Fusión sensorial en sistemas de ayuda a la conducción

Sensor fusion in driving assistance systems

UNIVERSIDAD CARLOS III DE MADRID

Aurelio Ponz Vila

Director: Dr. Ing. Prof. José María Armingol Moreno

Tutor: Dr. Ing. Fernando García Fernández

Departamento de Ingeniería de Sistemas y Automática
Universidad Carlos III de Madrid

This dissertation is submitted for the degree of
Doctor en Ingeniería Eléctrica, Electrónica y Automática

Universidad Carlos III de Madrid May 2017

TESIS DOCTORAL

Fusión sensorial en sistemas de ayuda a la conducción

Autor: Aurelio Ponz Vila

Directores:
Dr. Ing. Prof. José María Armingol Moreno
Dr. Ing. Fernando García Fernández

Firma del Tribunal Calificador:

Presidente:

Secretario:

Vocal:

Calificación:

Leganés, de de

Declaration

This thesis is submitted to the Departamento de Ingeniería de Sistemas y Automática of the Escuela Politécnica Superior of the Universidad Carlos III de Madrid, for the degree of Doctor of Philosophy. This thesis is entirely my own work, and, except where otherwise indicated, describes my own research.

Copyright © 2017 Aurelio Ponz Vila.

Aurelio Ponz Vila
May 2017

Acknowledgements

En primer lugar, quiero agradecer a mis directores de tesis, y en particular a Fernando García, su dedicación y generosidad durante estos años, en los que tanto me han ayudado a aprender y crecer.

Agradezco también a mis compañeros de laboratorio (que incluyen a la jefatura) su colaboración cuando fue precisa, por compartir sus conocimientos cuando lo requerí, y por tantas risas y buenos momentos (y también butacas rojas) que nos hacen olvidar que estamos trabajando. Sus opiniones y puntos de vista me enriquecen en todos los órdenes.

Y no quiero olvidar a otras personas de la Universidad que no suelen aparecer en los agradecimientos, y que han colaborado activamente para que mis descabellados prototipos no ardan ni se desmonten. Sin ningún orden en particular, gracias, Ángela Nombela, Fernando Sandeogracias, Jose Antonio Campo, Carlos Fernández y Fernando Serrano. Provoco envidia allá donde voy por poder contar con vosotros.

Y a las piedras del camino con las que tropecé, porque las victorias saben mejor cuanto más cuestan.

Resumen

La vida diaria en los países desarrollados y en vías de desarrollo depende en gran medida del transporte urbano y en carretera. Esta actividad supone un coste importante para sus usuarios activos y pasivos en términos de polución y accidentes, muy habitualmente debidos al factor humano. Los nuevos desarrollos en seguridad y asistencia a la conducción, llamados Advanced Driving Assistance Systems (ADAS), buscan mejorar la seguridad en el transporte, y a medio plazo, llegar a la conducción autónoma.

Los ADAS, al igual que la conducción humana, están basados en sensores que proporcionan información acerca del entorno, y la fiabilidad de los sensores es crucial para las aplicaciones ADAS al igual que las capacidades sensoriales lo son para la conducción humana. Una de las formas de aumentar la fiabilidad de los sensores es el uso de la Fusión Sensorial, desarrollando nuevas estrategias para el modelado del entorno de conducción gracias al uso de diversos sensores, y obteniendo una información mejorada a partid de los datos disponibles.

La presente tesis pretende ofrecer una solución novedosa para la detección y clasificación de obstáculos en aplicaciones de automoción, usando fusión

sensorial con dos sensores ampliamente disponibles en el mercado: la cámara de espectro visible y el escáner láser. Cámaras y láseres son sensores comúnmente usados en la literatura científica, cada vez más accesibles y listos para ser empleados en aplicaciones reales. La solución propuesta permite la detección y clasificación de algunos de los obstáculos comúnmente presentes en la vía, como son ciclistas y peatones.

En esta tesis se han explorado novedosos enfoques para la detección y clasificación, desde la clasificación empleando clusters de nubes de puntos obtenidas desde el escáner láser, hasta las técnicas de domain adaptation para la creación de bases de datos de imágenes sintéticas, pasando por la extracción inteligente de clusters y la detección y eliminación del suelo en nubes de puntos.

Abstract

Life in developed and developing countries is highly dependent on road and urban motor transport. This activity involves a high cost for its active and passive users in terms of pollution and accidents, which are largely attributable to the human factor. New developments in safety and driving assistance, called Advanced Driving Assistance Systems (ADAS), are intended to improve security in transportation, and, in the mid term, lead to autonomous driving.

ADAS, like the human driving, are based on sensors, which provide information about the environment, and sensors' reliability is crucial for ADAS applications in the same way the sensing abilities are crucial for human driving. One of the ways to improve reliability for sensors is the use of Sensor Fusion, developing novel strategies for environment modeling with the help of several sensors and obtaining an enhanced information from the combination of the available data.

The present thesis is intended to offer a novel solution for obstacle detection and classification in automotive applications using sensor fusion with two highly available sensors in the market: visible spectrum camera and laser scanner. Cameras and lasers are commonly used sensors in the scientific literature, increasingly affordable and ready to be deployed in real world applications. The solution proposed provides obstacle detection and classification for some obstacles commonly present in the road, such as pedestrians

and bicycles.

