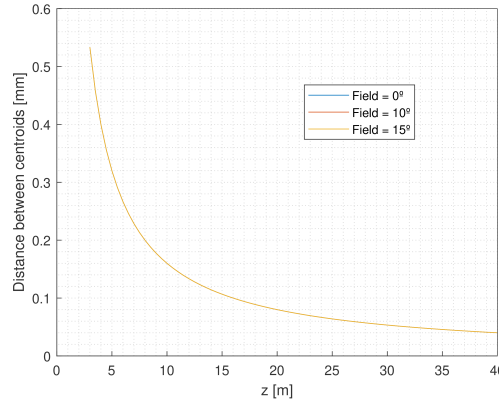


Figure 5.10: Distance between the geometrical centroids according to the spot's distance for several fields.

Up to a distance of around 20 m, the variation on the distance between centroids is very noticeable, although, for distances greater than that, the variation according to the spot's distance starts to be less perceptible, which leads to conclude that, for larger distances, this method is not suitable because there is not enough resolution. Also, as seen on figure 5.10, the distance between centroids does not depend on the spot's field, considering the ideal centroid's estimation.

Now, instead of considering the ideal case, that is, the geometric centroids, these were estimated using the intensity-dependent centroid estimation. On the following figures 5.11a and 5.11b, it is represented the distance between centroids according to the spot's distance, for several fields. The distance between centroids is not influenced by the spot's field and there are no significant differences on the obtained graphs for both sensors. The small

differences on the curves, for each field, are due to the sensor's resolution.

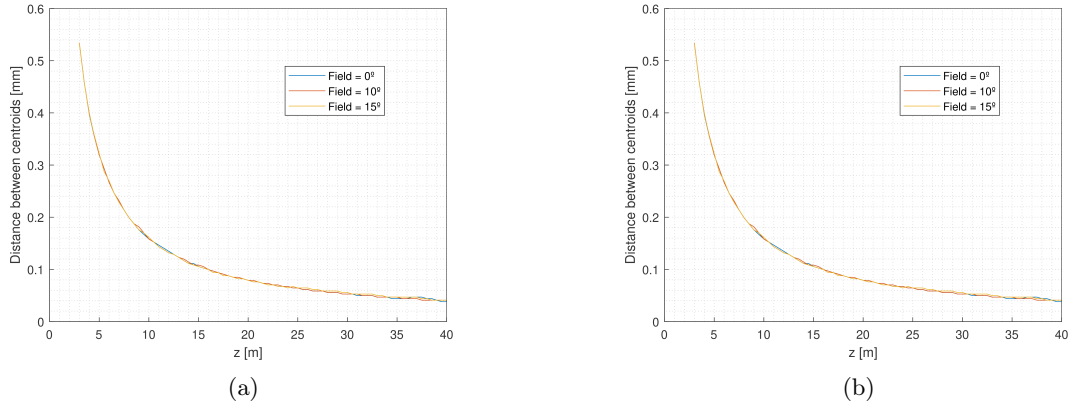


Figure 5.11: Distance between centroids according to the spot's distance for an intensity-dependent scale for the left (a) and right (b) sensors, for several fields.

After estimating the centroids and having the correspondence table between the distance between centroids and z , it was obtained the estimation error using this method, as illustrated on figure 5.12.

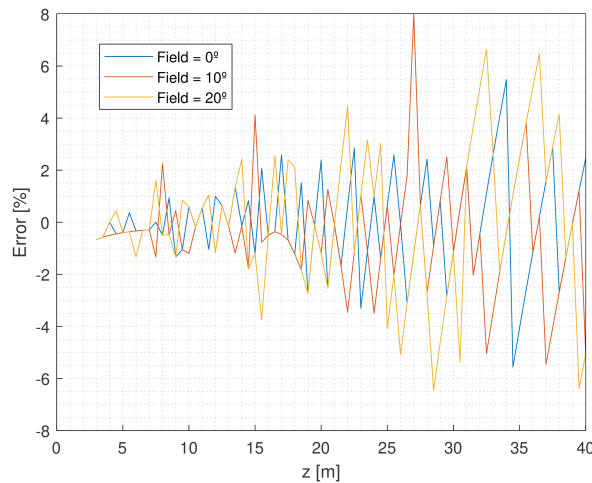
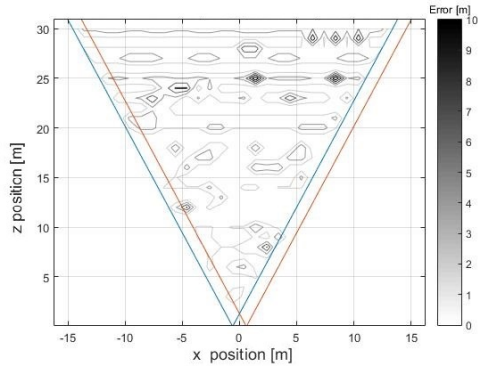


Figure 5.12: Distance estimation error according to the spot's distance based on the distance between centroids for a field of 0° and intensity-dependent scale.

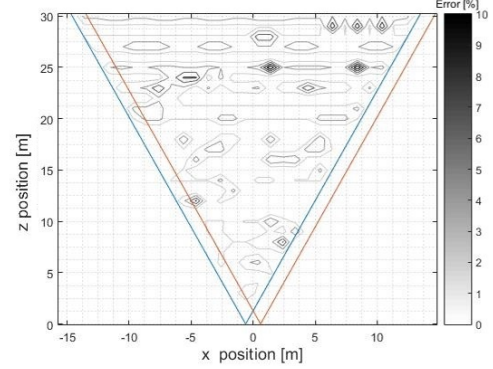
On the figure above it is clear the error's saw-tooth behavior, that, again, is due to the discretization made by using a look-up table. Comparing the errors of figures 5.3a and 4.9a, that is, the estimation methods that use two cameras, the error is greater using this method than using the one or two spots per beam estimation methods because the distance between centroids has a very small variation with the distance and the sensor's resolution is not enough. From 5 m to 15 m it varies 0.2 mm and, for example, from 15 m to 35 m varies 0.05 mm at the sensor. More precisely, at a distance of around 20 m, the error starts to grow even more, proving the unsuitableness of this method.

5.5 Results

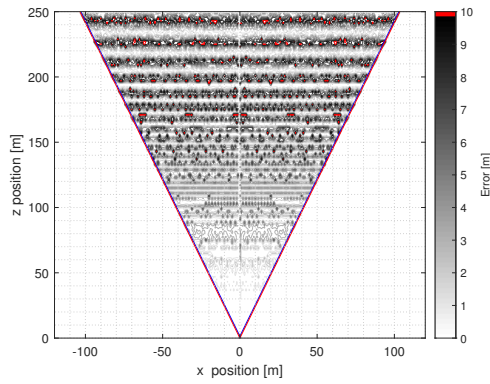
In order to compare in a better and more intuitive way the error for all types of estimations, the following graphs represent the estimation errors for several cases. Considering the distance between cameras, the blue and orange lines represent the FoV of each camera and, therefore, their common viewable area.



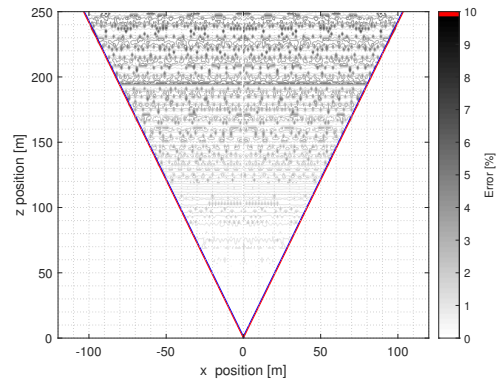
(a) 1 camera, 2 spots per beam.



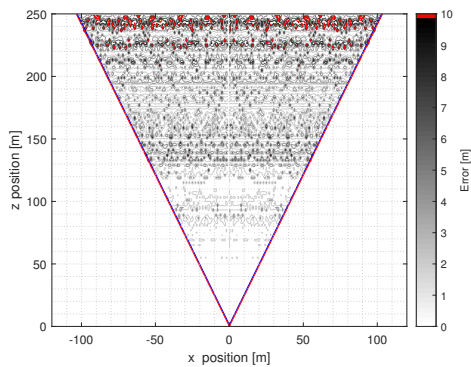
(b) 1 camera, 2 spots per beam.



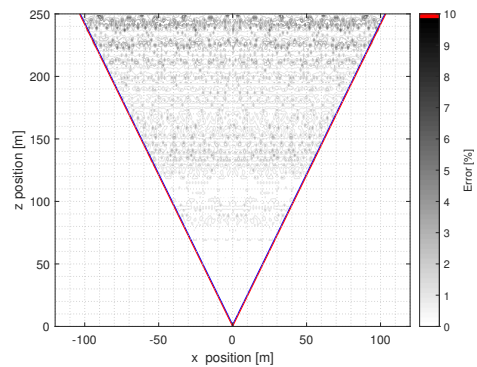
(c) 2 cameras, 1 spot per beam.



(d) 2 cameras, 1 spot per beam.



(e) 2 cameras, 2 spots per beam.



(f) 2 cameras, 2 spots per beam.

Figure 5.13: Distance estimation error according spatial position of the spot(s) for intensity-dependent centroid estimation, being the common area between the orange and blue lines the viewable area, corresponding each one to each camera's FoV.

On figure 5.13c is represented the distance error estimation considering a one spot per beam distance estimation. The closer the obstacle, the smaller the error, that is why the the graph is lighter the closer it is get to the receiver. The further the object is, the darker the points, both for the graph in meters and in percentage. For a matter of simplicity, it was chosen to put all points with a distance estimation error bigger than 10 m or 10% with the colour red. Above the 200 m, a series of points in red start to appear as seen on figure 5.13c, demonstrating the larger error values for larger distances.

Lastly, the same graphs where obtained considering two spots per beam and two cameras, in figures 5.13e and 5.13f. It is immediate to conclude the difference, for better results, between the single and double spot approaches, validating the super-resolution approach.

Another way of comparing these approaches is to plot the estimated distance according the real distance, in meters, for both one spot and two spots per beam and considering an intensity dependent and independent methods, as illustrated in figures 5.14a and 5.14b.

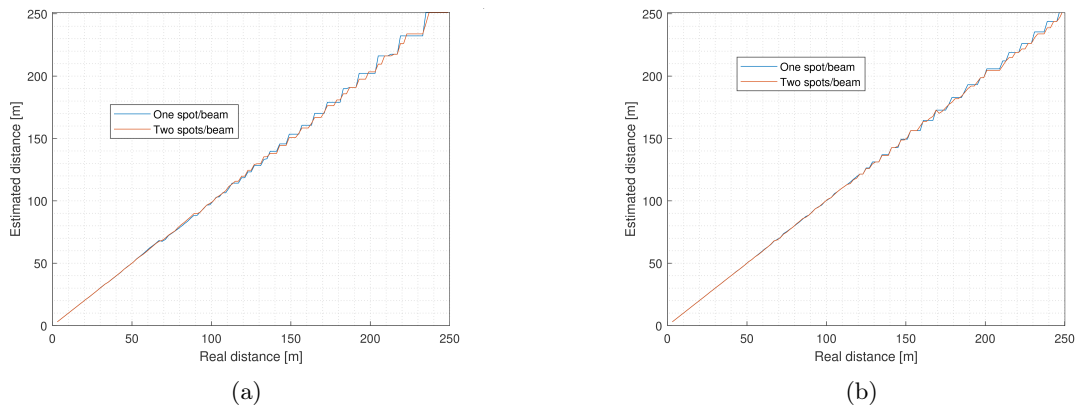


Figure 5.14: Estimated distance according real distance, z for intensity dependent (a) and independent (b) scales considering one and two spots per beam.

If every distance was ideally estimated, the graphics above would be straight instead of showing a stair-like variation, which represent the distance estimations made with a certain error. The less straight the line, the bigger the error.

Considering all the previous obtained results, the following conclusions can be taken:

1. The single camera method is not usefull taking into account the obtained results. The single camera method could be helpful to “see” on the dark zone, that is, the area that is not covered by both cameras FoV, still, that zone is very small, so, it is not very serious the fact it is, indeed, a dark-zone. Comparing to the dual camera method, the error is much higher, which causes this method to be discarded.
2. Considering the use of two cameras, the comparison is between the use (or not) of 1 spot at a time or considering 2 spots per beam. As seen on the figures above, until around 100m there are no major differences, but after that it is clear that the use of pairs of spots has a better performance for greater distances because the lack of resolution has a stronger impact for the medium and long ranges.
3. Considering one spot per beam, the average gain is around 1.5 passing from the intensity-independent for the intensity-dependent method. The average gain (in

terms of number of pixels) from 1 to 2 spots per beam, considering an intensity-dependent scale, is approximately 2, which sustains the obtained improvement of estimating centroids using pairs of spots.

4. However, these results do not take into account a possible tilt that the cameras might have. As simulated, the camera's tilt has a very serious effect on the distance estimation's error, leading to huge error percentages even for small tilts. It is safe to say that the tilt's effect can be compensated by software and, if that effect is eliminated, in practice, a very low error percentage can be achieved.

Chapter 6

Decoupling LiDAR receiver from LiDAR transmitter

So far, only ideal scenarios were considered, that is, ideal spots without any interference, being assumed that the illumination pattern was known. In a real life scenario this will not happen, which makes it imperative to study the effects of possible interference.

By interference it is meant cases where, instead of having well distinguishable spots, an interfering spot can be between a pair of spots, figure 6.1a; an interfering spot can be partially overlapped with one of the spots, figure 6.1b; or, simply an isolated interfering spot that is not part of the LiDAR illumination pattern, figure 6.1c, being the origin of these interfering LiDARs from other vehicles.

At the sensor, the system cannot distinguish between non-interfering and interfering spots because what it “sees” are intensity values on each pixel. When an interfering spot partially overlaps another, it will cause the centroid to shift, possibly originating a wrong estimation and, consequently, a larger error. Still, there may be cases where the interference might not have any impact on the distance estimation, as will be explained.

Therefore, it is important to study all the possible interference cases and how to deal with them in order to, in case, minimize the estimation error they might cause, or even finding ways of taking advantage of the interference.

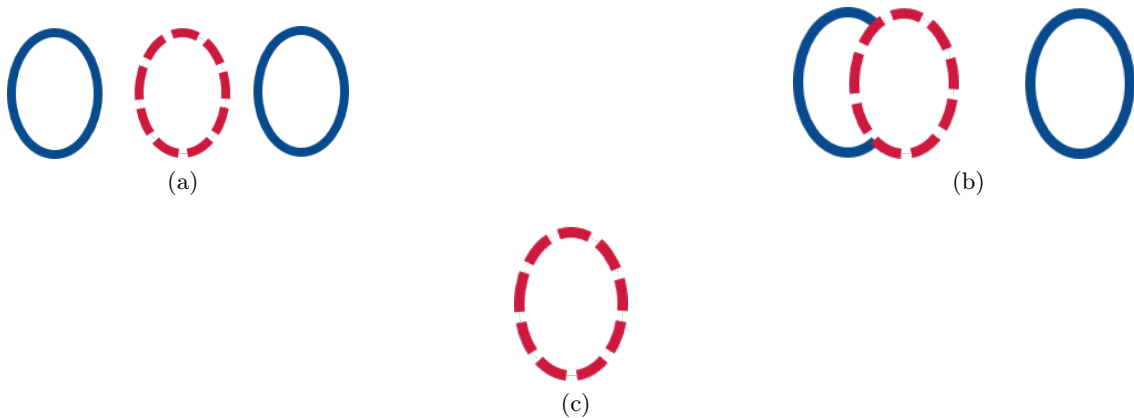


Figure 6.1: Types of interference: (a) interferer (dashed) between a pair of spots, (b) interferer that overlaps with one spot of the pair and (c) single interferer.

In order to minimize possible adverse effects of these interferences, an algorithm was design to, being given a frame with several spots and the respective centroids, find possible interferers and correct them, as will be explained in the next section.

6.1 Operation principle

The receiver processes the sensor image according to the algorithm described on figure 6.2, which is explained as follows:

1. Starting from a single frame, the first step is to locate every resolvable spot, which may or may not have an overlapping interfering spot;
2. The distance is estimated considering one spot per beam;
3. The pair of each spot is searched: possible pairs are assumed to be on the same horizontal line of pixels and within a certain limit area (of pixels). For instance, for a spot 10 m away, the distance, on the sensor, of one pair, is between 8 and 10 pixels; for 100 m, between one and two pixels and for 200 m, less than one pixel. Possible pairs that not comply with such rule are discarded;
4. If a pair is found, then the distance is estimated considering two spots per beam;
5. If no pair is found, the distance is estimated considering one spot per beam.

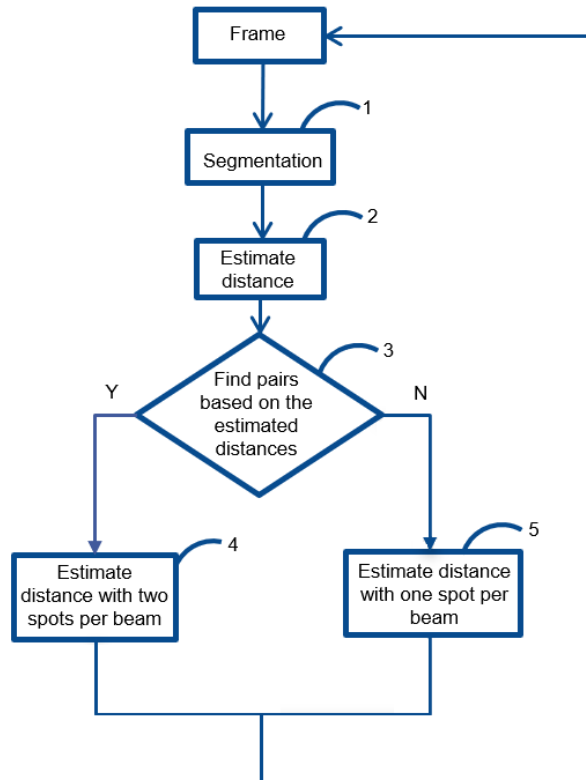


Figure 6.2: Diagram illustration of the illumination independent algorithm for processing detected spots.

6.2 Types of interference

As illustrated on figure 6.1, there are three main different ways of interference: a non-overlapping spot between a pair, an overlapping spot and a single interferer spot. For a matter of simplicity, the correspondence is addressed relatively to pairs on each frame, considering, *a priori*, that the correspondence between each spot on both sensors is unequivocal.

Throughout all of this section, the spots represented in blue are the ones that are part of the LiDAR's illumination pattern and those dashed in red are interferers, being all centroid estimations done considering an intensity-dependent scale.

6.2.1 Overlapping interference

In practice, in case there is an overlap, the system is unable to know, on the first place, that two spots are overlapped. Figure 6.3 represents a pair of spots with an overlapping interferer, in red, for both sensors, in order to understand how the centroid estimation is done when there is an overlap. In the middle are represented the pixel's intensity diagrams for each overlapping spot that is used to determinate the centroid, where are represented the numerical position of the hit pixels on the horizontal axis and the number of rays that hit each pixel on the vertical axis.

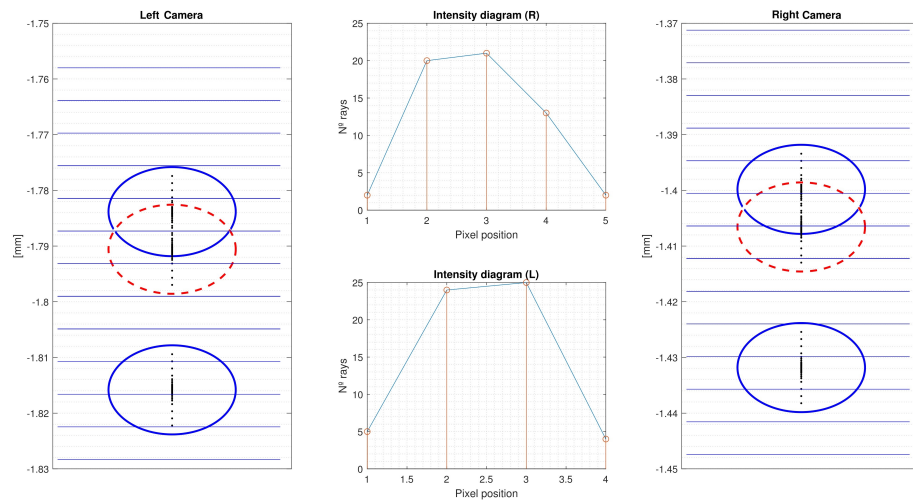


Figure 6.3: Spot and pixel intensity diagrams for a pair of spots with an overlapping interferer on both sensors for a field of 5° and $z = 50$ m.