Novel approaches for detection and classification have been explored in this thesis, from point cloud clustering classification for laser scanner, to domain adaptation techniques for synthetic dataset creation, and including intelligent clustering extraction and ground detection and removal from point clouds.

Table of contents

List of figures

List of tables

Nomenclature

Acronyms / Abbreviations

ADAS Advanced Driving Assistance System

AGM Absorbed Glass Mat

DM Disparity Map

DPM Deformable Part Model

FIR Far Infra Red

HOG Histogram of Oriented Gradients

iCab Intelligent Campus Automobile

IMU Inertial Measurement Unit

INRIA Institut National de Recherche en Informatique et en Automatique

ISL Intelligent Systems Lab

IVVI Intelligent Vehicle based on Visual Information

JDL Joint Directors of Laboratories

LBP Local Binary Pattern

LRR Long Range Radar

LSI Laboratorio de Sistemas Informáticos

MSAC M-estimator-SAmple-Consensus

NHTSA National Highway Traffic Safety Administration

OOI Object Of Interest

PERCLOS Percentage of Eye Closure

RADAR Radio Detection and Ranging

RF Random Forest

ROI Region Of Interest

SIMBA Sistema Integrado de Monitorización Bidireccional del Automóvil

SLAM Simultaneous Localization and Mapping

SRR Short Range Radar

SUV Sport Utility Vehicle

SVM Support Vector Machines

TCP/IP Transmission Control Protocol / Internet Protocol

TESLA Information treatment device installed in the IVVI 2.0 platform where the ADAS algorithms are run

TOF Time Of Flight

UC3M Universidad Carlos III de Madrid

Chapter 1

Introduction

Today's lifestyle depends heavily on automobile transport for passengers and goods. The high cost of this activity, both in impact on active and passive users' health and in environment pollution, makes the investment in transport security and economy inexcusable and highly socially and economically profitable.

Intensive automobile transport involves an important overhead in the form of accidents, largely attributable to the human factor. Advanced Driving Assistance System (ADAS) can reduce significantly both frequency and severity of the accidents, assisting the driver in the most risky maneuvers, such as two-way road overtaking, road access and other vehicles and pedestrian crossing.

Up to date ADAS applications are already providing semi-autonomous driving in well maintained highways and highly structured environments,reducing danger of distraction inherent to long distance journeys. Less structured and controlled environments, such as urban streets or interurban two-ways roads, offer greater difficulty for autonomous driving, together with a higher accident rate for human driving.

The most vulnerable actors in traffic are pedestrians and cyclists, because of the lack of vehicle protection around the person, and the great difference in speed with respect to other actors, such as motorbikes or cars. Table 1

displays statistics on pedestrians killed in traffic accidents in Europe 2013, with a helping figure 1. These data shows that mortality rate for pedestrians is very high when involved in road accidents, taking into account that pedestrians are very rarely implicated in them. This high vulnerability for pedestrians and cyclists inspires the special attention of this thesis to the accurate detection and classification of these traffic actors.

Table 1 Data of persons killed in road accidents in Europe, by road user, 2013 [1]

	Driver	Passenger	Pedestrian
Belgium	71,4	14,8	13,7
Czech Republic	59,5	15,7	24,8
Denmark	63,4	19,4	17,3
Germany	69,2	14,0	16,8
Ireland (1)	62,3	19,8	17,9
Greece (1)	65,9	16,9	17,2
Spain	60,6	17,3	22,1
France (1)	69,1	17,5	13,4
Croatia	58,4	22,8	18,8
Italy	67,9	15,9	16,2
Cyprus	65,9	15,9	18,2
Latvia	46,4	14,5	39,1
Luxembourg	77,8	11,1	11,1
Hungary	55,0	20,1	24,9
Netherlands	80,3	9,0	10,7
Austria (1)	70,4	14,3	15,3
Poland (1)	49,2	18,4	32,4
Portugal	60,8	16,6	22,6
Romania	38,7	22,3	39,0
Slovenia (1)	70,8	14,6	14,6
Finland	70,2	16,7	13,2
Sweden	65,8	16,2	16,2
United Kingdom	60,9	16,2	22,9
Iceland	66,7	26,7	6,7
Norway	72,7	17,6	9,6
Switzerland	63,6	10,8	25,7
Average	64,0	16,7	19,2

(1) 2012 data.
Source: Eurostat/CARE (online data code: tran_sf_roadus)