As observed, for the right sensor, the spot and its interferer occupy 5 pixels, being hit, on each pixel, by a different number of rays. For each pixel, the number of rays is determined, originating the pixel's intensity matrix (Intensity diagram (R)), which stores the number of rays per pixel on the sensor. Here, the algorithm finds the most intense pixel and considers it as the centroid. For the right and left sensors, the resulting centroid for the overlapping spots with and without interference was, respectively, -1.4035 mm / -1.4035 mm and -1.7873 mm / -1.7814 mm, using an intensity-dependent centroid estimation.

So, it can be concluded that, for the right sensor, the interferer did not make any change to the centroid but, on the left sensor, it moved the centroid, originating a distance estimation error of 0.204% against the 0.09% that would have been obtained in a non-

interference case. This means that a partially overlapping spot can increase the distance estimation error, depending on the degree of overlap. In case there is a complete overlap, then nothing changes.

6.2.2 Non-overlapping interference

The next possible interference case is the one plotted on figure 6.4, which represents a pair of spots from the LiDARs pattern and an interferer in the middle, with no overlap.

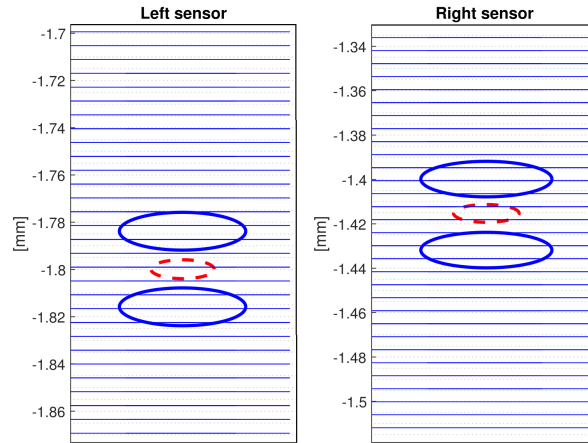


Figure 6.4: Pair of spots with a non-overlapping interferer on both sensors for a field of 5° and $z = 80$ m.

For this particular case, in more detail, the algorithm does:

1. Estimates each individual centroid, obtaining, in this case, three distinct values;
2. Estimates the distance considering each individual centroid;
3. Assuming that a spot's pair is the one below (according figure 6.4) or the second one (in case there is an interferer in the middle), the algorithm creates a table with these possible combinations of pairs;
4. For each combination, it is obtained the distance between centroids;
5. By using a look-up table that corresponds the spot's distance with the minimum distance that a pair of spots are from each other, it is obtained a reference distance between centroids for that estimated spot's distance, z ;
6. The obtained reference distance is compared with the obtained distance between centroids;
7. If the distance between the first spot (in blue) and the interferer is shorter than the reference one, it means that spot is not the correct pair, being the next spot in line considered as the correct pair;
8. When found the pair, the distance is estimated considering two spots per beam;
9. The pair and the interferer are discarded from the frame.

So, in the case represented on figure 6.4, the algorithm identifies the interfering spot because it knows that, for $z = 80$ m the interferer is too close to the first spot to be its pair, discarding it, and estimates the distance considering the two spots in blue, having obtained an error of 0.944%. Having this result it can be affirmed that the algorithm was successful, having correctly identified the pair and discarded the interferer.

6.2.3 Isolated interference

Lastly, the most obvious case of interference is when there is a single “outsider” spot, that is, a spot that is not part of the LiDAR’s illumination pattern, represented in dashed red on figure 6.5.

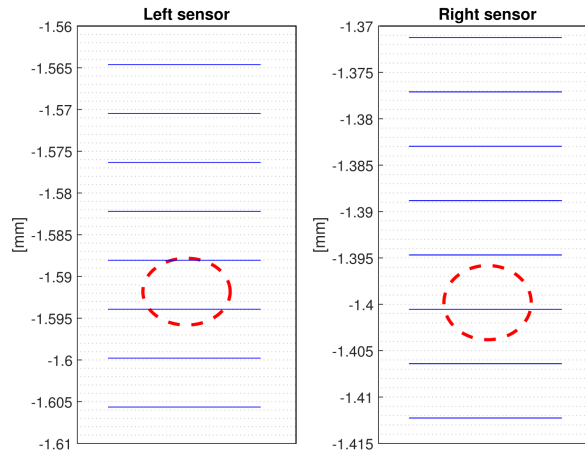


Figure 6.5: Single interferer spot on both sensors for a field 5° and $z = 80$ m.

In this case, the algorithm simply obtains the centroid and, since there is no other spot in the frame, the distance is estimated considering one spot per beam, being the obtained error of 0.8264%, that is, the error obtained considering one spot per beam and an intensity-dependent scale. Since the system does not know that this spot does not belong to its LiDAR illumination, this interferer does not do any prejudice, providing, in fact, an extra point for the point cloud.

6.3 Interfering spots: a curse or a blessing?

As seen on the previous sections, there are mainly three types of interference that can happen: when there is an explicit overlap of the spots, when there is a interfering spot in between a pair of spots and an isolated interferer.

In practice, it can happen that an overlapping interferer shifts the centroid to a more correct position, leading to a smaller error. On the other hand, it can shift the centroid to a further position, leading to an even worse error. Also, in case there is a complete (or almost) overlap, nothing happens and all centroids are estimated as if there was no interferer. The other good aspect about this approach is that a single interferer from any other source is used as if it were part of the LiDAR illumination pattern, working as extra information, not causing any problem.

After studying these cases, it can be concluded that the proposed LiDAR is very robust against interference, not needing very sophisticated algorithms to detect of correct interference, being also possible to escalate the present algorithm to consider a greater number of interferers.

Chapter 7

Experimental validation

An experimental validation was produced with the following goal: experimentally prove the previously simulated concept of estimating a distance using two cameras (triangulation), illuminating the scene with spots, in a real scenario.

7.1 Setup

The experimental set-up includes two cameras (IDS Imaging UI-3260CP-M/C) and objectives with a focal length of 16 mm (Thorlabs - MVL16M1), each with a manually adjustable focus, as shown on figure 7.1. Both cameras have a monochromatic sensor with a resolution of 1920x1200 pixels and a pixel size of $5,86 \times 5,86 \mu\text{m}^2$, being connected to the computer by an USB 3.0 connection. The cameras are placed over a linear guide (IGUS - WS-10-40-SL-1500), being over a slidable adapter (IGUS - WW-06-30-06-S) that allows to change and fix the cameras to the rail, as shown on figure 7.2. The rail with both cameras is placed over the two tripods (Rangers - B01FVW235G) so all the tests could be performed.

Concerning the laser (Thorlabs - HL6738MG), shown on figure 7.3, and the cage where it is mounted, represented in figures 7.4a and 7.4b, it works at a wavelength of 690 nm, which makes part of the visible spectrum but is almost on the infrared area, allowing a maximum power of 30 mW. A PCB was made so that the laser power could be adjusted to 1 mW to fulfil class-1 eye-safety standards [7], being powered by a mini-USB connection. The laser cage also includes a collimating lens (LightPath - 354057Y-00-MT) in order to collimate the light to produce the spot. The laser was mounted on a breadboard and placed in-between both cameras at the same height. The setup includes two lasers spaced by 10 cm, as shown on figure 7.5.

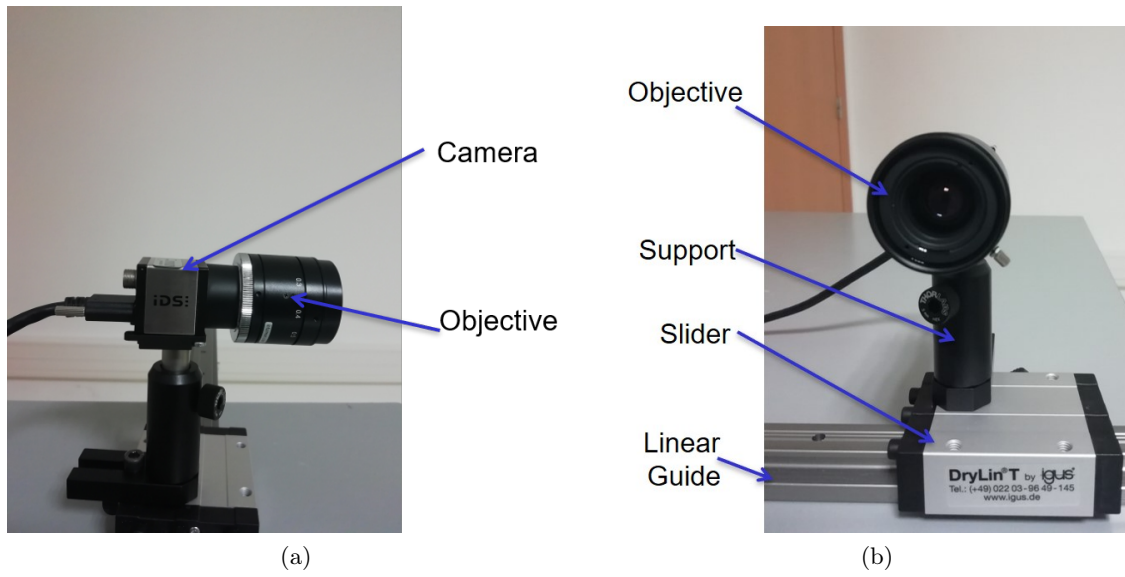


Figure 7.1: (a) Side and (b) front view of the cameras used in the experimental setup.

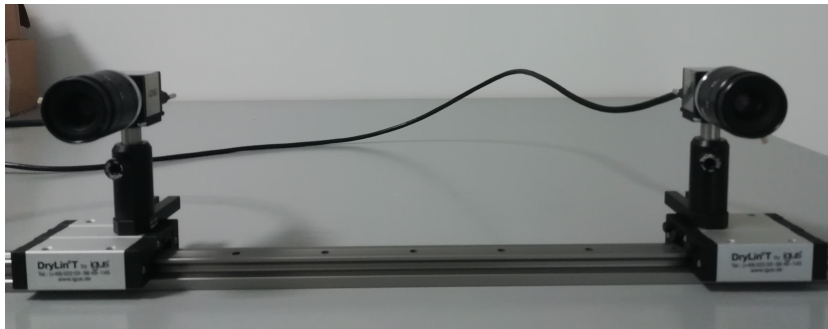


Figure 7.2: Front view of both cameras on the slider, spaced by 1.2 m.

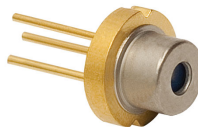


Figure 7.3: Laser used to produce a spot [42].