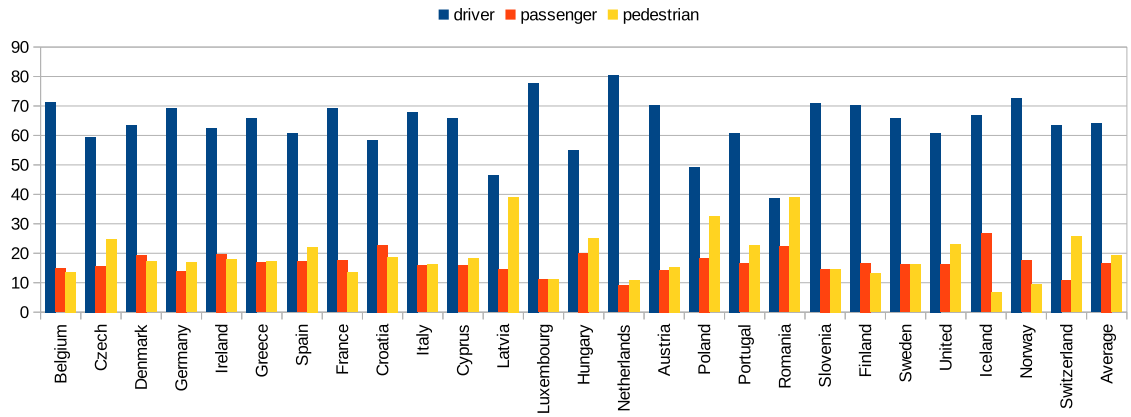


Fig. 1 Persons killed in road accidents, by road user in Europe, 2013 [1]

Table 2 shows statistics on pedalcyclists fatalities versus total fatalities in traffic crashes. As the number of bicyclists is growing rapidly in the USA [2], accurate and reliable bicyclists detection and classification is a major concern for ADAS researchers. Bicycle trips in USA increased from 1.700 million trips in 2001 to 4.000 million trips in 2009, this is the reason why pedalcyclist fatalities are rising since 2009 while total fatalities are descending, as explained graphically in figure 2.

Table 2 Total Fatalities and Pedalcyclist Fatalities in Traffic Crashes, USA 2005–2014 [3]

Year	Total Fatalities	Pedalcyclist Fatalities	Percentage of Total Fatalities
2005	43,510	786	1.8%
2006	42,708	772	1.8%
2007	41,259	701	1.7%
2008	37,423	718	1.9%
2009	33,883	628	1.9%
2010	32,999	623	1.9%
2011	32,479	682	2.1%
2012	33,782	734	2.2%
2013	32,894	749	2.3%
2014	32,675	726	2.2%

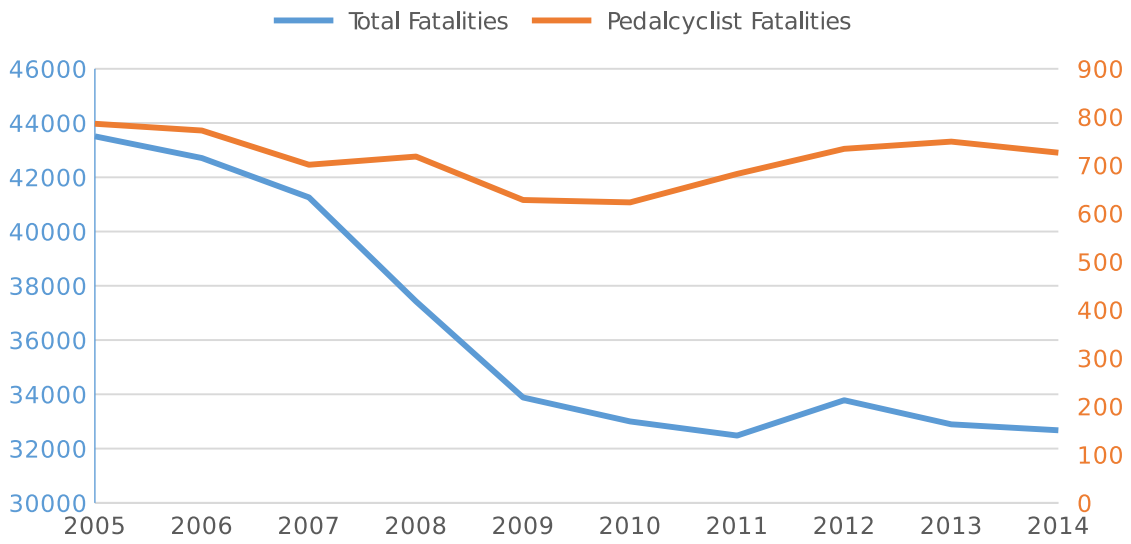


Fig. 2 Total Fatalities and Pedalcyclist Fatalities in Traffic Crashes, USA 2005–2014 [3]. Left Y scale is for total fatalities, right Y scale is for pedalcyclist fatalities

It is indisputable that the transportation activity faces a future of autonomous driving. Nevertheless, an initial scenario of shared road between unautomated fully human controlled, partially automated, and fully automated vehicles will occur. During this transitional period, ADAS will be increasingly common as an starting point towards fully automated vehicles.

Autonomous driving will lead to crucial advances in transport economy, so radical changes in the conception of the transportation means are expected. The ownership of a car, an hegemonic concept nowadays, will give rise to new systems of vehicle sharing, or pay-per-service, reducing significantly the fixed costs of acquisition and maintenance of vehicles, whose current individual usage is evidently uneconomic.

The deployment of complex infrastructures with automatic communication infrastructure-vehicle and vehicle-infrastructure will involve major improvements in transport efficiency. A change of paradigm, from the current human competitive driving towards a more effective collaborative autonomous driving, will optimize traffic flows, predict and optimize duration and cost of the paths, and reduce significantly accidents and incidents.