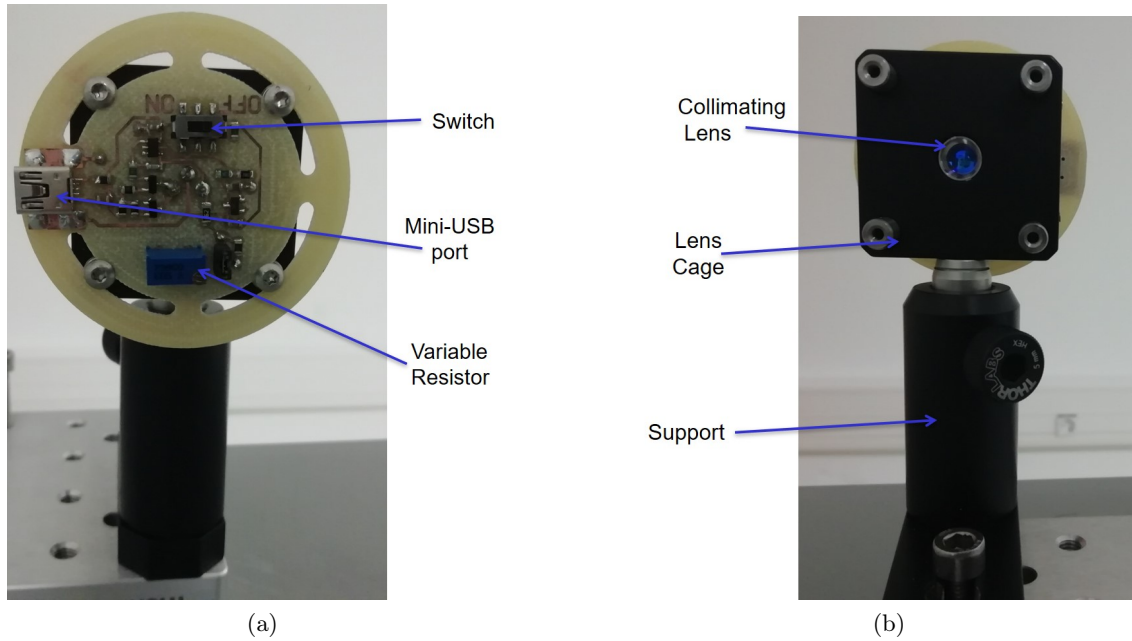


Figure 7.4: (a) Rear view of the laser containing a PCB with a variable resistor that allows to vary the laser power and (b) front view of the laser including the collimator lens and respective cage.

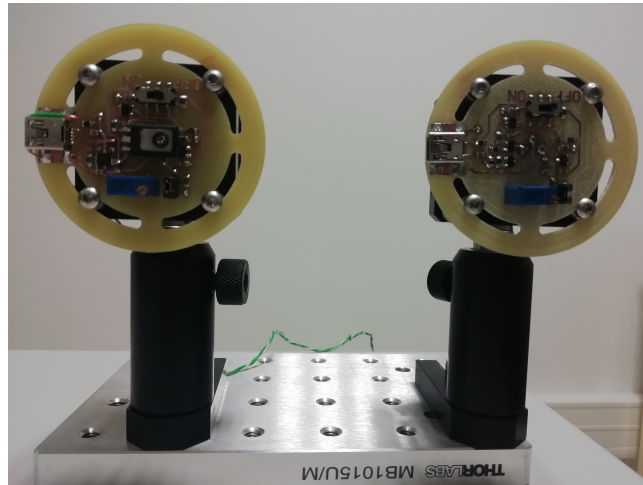


Figure 7.5: Rear view of both lasers side by side needed to produce a pair of spots spaced by 10 cm.

7.2 Camera calibration

When dealing with cameras, it is first necessary to calibrate them. Camera calibration consists on estimating the distortion coefficients and both intrinsic and extrinsic parameters [43] that will be necessary to obtain, in practice, α and β . These parameters can be used to correct lens distortion, measure the size of an object in world units, or determine the location of the camera in a certain scene. These tasks are used in applications such as machine vision or to detect and measure objects.

In order to estimate these parameters are necessary 3D world points and their centroids on the sensor. These correspondences can be obtained using multiple calibration patterns, that is, patterns that contain very sharp forms like squares or rectangles. For that reason, and as usually done, a chessboard pattern was chosen.

For the calibration, there were used a set of 30 photos, being the goal to catch all the camera's FoV with the chessboard. Figure 7.6 represents two examples of the pictures taken for calibration purposes. It should be noted that the calibration process is the same for each camera and that every time the focus is changed it is required to calibrate the camera again.

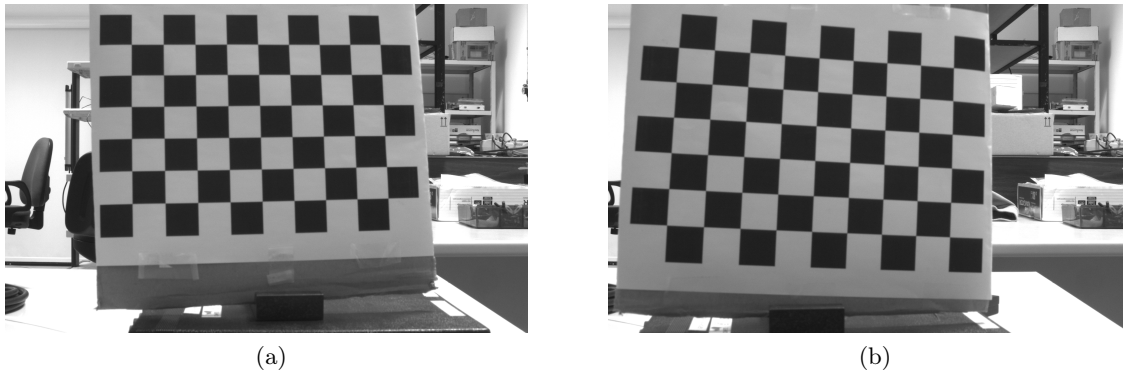


Figure 7.6: Calibration procedure: chessboard in several positions on the camera's FoV.

To obtain these parameters, it was used Open Computer Vision (Open CV) due to the several available functions concerning camera calibration, as well as for the fact that it is an open source software, being the most used tool by the scientific community and industry. The intrinsic parameters include the optical center and focal length of the camera, being usually presented with a matrix K [44], given by:

$$K = \begin{bmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (7.1)$$

where α_x and α_y represent the focal length in pixels on the xx and yy axis, respectively, s is a skew coefficient, which is non-zero if the image axes are not perpendicular, and x_0 and y_0 are the optical center coordinates, in pixels.

7.3 Estimating α and β

In order to determine the referred α and β (see figure 4.3) it is necessary to consider some concepts such as the camera's optical axis and the angle between two pixels. The optical axis is a line that passes through the centre of curvature and vertex of a lens, represented by the black dashed lines on figure 7.7.

If it is considered the pixel correspondent to the optical center (x_0, y_0) obtained by the matrix K , on 7.1, and the pixel correspondent to the spot's centroid, (x_c, y_c) , then, by definition, the angle between those two pixels, Θ , can be estimated [44].

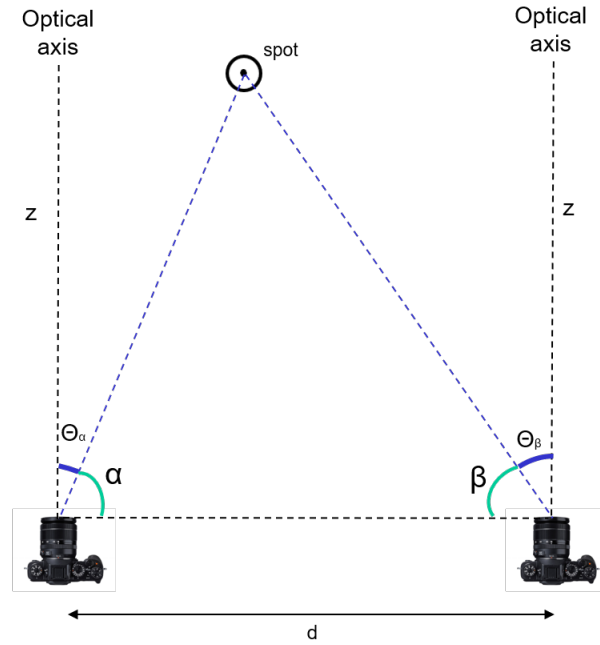


Figure 7.7: Scheme illustrating α and β and the auxiliary angle Θ obtained by calculating the angle between two pixels.

The angle between pixels is also illustrated in figure 7.8.

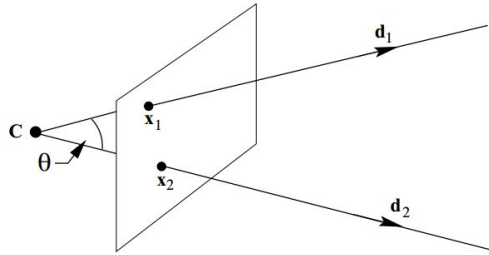


Figure 7.8: Angle between two pixels (vectors) in a given frame [44].

By knowing the matrix K , the optical center, x_1 , and the coordinates of the spot's centroid, x_2 , it is possible to estimate the angle between them, using (7.2), being, basically, the angle between two vectors, as stated on [44]:

$$\cos \Theta = \frac{x_1^T (K^{-T} K^{-1}) x_2}{\sqrt{x_1^T (K^{-T} K^{-1}) x_1} \sqrt{x_2^T (K^{-T} K^{-1}) x_2}}. \quad (7.2)$$

Considering (7.2) and figure 7.7, if it is assumed that x_1 is the vector corresponding to the optical center's coordinates and that x_2 corresponds to the spot's centroid coordinates (in pixels), then, the obtained angle is $90^\circ \pm \alpha$ (or β). For example, for the right camera: if the pixel correspondent to the spot's centroid is on the left of the optical center (as illustrated on figure 7.8), then $\beta = 90^\circ - \Theta$, otherwise, $\beta = 90^\circ + \Theta$. The same reasoning is valid for the left camera. This way, α and β are obtained and, knowing d , the distance z is estimated.

7.4 One spot per beam experiment

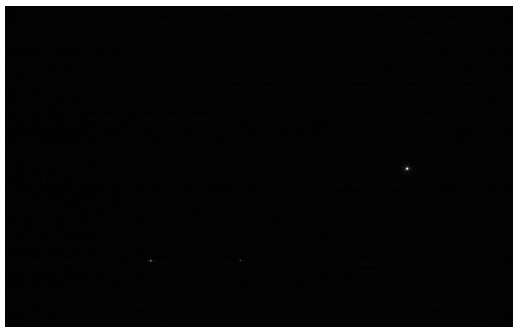
7.4.1 Centroid estimation

The first and simplest test case was the one spot per beam scenario, that consisted in having a laser illuminating a scene/object with a one spot per beam. The first test consisted, then, in taking two pictures of that spot, in a dark room, for several distances.

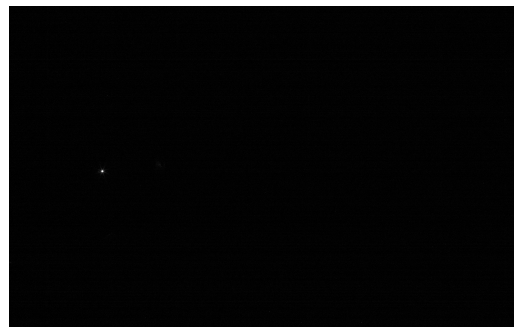
Considering the cameras' displacement, that same spot will be in different pixels on both frames and on the same line of pixels, assuming, for simplicity, that the cameras were perfectly aligned and on the same exact height. In order to determine x_2 (the centroid's coordinates), it was taken a picture of a single spot, as seen in figure 7.9a, being the spot the white circle, also called a "blob" [45], that is, a group of connected pixels in an image that share some common property, for instance, a gray scale value.

Even though the experiments were made in a dark room, it was still necessary to apply a filter, that is, a certain threshold to the intensity values on the pixels to be considered for the centroids, simply using the *threshold* function on OpenCV. After applying the filter, the blob's center can be found using "image moments" [46]. Image moments is a particular weighted average of image pixel intensities, with the help of which it can be found some specific properties of an image (in this case, the spot), like radius, area and centroid; in this case, the first order moment. Having identified the spot on the sensor, this method returns the exact centroid's position in pixels, x_2 . Finally, having the matrix K and the pixel's coordinates, the angles are directly obtained from (7.2). This procedure is the same for both cameras.

As it can be seen on both figures 7.9a and 7.9b, the same (and unique) spot on these frames occupy very different x coordinates. The closer the object hit by the spot is, the bigger the displacement; the further it is, the more similar are the centroids positions on each frame. This evident difference corroborates the triangulation concept.



(a) Left camera.



(b) Right camera.

Figure 7.9: Obtained frames in a dark room containing one spot per beam for $z = 2$ m.

7.4.2 Distance estimation error

After validating the centroid's estimation, some cases were tested in order to obtain a plot relating the distance estimation error according to the spot's distance, z .

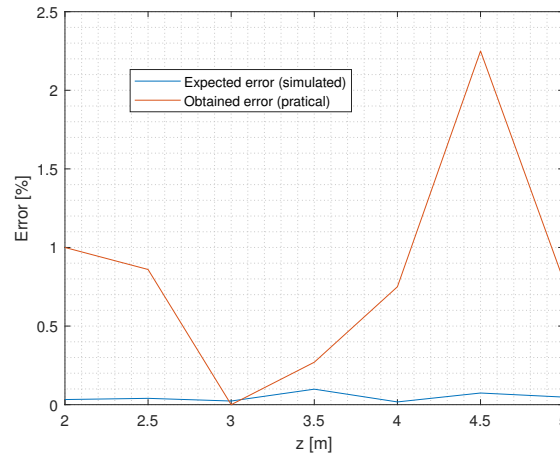


Figure 7.10: Distance estimation error in percentage for single spot estimation with a baseline distance of 1.2 m.

The results on figure 7.10 were obtained with the cameras aligned (as best as possible) perpendicularly to the baseline. The laser that represented the spot was put on the middle of the baseline, perpendicularly to it, being that approximately equivalent to a field of 0° . Due to the limited available space to perform the experiments, the measured distances were varied from 2 m to 5 m, being each point taken every 50 cm. The maximum obtained error was around 2.2%, still, the results were expected to be much better considering that the simulated error for this case was much smaller.

The most important conclusion from these results is that the camera's tilt has an important impact on the distance estimation error. As simulated before, a tilt of 5° , even if only on one camera, has a huge impact on the error, even for short distances. So, even a relatively small tilt has a considerable impact on the error. This problem can be corrected by manual alignment only to a certain extent, meaning that an algorithmic solution must be devised.

7.5 Two spots per beam experiment

Even though the obtained practical results for the error considering one spot per beam were not as good as expected, it was still tested considering two spots per beam in order to see the influence it would have on the distance estimation error. In practice, this scenario was tested by using two identical laser sets (laser and collimating lens), with the same power, placed next to each other and spaced by 10 cm, as shown on figure 7.5.

7.5.1 Centroid estimation

To estimate a centroid on a frame that only has one spot is simple: the following question is how to find several centroids on the same frame. Considering that the *moments* method only returns one blob ("moment") at a time, it was necessary to find a function that identifies several "blobs" and returns their center.

The initial filtering is done the same way as for the one spot per beam case. Then, it was used the *HoughCircles* function (uses the *Canny* algorithm), which finds circles in a gray-scale image using the Hough transform [47]. This method returns an array of “circles”, that is, a vector that contains the center of each circle (x, y) and its radius, in other words, the centroid of all detected spots. The correspondence between frames is trivial, considering that there are only two spots per frame. After having the spot’s centroids for both cameras, illustrated on figure 7.11, it is calculated their average and the remaining procedure is the same as for the one spot per beam case.



Figure 7.11: Obtained frames in a dark room containing a pair of spots for $z = 3$ m.

7.5.2 Distance estimation error

Considering two spots per beam, the distance estimation error was estimated using the same setup and procedures as in the previous section, being the obtained results plotted on figure 7.12.

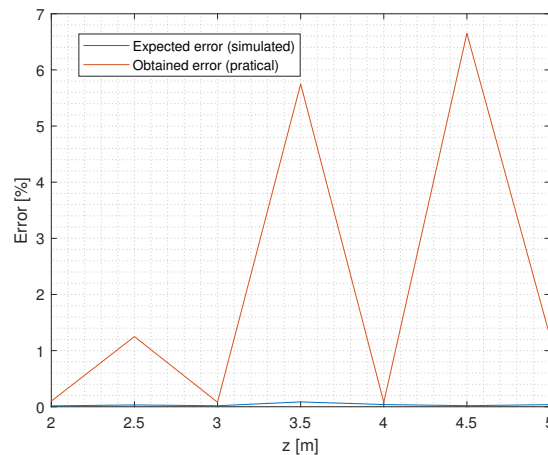


Figure 7.12: Distance estimation error for two spots per beam estimation with a baseline distance of 1.2 m.

In this case, even though the error gets close to zero at some distances, it gets to relatively high values at some points as well, reinforcing the idea that it will be necessary a software correction for the tilt. It was tried to correct the camera’s tilt “by hand”: at

some distances the results got better, but at the same time, others got even worse, leading to unsatisfactory results.

Still, the problem is the same as in the one spot per beam case: camera tilt introduces a lot of error on the distance estimation, being necessary, as well, to compensate that error in order to obtain reasonable results.

7.6 Point cloud generation

When able to obtain a point's position in the 3D space, the next step is to create a point cloud of a certain environment. A point cloud is a collection of data points defined by a given coordinates system. Point clouds are usefull because allow to capture important information about an object like its distance and shape, leading to a very high detail of a certain scene.

After correcting, even if just manually, the centroid's displacement due to the camera's tilt, it was a made a 3D mapping of a certain scenario with two objects placed at two different distances and positions, as seen of figure 7.13. On figure 7.14 is represented the obtained point cloud for the chosen scenario.



Figure 7.13: Real scenario used for obtaining a point cloud.

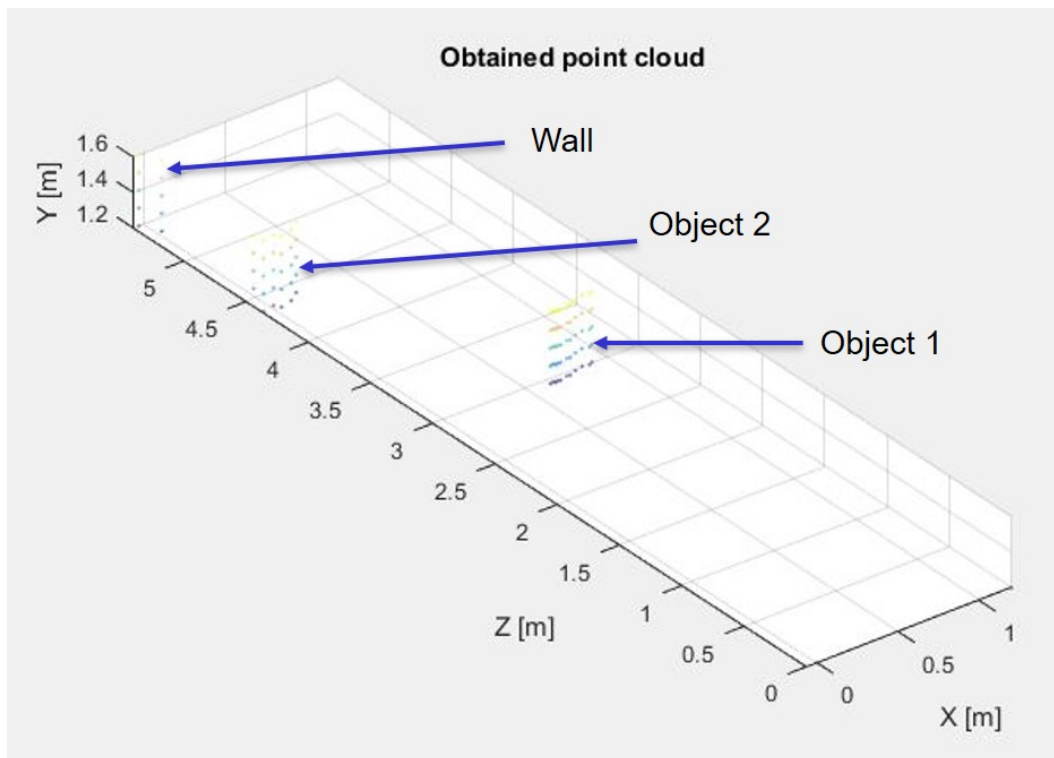


Figure 7.14: Obtained point cloud.

The first two clusters refer to the two placed objects and the further one refers to a wall. These results show what can be obtained with this system in terms of a point cloud to be later used for segmentation and identification. This one was obtained point by point, that is, a pair of frames was taken considering one spot per beam and the distance and position estimated.