National Highway Traffic Safety Administration (NHTSA) defines automated vehicles as "*those in which at least some aspects of a safety-critical control function (e.g., steering, throttle, or braking) occur without direct driver input*" , and differentiates five levels of automation in vehicles [4]:

- Level 0 – No-Automation. The driver has total control of all the primary vehicle controls (steering, throttle, or braking and motive power) at all times.
- Level 1 – Function-specific Automation. At this level, automation involves at least one specific control functions, such as stability control, cruise control, lane keeping or brake assist. Should more than one function is automated, they operate independently from each other. The vehicle may have multiple capabilities combining individual driver support and crash avoidance technologies, but does not replace driver vigilance and does not assume driving responsibility from the driver.
- Level 2 - Combined Function Automation.

  This level involves automation of at least two primary control functions designed to work in unison to relieve the driver of control of those functions. The driver keeps responsibility for monitoring the road at all times and on short notice. An example of this level are adaptive cruise control combined with lane centering.
- Level 3, - Limited Self-Driving Automation. Vehicles at this level of automation enable the driver to cede full control of all safety-critical functions under certain traffic or environmental conditions and in those conditions to rely heavily on the vehicle to monitor for changes in those conditions requiring transition back to driver control. The driver can be required to take control of the vehicle, but with sufficient transition time, e.g. in case of oncoming construction area, when automation is not supported.

- Level 4, Full Self-Driving Automation. The vehicle is designed to perform all safety-critical driving functions and monitor roadway conditions for an entire trip. The driver just provides destination and is not expected to be able to take control of the vehicle at any time.

Some authors are surprised by several parts of the definition, as this preliminary statement does not mention that the crash avoidance and self-driving technologies are actually aimed at ameliorating driver shortcomings, whether from inattention or inability. Rather, it stresses that the driver remain vigilant and be "prepared to take immediate control" even though it is well known that more than 90% of crashes involve human error. Author's opinion is that drivers will be distracted and will not be able to take control immediately [5]. The foreseeable future of autonomous driving NHSTA level 3 and 4 will not be possible without the current evolution in Information Technologies and communications, that will allow the development of increasingly powerful computers and precise sensors in order to support complex multisensor systems for a reliable and accurate acquisition of the surrounding reality.

There are already several car makers offering commercially available level 2 and level 3 automated cars. Figure 3 shows the current state of the art on automated vehicles and the announced plans for the future [6]. As we can see, Google is the only developer as of 2016 with a level 4 car, though it is just an experimental model. One of the advertised testers of the Google car is a blind person, to remark the fully automated level 4 capabilities of the vehicle.
The former is just part of the programs in process for ADAS and autonomous cars research. Car manufacturers, software and hardware companies and universities are developing ADAS systems and prototypes of autonomous cars: Stanford University, Free University of Berlin, University of Parma, Griffith University in Australia, Oxford University, Massachusetts Institute of

Technology and many other companies, as well as all the car makers, whose survival in the near future will depend on its ability to make autonomous cars.

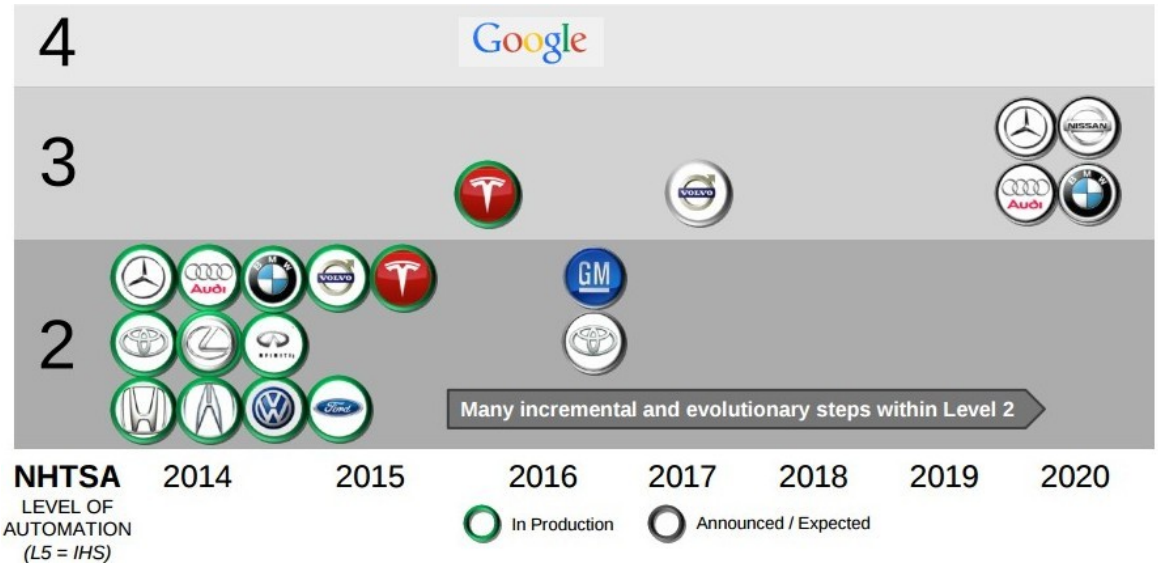


Fig. 3 Current state of the art and announced plans for automated vehicles as of June 2015 [6]

Tesla Motors is currently selling several models complying level 3 requirements like the S P85D including the Autopilot feature with good practical results [7] using cameras, ultrasounds and radar as sensors (figure 4a). The Mercedes Benz S65 AMG is a level 2 of automation vehicle (figure 4b) and is offering the Distronic Plus system with steering assist, adaptive brake technology and active lane keeping assist using one stereo camera and five radar sensors. Infiniti sells the level 2 vehicle Q50S (Figure 4c) using one camera and one radar sensor, supporting Intelligent cruise control, predictive forward collision warning, lane departure warning and prevention and active lane control. BMW is offering also a level 2 vehicle (Figure 4d), the 2016 750i xDrive with active driver assistance plus as ADAS, provided by one stereo camera and five radar sensors.

(a) Tesla Model S

(b) Mercedes-Benz S65 AMG

(c) Infiniti Q50S

(d) BMW 750i

Fig. 4 Some of the state of the art commercial vehicles for sale as of 2016 [7]

The only NHTSA level 4 vehicle advertised as of 2016 driving in public roads is the Google Self-Driving Car Project [8]. Google is offering monthly reports on statistics about the Google car experience [9] with examples about concrete situations, how they addressed it and lessons learned. Information and experiences obtained from Google data can be very useful for ADAS researchers and developers.

Google started his project using modifications of commercially available cars, like Lexus SUVs, but finally developed his own vehicle designed from the scratch to meet the technical and aesthetic requirements for the project, as seen in figure 5. The external shape of the vehicle is determined by the multilayer laser located in the top, that needs to be elevated from the edges of the roof to be able to detect obstacles near the vehicle.

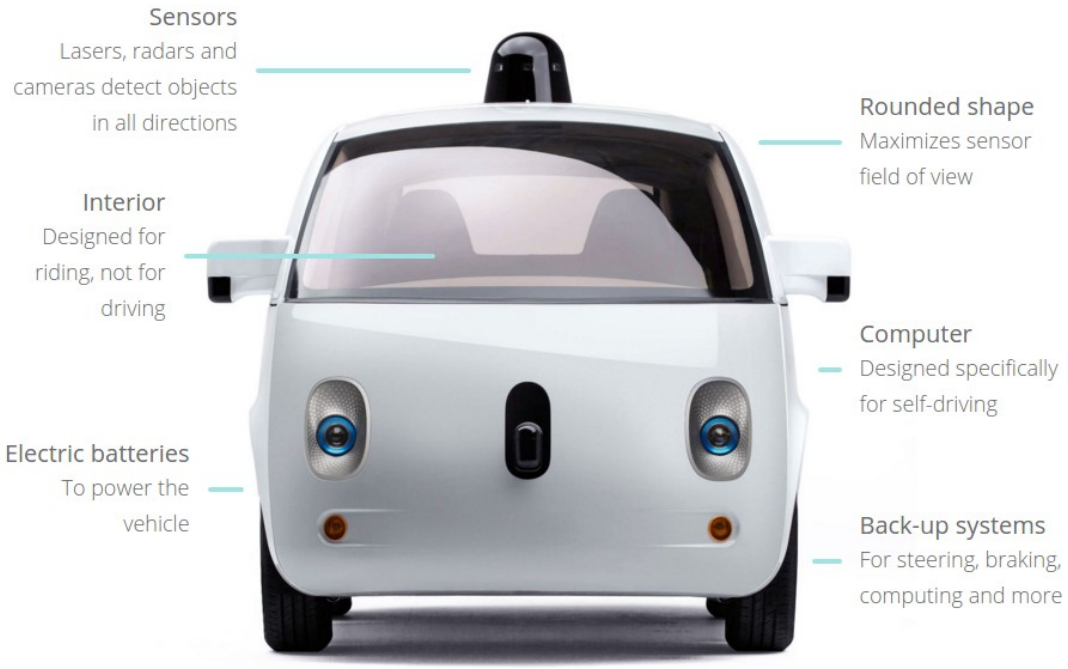


Fig. 5 Google Self-Driving Car Project [8]

A representation of the perception of the reality that the Google car extracts from the sensors is shown in figure 6 with a pre-mapped representation of the road and dynamically detected moving obstacles such as cars, vans, pedestrian from lasers, radars and cameras.

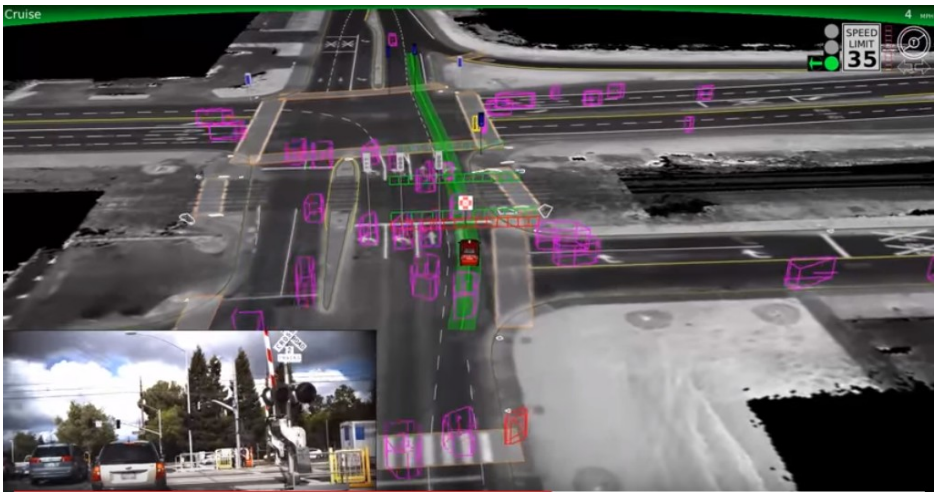


Fig. 6 Representation of Google Car perception [8]