Chapter 8

Performance limits

All the simulations and experiments presented were based on fairly ideal and unrealistic conditions. In reality, performance is limited by multiple factors.

This chapter addresses the performance-limiting factors like eye safety, noise and background light, diffraction limit, the non-ideal form of the spot, aperture limit, super resolution trade-offs and others that will be discussed throughout this chapter.

8.1 Eye safety

The proposed LiDAR is based on active structured illumination that consists on a dot pattern, making it imperative that safety standards are met. Figure 8.1 represents the “eye injury hazard” as a function of the laser’s power. The possible hazard caused by a laser depends not only on its power but also on the distance that the laser is, as well as the exposure time. In this context, it is assumed that a laser corresponds to a single spot, so it has to be guaranteed that, when emitting a spot, the eye-safety conditions are fulfilled.

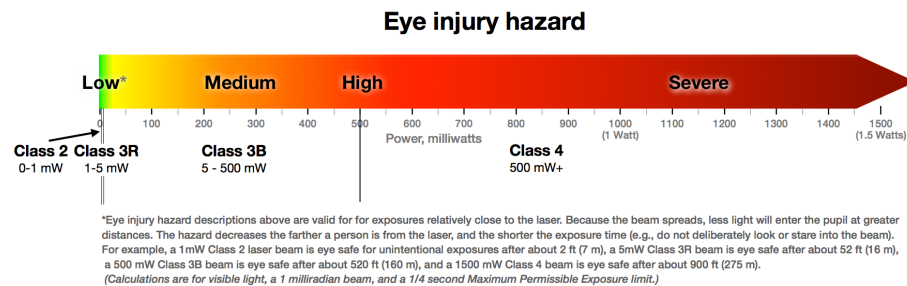


Figure 8.1: Scale of a laser’s power according to how dangerous that would be to a human eye [48].

On figure 8.2 is represented the laser's produced hazard according to wavelength.

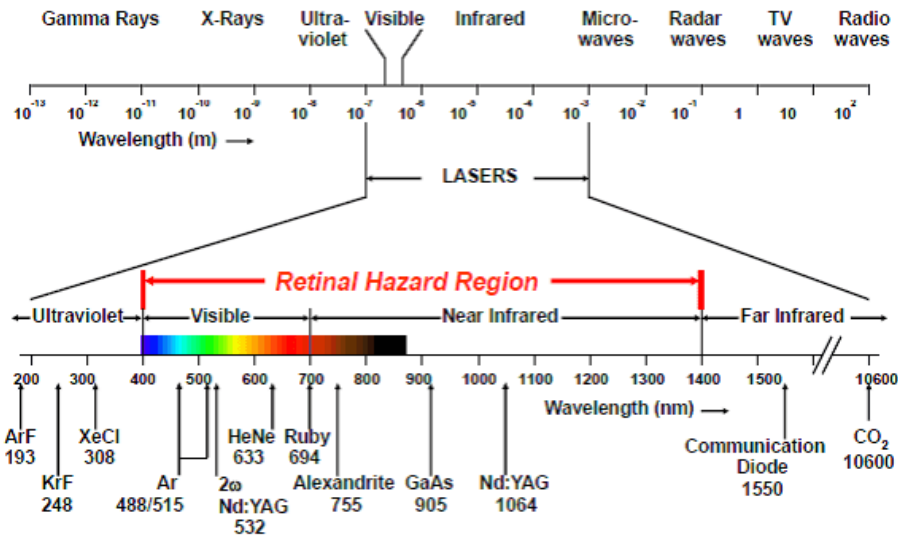


Figure 8.2: Laser's retinal hazard regions according to wavelength [7].

Lasers are divided in several classes depending on their characteristics according to specific standards.

1. Class-1 lasers are safe under all conditions of normal use: in other words, this means that the maximum permissible exposure time is never exceeded when viewing a laser with the naked eye [7]. Automotive LiDARs must be Class-1 certified devices.
2. A class-1M laser is safe for all conditions except if considering that it passes through magnifying optics such as microscopes and telescopes.
3. Class-2 lasers are limited to 1 mW continuous wave or more if the emission time is limited to 0.25 s (it is the estimated response time that an eye takes to close in order to avoid injuries). This class only exists for lasers that emit in the visible region.
4. Class-3R is considered safe if handled carefully with a restricted beam viewing. With this type of laser, the MPE can be exceeded, still, with a small risk of injury.
5. Class-3B lasers are dangerous if the eyes are directly exposed to them: these require protective eye-wear.
6. Class-4 lasers are the most dangerous ones: these can burn skin and cause permanent eye damage as a result of direct or even indirect beam viewing.

For all experiments, the power of each laser was set to 1 mW, guaranteeing class-1 eye-safety conditions.

8.2 Noise and background light

Even though the experiments were made in a dark room so that no other type of illumination interfered with the spot light, in a real life scenario, the system will be subject to several external light sources, mainly sunlight. This makes the spots less distinguishable

on the sensor, or even indistinguishable. For that reason, it is necessary to analyze the influence that background light has on the system and how does it limit the maximum achievable range of the proposed LiDAR.

In order to do so, a LiDAR equation was derived that estimates the power of the signal returning to the photodetector, given by

$$P_{r,s} = P_t \cdot \frac{1}{4\pi d^2} \cdot A_r \sigma, \quad (8.1)$$

being P_t the transmitted power, A_r is the area of the objective, σ the reflectivity of the hit object and d the distance from the target to the receiver.

The background light power depends on the FoV of a single pixel, which is of 0.024° . Assuming that both horizontal and vertical FoVs are identical and that the target is flat, the diameter of a circle captured by a single pixel is given by

$$s = 2d \tan\left(\frac{\text{FoV}}{2}\right), \quad (8.2)$$

being the corresponding spot's area given by

$$A_{BL} = \pi \left(\frac{s}{2}\right)^2 = \pi d^2 \tan^2\left(\frac{\text{FoV}}{2}\right). \quad (8.3)$$

The background light power captured by a single pixel thus is

$$P_{r,b} = A_{BL} \cdot I \cdot SW \cdot \frac{1}{4\pi d^2} \cdot A_r \cdot \sigma, \quad (8.4)$$

where A_{BL} is area shining background light into a single pixel, I is the irradiance of the background light source, SW is the spectral width of the background light captured by the sensor, here considered to be of 10 nm, and σ is the material's reflectivity, here considered to be 10 %. Around 761 nm, the value of $I \cdot SW$ is of 8.82 W m^{-2} . Replacing A_{BL} on (8.4), it is obtained

$$P_{r,b} = \frac{1}{4} \tan^2\left(\frac{\text{FoV}}{2}\right) I \cdot SW \cdot A_r \sigma. \quad (8.5)$$

Considering a laser power of $P_t = 1 \text{ mW}$, it was observed that for a continuous emission, the background light overpasses signal power from 30 m on, making it undetectable by the sensor, as seen on figure 8.3a. For that reason, it was considered to send bursts, that is, sending more power but in shorter periods of time, while maintaining the same average power (to satisfy eye-safety standards), as seen on figure 8.3b.

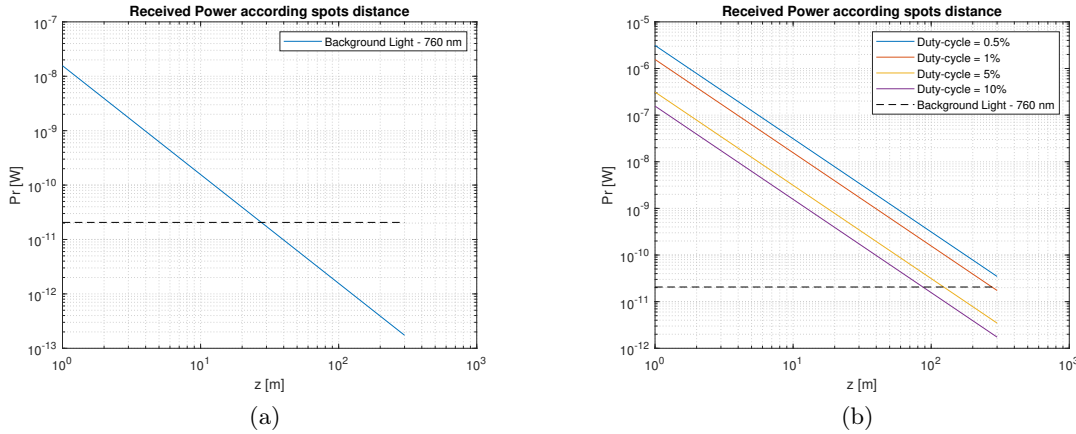


Figure 8.3: Received power according spot's distance (a) considering a continuous emission and considering (b) bursts for different duty-cycles.

In practical terms, it has to be guaranteed that the $P_{r,s} \gg P_{r,b}$ so that a spot can be distinguishable. As observed on figure 8.3b, for a duty-cycle of 0.5% a range of 300 m is reached. Knowing that the minimum exposure time of the used cameras is $30 \mu s$, for a frame-rate of 50 Hz and a duty-cycle of 0.5%, the burst duration is of $100 \mu s$, proving the feasibility of this approach.

8.3 Number of cameras vs. angular resolution

In this specific setup, there have been considered two cameras, each one with a FoV of 45° . Considering this FoV, it means that there are 1920 pixels (horizontally) for 45° , meaning that one pixel corresponds to 0.024° . The point is, if it was considered a system with more cameras, each one with a smaller FoV, to investigate how much gain would the system have and what are the physical limitations imposed by the setup.

Ideally, it could be made a system with an infinite number of cameras, each one with an infinitesimal FoV, thus obtaining an infinite angular resolution. Still, the main restraint is that the system is limited by the space occupied by the cameras themselves, meaning that there is no theoretical limit to a system's resolution but a physical one.

8.4 Diffraction limit

The resolution of an optical imaging system (like a camera) is limited by factors such as imperfections in the lenses or misalignment. However, there is a principal limit to the resolution of any optical system due to the physics of diffraction [44]. The diffraction-limited angular resolution is proportional to the wavelength of the light and inversely proportional to the diameter of the objective's entrance aperture,

$$\frac{d}{2} = 1.22\lambda N, \quad (8.6)$$

where λ is the wavelength and N is the f-number [49] of the imaging optics.

Being an instrument's resolution limited by these variables, it means that it does not make any difference increasing the number of pixels of the sensor from a certain point, that is, the diffraction limit.