As we have seen, ADAS are based on the perception extracted from the reality surrounding the vehicle, so the sensors used are crucial for the reliability of the whole system. When considering human driving, we are aware of the enormous constraints in the driver's ability for reality perception and management and execution of the driving task. Humans are extremely fallible subjects, and we accept socially that reality. Nevertheless, we are also extremely demanding with ADAS, and specially with autonomous vehicles. As could not be otherwise, we aim to reach perfection in ADAS, and sensors and data processing are crucial for that purpose. The intention of the present thesis is to make progress in the research of methods and algorithms to move closer to that objective.

1.1 Sensor Fusion

Human sensing system and the treatment performed of the available information are extremely complex. Human driving ability is the result of years of training in perception, reality interpretation, inference and educated forecast. The adaptation of these senses and the human system for information processing demands complex sensing systems not only meeting human sensing abilities, but extending them with new capabilities beyond human nature.
ADAS systems are highly demanding in terms of reliability and availability, so a coordinated combination of different sensors is needed, increasing their strengths and compensating their weaknesses. Along this thesis, two complementary sensors have been used: laser scanner and camera for computer vision.
Laser scanner provides reliable and robust perception of the environment in virtually any weather and illumination condition, although the information provided is limited. Computer vision provides very rich information, but it is vulnerable to poor illumination conditions such as darkness, high brightness,

tunnel entrance or exit and fast camera movement as in roundabout turns. The conjunction of both sensors supplies much more than the mere addition of the individual sensor capabilities, providing accurate and reliable obstacle detection and classification.

1.2 Proposal

The present thesis intends to demonstrate the enhanced ability for obstacle detection and classification of the sensor fusion and the novel classification algorithms for laser scanner data. These systems will be deployed and tested in the research platform IVVI 2.0 [10] from the Intelligent Systems Lab (LSI) as seen on figure 7 and takes advantage of the experience obtained from previous researches [10–13] from the LSI.
System capabilities for detection and classification will be studied for several driving actors, such as motorbikes, bicycles, pedestrians and cars, also in poor illumination conditions, when laser scanner detection and classification will play a leading role. Classification using laser scanner data will be trained using multiple positive and negative samples for point clouds from all the aforementioned actors, extracting from them the distinctive characteristics. Databases extracted from sensors in the IVVI 2.0 platform will be used.
The classification of the actors will also be performed using computer vision techniques, training the system with both real world and synthetic images, testing the classification performance of these type of training methods. Every database extracted for this purpose will be public for scientific community usage.
Several phases are fulfilled in this thesis:

- Research in laser scanner point cloud representation, segmentation and intelligent clustering extraction. Laser scanner is widely used in ADAS applications for obstacle detection due its reliability and accuracy in

nearly every condition. As new and more affordable laser scanners reach the market, the use in ADAS of this type of devices will expand for obstacle detection and classification as in [14], fusing information with camera and semantic information as in [15], and for vehicle navigation, being specially useful in rural areas and unmarked roads, both urban and interurban, where cameras offer less information [16].

- Research in road plane detection and automatic and unattended extrinsic parameters computation between camera and laser. Data alignment is crucial in sensor fusion, and an unattended and fast method for extrinsic parameters estimation between the sensors involved is needed for real world operation. The work [13] explains a reliable, accurate and unattended method for data alignment.

- Laser scanner point cloud database generation for pedestrians and other actors. This thesis intends to improve system's classification capabilities in poor illumination conditions, through point cloud classification. Point cloud based obstacle classification algorithms are very sensor dependent, as the extracted information from point clouds is scarce. Datasets of pedestrians and other commonly found obstacles are needed for classification research.

- Laser scanner point cloud feature definition for Support Vector Machines training and classification. Some type of features are standard in computer vision classification algorithms, but laser scanner point cloud classification needs a definition of the features defining the different kinds of obstacles. The works [14] and [17] propose diverse approaches to data extraction from point clouds for SVM classification.

- Real world images recompilation for computer vision classification training of all the authors to be classified. Although public images datasets exist for computer vision classifier training, better results are

obtained when the same set of sensors and in the same conditions is used for dataset generation and for data classification. Datasets for pedestrians, cars, motorbikes and bicycles are compiled using the IVVI 2.0 research platform.