8.5 Non-ideal spot generation

Ideally, all the generated spots would be perfect circumferences, but, in practice, beam waists may vary on the transverse axis according to the spot's distance, given by (8.7), where w_0 is the waist at the point of its focus, z is the spot's distance and z_R is the Rayleigh's range [50]:

$$w(z) = w_0 \cdot \sqrt{1 + \left(\frac{z}{z_R}\right)^2}. \quad (8.7)$$

Equation (8.7) allows to obtain the lowest theoretical attainable beam divergence, that is, the angular measure of the increase in beam diameter according distance from the optical aperture from which the beam emerges [51], given by (8.8), where n is the refractive index of the medium where the beam propagates and λ is the beam's wavelength,

$$\Theta \approx \frac{\lambda_0}{\pi n w_0}. \quad (8.8)$$

Since the divergence is inversely proportional to the spot's size, for a certain wavelength, a Gaussian beam that is focused to a small spot diverges as it propagates away from the focus. The relationship between beam width and divergence is a fundamental characteristic of diffraction [52]. Typical values for the beam's divergence considering a system of a laser and a collimator lens are around 0.5 mrad. If divergence is smaller than a pixel's FoV there is no problem; on the other hand, it can happen that the divergence is such that the pair of spots can overlap at the sensor.

Chapter 9

Conclusion

In this last chapter, the final considerations regarding the work developed throughout this dissertation project are presented.

In order to be able to simulate the studied concepts, a ray tracing simulation tool was designed and validated to be used throughout the scope of this thesis. Based on simple trigonometry, it was derived an equation to estimate distance, z , as well as the theoretical expected estimation error, having been concluded that an error of one pixel on the centroid estimation on one of the sensors (considering the error on the other sensor as null) implies a maximum error of 8% at 250 m; in case there is an equal variation on α and β , an error of one pixel produces a maximum error of 16%; for a maximum variation of 0.1 m on the baseline distance, the maximum obtained error is 8%.

In order to estimate the centroids, two ways were considered: using intensity independent and dependent methods. Considering one spot per beam and an intensity-independent scale, the maximum obtained error is 4%, which corresponds to maximum error of 8 m, being the maximum error until 70 m of 0.8%; For the intensity-dependent scale, the maximum obtained error is 6%, corresponding to a maximum error of 15 m at around 240 m, being the maximum error until 70 m of 1%. Assuming a systematic error of one pixel on one sensor, a maximum error of 20 m at 250 m is obtained as expected theoretically. The maximum obtained error for the intensity-independent method is smaller than for the intensity-dependent method. Still, for the long range of distances, the error increases with distance regardless because the spot's size gets increasingly small, that is, there is not enough resolution to estimate the centroids with precision. By the obtained results for the gain from the intensity-independent to the intensity-dependent method, the average gain was 1.44, which corroborates the general improvement obtained by this method.

Due to the lack of resolution, there were considered pairs of spots to estimate the centroid, that is, the super-resolution approach. For the intensity-independent method, the obtained error is less than the one obtained for the one spot per beam case, being the maximum error of 3%, which represents a huge upgrade in general. For the short range, the maximum error is 1.1%, which corresponds to an error of 0.8 m at 70 m. Considering an intensity-dependent centroid estimation, the maximum error is 4%, representing an error of less than 10 m at 250 m. For the short range, the results are better, being the maximum obtained error of 0.8%, that is, 0.4 m at 54 m.

In order to compare the improvement by passing from one to two spots per beam considering an intensity-dependent method, the average obtained gain was 1.5. Considering two spots per beam leads to better results on 80% of the times, proving that the best approach is using super-resolution.

It was also tested the influence that a possible horizontal tilt would have on the distance estimation, being concluded that it has a very negative impact on the distance estimation, being imperative to correct it.

Considering the single-camera LiDAR approach, the obtained results showed a relatively large error, that is, for a range of 40 m, the error reaches 8% comparing, for instance, to the less than 0.5% obtained for two spots per beam, making this method unsuitable for the desired goals.

A standalone algorithm that processes the detected spots on both sensors was devised, such that it would be possible to operate the LiDAR independently of the illumination pattern. The algorithm was successfully validated, also showing that overlapping interfering spots have negligible impact on estimation error, and that, isolated interfering spots can in fact be used for obtaining an extra estimation.

Reached the experimental part, it was verified that the impact of the tilt is very severe and cannot be corrected manually. In order to summarize the achieved milestones, a well-defined point cloud was obtained.

Considering performance boundaries, it is imperative to guarantee the eye-safety limits, having those been taken into account when estimating the maximum obtainable range considering background light. If the lasers emit continuously, then the maximum range would be 30 m. After simulating sending bursts with a duration of 100 μ s at a frame-rate of 50 Hz, it can be obtained a maximum range of 300 m, which satisfies the desired goals. Also, there are no theoretical limits imposed regarding resolution, and non-ideal spot generation appears not to be deleterious.

The main conclusion of the present thesis is that the potential of a triangulation-based LiDAR as an alternative to a ToF-based LiDAR is confirmed. However, such a potential can be materialized only if a suitable tilt-canceling calibration is performed and if the background light levels are kept much lower than the signal levels.

Chapter 10

Future work

Having validated the proposed triangulation LiDAR considering one and two spots per beam, the next step is demonstrating robust methods for processing entire frames with several spots.

Future tasks include extending the simulation tool capabilities to 3D that will allow to test several scenarios in a more realistic way. This new simulator should perform a weighted-average centroid estimation, as well as being able to simulate the real impact that horizontal and vertical tilts can have on the distance estimation. This will be followed by the experimental validation of the simulated cases with the final goal of implementing a functional real-time prototype.

After the main tasks to be done are complete, upgrades such as a possible extension to a multiple camera scenario, adaptable illumination schemes or automatic calibration can be explored. Thus, on the next sections, the referred topics will be discussed in more detail.

10.1 Simulation

10.1.1 Extension to 3D

So far, all the previous simulations have been done in 2D, that is, considering that all the simulated spots (that is, obstacles) were displaced on the same horizontal plane as the camera's optical axis at different distances. The simulator does not comprise points in different vertical positions, which limits the type of simulations that can be done. An extension to 3D would allow to simulate objects at all the possible vertical and horizontal positions on the camera's FoV, being available two possible approaches: using ray optics in 3D or projective geometry [44].

10.1.2 Alternative approaches

By trying to answer to question of how could it be simulated a LiDAR system in a real case scenario in order to study how the system behaves, it was found out that the game GTA [53] allows to save point clouds of the FoV of the user. Having those point clouds referring to each frame and using computer vision algorithms it can be extracted some useful information.

According to [54], there are two main available simulators: CARLA [55], and GTA V (gets the point-clouds through Script-hook). The goal in using these simulators is the fact that they allow to obtain ideal point clouds (that can also be degraded) to be compared with the one obtained by the proposed LiDAR on this thesis.

10.2 Calibration

10.2.1 Stereo matching

So far it has been considered one or two spots per beam, which makes it trivial to correspond points on a pair of frames. Still, when considering more than two spots, that correspondence is not so trivial: there is the need of finding a correspondence between a point in one frame and its match on the other frame. This problem introduces the concepts of stereo vision and epipolar geometry [44].

In order to perform the correspondence, it is necessary to determinate the fundamental matrix, F [44], which allows to determinate the epipolar line [56], that is, the horizontal line where the pair might be on the other frame. The next steps include optimize this method and build an algorithm that allows to correctly find pairs.

10.2.2 Tilt correction

As seen, even the smallest tilt produces a very severe effect on the distance estimation error. A possible solution would be to have reference points that would be known its distance and correct the centroid's position until obtaining a match. Still, this method would imply the existence of those reference points, which might not be practical, meaning that the priority is find the best and simplest way to correct the cameras tilt.

10.3 Extensive experimental validation

At an initial stage there were only tested experimentally the one and two spots per beam scenarios, that is, frames with only one or two spots at time so that the correspondence between pairs and frames would be trivial, being all experiments made in a dark room.

The next phase of experiments includes testing with a pattern of spots for several (and larger) distances. Considering the limited space and conditions, the initial tests were only until a 5 m range and for several different fields.

10.4 Development of a real-time prototype

Even though the simulations were made in Matlab, Open CV was chosen for the practical implementation because it is an open source program. Along with Open CV, there is Robot Operating System (ROS), which is a set of software libraries and tools that help building robot applications [57]. It provides services designed for a heterogeneous computer cluster such as hardware abstraction, low-level device control, message-passing between processes and package management [58].

If, in the future, it is necessary to create a prototype with a micro-controller that implements the proposed concepts, then it would be implemented with Open CV and ROS.

10.5 Devising suitable/adaptable illumination schemes

Being illumination a key aspect of this setup, it is important to ascertain the most appropriate pattern of illumination and which is the best way to generate that pattern. Concerning illumination, for this specific setup there are two main options to consider:

1. Scan: consists on tilting the light source in order to cover all the viewable area, whether using a DOE, an array of lasers or MEMs [59].
2. DOE: this approach consists on having a diffracting element that creates a matrix of dots.

The biggest advantage of the scanned (rotary) approach, like Velodyne does, is the 360° FoV, allowing to have a complete image of what is surrounding the car, still, it is limited by the mechanical system and its maximum rotation speed. When using DOE, the power is divided, meaning that a more powerful laser is necessary in order to achieve the intended range.

10.6 Automatic calibration

The camera's calibration can be a problem if not done with a very high accuracy. There are several ways to do it but most of them imply the use of a calibration patterns like the used chessboard. As known, that method works very well if there is a sufficient number of photo pairs. The problem comes if it is necessary to calibrate the cameras on-the-fly, that is, continuously calibrating the cameras in order to correct the camera's tilt or any other error factor.

Two approaches can be taken regarding calibration: supervised and unsupervised. The first one consists on having a calibration data-set from data collected at different times that can allow the estimation of the calibration parameters. It usually requires a dedicated calibration target and many hand measurements.

Unsupervised calibration requires no calibration target, no labeling, and no manual measurements. According to [60], it was demonstrated the unsupervised calibration of a custom Velodyne-like sensor, making use of the known spinning motion of the sensor base to automatically calibrate beam angles, relative laser positions, and timing offsets.

10.7 Object ID via intensity analysis

When a spot hits a certain object, depending on the object properties, a different type of reflection occurs: the Lambertian and Specular reflections, as seen on figure 10.1. Lambertian reflection [61] is the property that defines an ideal “matte” or diffusely reflecting surface. The apparent brightness of a Lambertian surface to an observer is the same regardless of the observer's angle of view. More technically, the surface's luminance is isotropic, that is, uniform in all directions.

Specular reflection [62] is a type of surface reflectance often described as a mirror-like reflection of light from the surface. In specular reflection, the incident light is reflected into a single outgoing direction: it occurs for all polished and smooth surfaces, like mirrors.