- Croma pedestrian images recompilation, treatment and labeling for synthetic pedestrian classification. Some novel techniques for obstacle classification make use of synthetic samples from computer generated worlds, applying domain adaptation algorithms as in [18]. A new perspective of this techniques is used in the presented thesis.
- Pedestrian classifier training, using both synthetic and real samples. A study and statistic performance comparative of the data available for classifier training is important in order to select the most effective combination of real and synthetic datasets.
- Research in fusion of data from computer vision classification and laser point cloud classification. Information fusion from the different sources as in [19] is key for a sensor fusion system in order to obtain the most accurate information about the sensed environment.
- Testing and results generation.

1.3 Document structure

A brief outline of the document structure is presented next.

Chapter 2 will introduce a complete description of the state of the art on the topics related to the present thesis. After a brief introduction, generic sensor fusion and ADAS focused sensor fusion are presented, as they are the core of the system. Then, an overview of sensors technology in ADAS as a source of

the data to be fused.

Chapter 3 will introduce a general description of the work, with the IVVI 2.0 research platform, the sensors involved, and the information processing equipment, both physical and logical.

Chapter 4 is devoted to obstacle detection and classification using laser scanner. Point cloud clustering is presented for obstacle detection, and an explanation on obstacle classification is included, along with results.

Chapter 5 focuses on obstacle classification using computer vision. LSI datasets are explained, as well as training and classification techniques, and results are provided.

Chapter 6 presents the sensor fusion, including a data alignment section and results.

Chapter 7 presents the conclusions of the thesis, including contributions and future works.

Chapter 2

State of the art

2.1 Introduction

Reliable obstacle detection and classification in a driving environment is a key matter for ADAS and autonomous driving. One sensor alone might not be able to supply all the information needed for driving assistance in any weather and illumination conditions, keeping also a redundant, reliable and high quality information. This goal requires the use of different sensors and the fusion of the information acquired by all of them. The most commonly used sensors in ADAS applications are visible spectrum cameras, infrared cameras, laser scanner, sonar and ultrasonic sensors, radar, Global Positioning System (GPS) receivers, and Inertial Measurement Unit (IMU).

The fusion of the information coming from visible spectrum cameras and laser scanner is a commonly used approach in scientific papers. Although numerous datasets exist for classifier generation and testing, such as Karlsruhe Institute of Technology (KIT) databases [20, 21], the practice suggests that best results are achieved if the same sensors and in the same conditions are used in the dataset generation and in the real classification testing and real operation. For this reason, several pedestrian, bicycle, motorbike and car training datasets have been collected for our Intelligent Vehicle based on Visual Information (IVVI) 2.0 research platform, exploring even the generation of synthetic

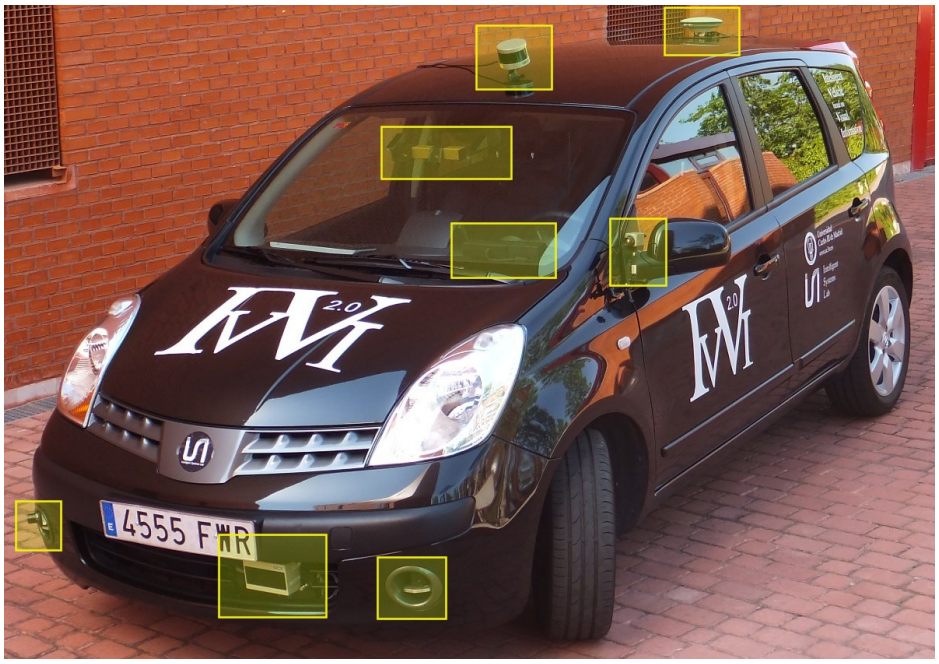
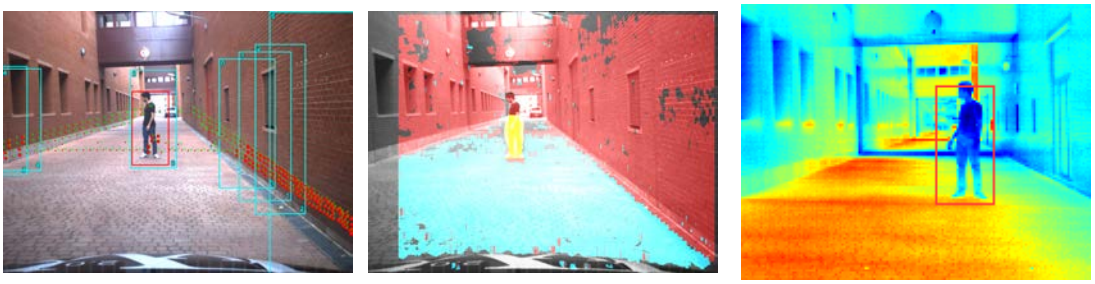