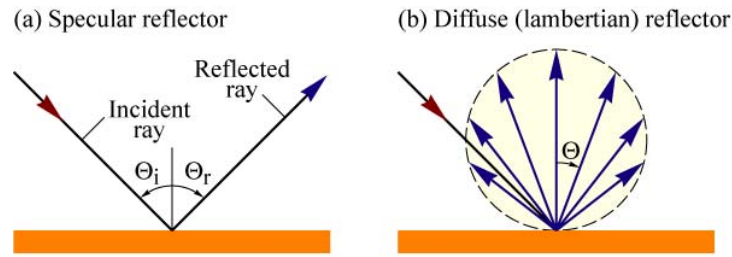


Figure 10.1: Illustration of types of reflection that can occur: specular and diffuse (Lambertian) reflection [63].

According to the shape and material of the illuminated object, the reflected spot will have different intensities. If the intensity on both right and left cameras is approximately the same, then it can be inferred that the reflection is Lambertian; in case the intensities are very different, then the reflection is specular. By analysing the variation of light's intensity on both cameras, this information can be used to distinguish different materials.

Bibliography

- [1] ASIRT. *Road safety facts*. URL: <https://www.asirt.org/safe-travel/road-safety-facts>.
- [2] Jon Walker. *The Self-Driving Car Timeline – Predictions from the Top 11 Global Automakers*. URL: <https://emerj.com/ai-adoption-timelines/self-driving-car-timeline-themselves-top-11-automakers/>.
- [3] Tim Dawkins. *Autonomous Cars 101: What Sensors Are Used in Autonomous Vehicles?* URL: <https://levelfivesupplies.com/sensors-used-in-autonomous-vehicles>.
- [4] Mobileye. *Mobileye*. URL: <https://www.mobileye.com>.
- [5] E.G. Bowen. *Radar Days*. Taylor and Francis, 1998. ISBN: 978-0-7503-0586-0.
- [6] Iitk. *Principle of LiDAR*. URL: http://home.iitk.ac.in/~blohani/LiDAR_Tutorial/Principle/20o/20LiDAR.html.
- [7] Sciencedirect. *Laser safety*. URL: <https://www.sciencedirect.com/topics/engineering/laser-safety>.
- [8] Waymo. *Waymo's Lidar*. URL: <https://waymo.com/lidar/>.
- [9] David Hall. *Velodyne Slashes Lidar Sensor Price in Half*. URL: <https://www.tu-auto.com/velodyne-slashes-lidar-sensor-price-in-half/>.
- [10] Ann Neal. *LiDAR vs. RADAR*. URL: <https://www.sensorsmag.com/components/lidar-vs-radar>.
- [11] Tesla. *Tesla Model S*. URL: <https://www.tesla.com/models/S>.
- [12] Electrek. *A look at Tesla's new Autopilot hardware suite: 8 cameras, 1 radar, ultrasonics new supercomputer*. URL: <https://electrek.co/2016/10/20/tesla-new-autopilot-hardware-suite-camera-nvidia-tesla-vision/>.
- [13] Valeo. *Valeo Scala*. URL: <https://www.valeo.com/en/valeo-scala/>.
- [14] MediaCenter. *Automated driving at a new level: The Audi AI traffic jam pilot*. URL: <https://www.audi-mediacycenter.com/en/techday-piloted-driving-the-traffic-jam-pilot-in-the-new-audi-a8-9276/automated-driving-at-a-new-level-the-audi-ai-traffic-jam-pilot-9283>.
- [15] Kevin Jost. *General Motors readies its Cruise AV for 2019*. URL: <https://www.autonomousvehicletech.com/ext/resources/AVT>.
- [16] Society of Photo-Optical Instrumentation Engineers Yoder P.R. *Mounting optics in optical instruments*. 2nd ed. 2008, p. 239.
- [17] Wikipedia. *Coincidence rangefinder*. URL: https://en.Wikipedia.org/wiki/Coincidence_rangefinder#/media/File:Coincidence_Range_Finder.jpg.

-
- [18] Thomas Wildenberg. *Range finders*. URL: <http://www.ibiblio.org/hyperwar/USN/ref/FireControlInst/FireControlInstsCh6.html>.
- [19] Socratic. *What is a parallax error?* URL: <https://socratic.org/questions/what-is-a-parallax-error>.
- [20] Scienceline. *How can we measure the speed of light?* URL: <http://scienceline.ucsb.edu/getkey.php?key=4790>.
- [21] Researchgate. *Optical Ranging System Employing High Power Injection Laser Diode*. URL: <https://www.researchgate.net/publication/224112492>.
- [22] Bashny. *Time of flight*. URL: <https://bashny.net/t/en/271646>.
- [23] Banggood. *Digital Range finder*. URL: <https://pt.banggood.com>.
- [24] Wikipedia. *Kinect*. URL: <https://en.Wikipedia.org/wiki/Kinect>.
- [25] Wikipedia. *Kinect*. URL: <https://pt.Wikipedia.org/wiki/Kinect/media/KinectSensor.png>.
- [26] Azure. URL: <https://opdhsblobprod03.blob.core.windows.net/contents/>.
- [27] Intel. *Intel RealSense*. URL: <https://www.intel.com/content/www/us/en/architecture-and-technology/realsense-overview.html>.
- [28] Intelrealsense. *Intel RealSense Depth Camera*. URL: <https://www.intelrealsense.com/depth-camera-d435i/>.
- [29] Murray Slovic. *Yole Anticipates 6B Market for Autonomous-Vehicle LiDAR in Five Years*. URL: <https://www.electronicdesign.com/automotive>.
- [30] Velodyne. *Velodyne LiDAR*. URL: <https://www.slideshare.net/BrettJohnson1/velodyne-lidar-comparison-chart>.
- [31] Leddar. *Leddar*. URL: <https://leddartech.com/>.
- [32] LedarTech. *Leddar Vu8*. URL: <https://leddartech.com/lidar>.
- [33] Joaquim dos Santos. *Proof-of-concept of a Single-point Time-of-Flight LiDAR System and Guidelines towards Integrated High-accuracy Timing, Advanced Polarization Sensing and Scanning with a MEMS Micromirror*. 2018, p. 73.
- [34] Robert E Fischer. *Optical System Design (2nd ed.)* 4th ed. Addison-Wesley, 2002, 2008, p. 603. ISBN: 0-07-147248-7.
- [35] CODE V Optical - Design Software. URL: <https://www.synopsys.com/optical-solutions/codev.html>.
- [36] Lambdares. *OSLO*. URL: <https://www.lambdare.com/oslo-usonly/>.
- [37] Eugene Hetch. *Optics*. 4th ed. Addison-Wesley, 2002, 2002, More on geometrical optics.
- [38] OpenStax. *Thin Lenses*. URL: https://phys.libretexts.org/Bookshelves/University_Physics/.
- [39] Britannica. *Parallax - Optics*. URL: <https://www.britannica.com/science/parallax>.
- [40] Tim Dobbert. *Matchmoving: The Invisible Art of Camera Tracking*. 2nd ed. 2012. ISBN: 978-1-118-35205-2.
- [41] David Mendlovic Zeev Zalevsky. *Optical super-resolution*. 2004, ISBN - 0-387-00591-9.
-

-
- [42] *HL6738MG Datasheet*. URL: <https://www.thorlabs.com/drawings/a7484d60bf3a229c-F2AF2B55-0B36-15CF-DA768C8FB359EB05/HL6738MG-MFGSpec.pdf>.
 - [43] Matlab. *What Is Camera Calibration?* URL: <https://www.mathworks.com/help/vision/ug/camera-calibration.html>.
 - [44] R.I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518. 2004.
 - [45] Learn OpenCV. *Find the Center of a Blob*. URL: <https://www.learnopencv.com/find-center-of-blob-centroid-using-opencv-cpp-python/>.
 - [46] OpenCV. *Image Moments*. URL: <https://docs.opencv.org/2.4.13.7/doc/tutorials/imgproc/shapedescriptors/moments/moments.html>.
 - [47] OpenCv. *Feature Detection*. URL: https://docs.opencv.org/2.4/modules/imgproc/doc/feature_detection.html?highlight=houghcircles.
 - [48] Lasersafetyfacts. *Laser Safety Facts*. URL: <https://www.lasersafetyfacts.com/laserclasses.html>.
 - [49] John E. Greivenkamp. *Field Guide to Geometrical Optics*. 2004, p. 29. ISBN: 0-8194-5294-7.
 - [50] A. E. Siegman. *Lasers*. University Science Books, 1986, pp. 664–669. ISBN: 0-935702-11-3.
 - [51] Sciencedirect. *Beam Divergence*. URL: <https://www.sciencedirect.com/topics/engineering/beam-divergence>.
 - [52] Malvin Carl Saleh. *Fundamentals of Photonics*. 80-107, ISBN: 0-471-83965-5. URL: <https://www.slideshare.net/BrettJohnson1/velodyne-lidar-comparison-chart>.
 - [53] RockStar Games. *Grand Theft Auto*. URL: <https://www.rockstargames.com/grandtheftauto>.
 - [54] Jacek Izydorczyk Tomasz Sulkowski. *In Search of the Ultimate Autonomous Driving Simulator*. URL: https://www.researchgate.net/publication/328527984_In_Search_of_the_Ultimate_Autonomous_Driving_Simulator.
 - [55] CARLA. *CARLA Simulator*. URL: <http://carla.org/>.
 - [56] Open CV. *Epipolar Geometry*. URL: https://docs.opencv.org/3.4/da/de9/tutorial_py_epipolar_geometry.html.
 - [57] ROS. *About ROS*. URL: <http://www.ros.org/about-ros>.
 - [58] Jackie Kay. *Robot Operating System*. URL: http://design.ros2.org/articles/realtime_proposal.html.
 - [59] MEMS. *What is MEMS Technology?* URL: <https://www.mems-exchange.org/MEMS/what-is.html>.
 - [60] Paul Newman Mark Sheehan Alastair Harrison. *Self-Calibration for a 3D Laser*. URL: http://www.robots.ox.ac.uk/mobile/Papers/2012IJR_mark.pdf.
 - [61] Renfu Lu. *Light Scattering Technology for Food Property, Quality and Safety Assessment*. 2016. ISBN: 9781482263350.
 - [62] Azooptics. *Specular Reflection - Definition*. URL: <https://www.azooptics.com>.
 - [63] Ecse. *LED*. URL: <https://www.ecse.rpi.edu/schubert/Light-Emitting-Diodes-dot-org>.
-