Fig. 7 Research platform IVVI 2.0 and its sensors [10]

datasets using the croma technology with a subsequent addition of textures and real world backgrounds. Figure 7 shows the IVVI 2.0 platform including the installed sensors, while figure 8 displays an example of some of the IVVI 2.0 abilities: Obstacle detection and classification using sensor fusion between laser scanner and computer vision in figure 8a, the use of computer vision for free space detection and obstacle detection and classification in figure 8b, and thermal camera use for pedestrian detection in 8c.



(a) Laser scanner and computer vision fusion obstacle detection and classification

(b) Obstacle and free space detection using computer vision

(c) Pedestrian detection using infrared camera

Fig. 8 Demonstration of some of the abilities of the IVVI 2.0 research platform

2.2 Sensor Fusion

Daily life of the animals in nature and their struggle for subsistence are based on the conjunction of all of their senses. Even those missing some of the senses trust the combination of the rest to resolve this shortcoming, and have evolved for survival. Similarly, vehicle driving is a daily activity for many humans, who also use several senses during the driving. Evolution has enabled us for automatic and unconscious fusion of the information received through our senses, in such a way that the sum of all those inputs supply more value than each of them individually. In a similar way, multisensor systems have to fuse information received from the sensors and manage it so the fusion process supply an added value to the inputs.
Sensor fusion is not a novel concept. Born in the military research, where it has become a key element in defense and intelligence, it has also been successfully adapted to multiple fields in civil technology. The most relevant civil uses for sensor fusion are robotics, industrial process automation, intelligent buildings and medical applications [22].

There are many definitions and multiple points of view for sensor fusion, as it is a multidisciplinary research, including fields such as statistics, signal processing, information theory, artificial intelligence, etc.
The definition offered by the Joint Directors of Laboratories (JDL) is "*A process dealing with the association, correlation, and combination of data and information from single and multiple sources to achieve refined position and identity estimates, and complete and timely assessments of situations and threats, and their significance. The process is characterized by continuous refinements of its estimates and assessments, and the evaluation of the need for additional sources, or modification of the process itself, to achieve improved results.*"[19].

This definition has been extended later by some authors. In [22] the following modifications are proposed:

- Avoid the *"correlation"* term, as it is considered "*merely one method for generating and evaluating hypothesized associations among data*".
- Not considering *"association"* as an essential component for data combination.
- Removing *"single or multiple sources"* from the definition.
- Extending the reference to*"position and identity estimates"* in order to include all the varieties of the state estimation.
- Pointing that not all of the applications require *"complete assessment"* and that *"timely"* is superfluous.
- Extending the definition by avoiding *"Threat assessment"*, because several situations exist where the threat is merely a factor. "*In general, data fusion involves refining and predicting the states of entities and aggregates of entities and their relation to one's own mission plans and goals. Cost assessments can include variables such as the probability of surviving an estimated threat situation*"
- Considering the second phase of the definition as simply illustrative, as not all the information combination processes require process refinement.

Other authors proposed the definition: *"Information fusion is the study of efficient methods for automatically or semi-automatically transforming information from different sources and different points in time into a representation that provides effective support for human or automated decision making*" [23].

After seeing the sensor fusion definition, let us focus on its advantages and goals.
The reasons for using sensor fusion against the individual consideration of sensors are multiple. One unique sensor suffers important drawbacks, such as a limited coverage in space or time, and lack of precision or security.
In contrast, sensor fusion offers important advantages, such as superior robustness and reliability, expanded spacial and temporal coverage, better trustability, ambiguity and uncertainty reduction, higher robustness against interference and higher resolution, as argued in [24].
Despite the numerous advantages of sensor fusion, it presents several limitations. Some authors, as in [25], consider that low quality data fusion does not represent any advantage, but produces delays and wrong decisions using a more expensive equipment.

2.2.1 Sensor fusion architectures

Sensor fusion is a very wide topic, treated by multiple disciplines. For this reason, definitions and categorizations are very diverse, depending on the source. This section will detail some of the concepts and divisions related with sensor fusion.
Several types of sensor fusion exists, depending on the concept of the fusion used for the classification, as explained in [24].

Sensor fusion categorization according to the level of abstraction

One of the ways for sensor fusion categorization is depending on the level of abstraction for the fusion, as in [19] and [22].

- Low level sensor fusion

Also called direct fusion or pixel-level fusion, it combines unprocessed data from different sources in order to create a more complex dataset [26], in principle of more quality than the individual inputs. This sensor fusion is dependent on the particular sensors used. An example of this type of fusion are stereoscopic cameras, in which two sensors (monocular cameras) are fused in order to obtain tridimensional information from bidimensional information using the adequate algorithms. This level involves the greatest computational cost, and provides the highest potential detection performance [22].

- Medium level sensor fusion

  Also known as characteristics level fusion or feature level fusion, it combines edges, corners, lines, textures or positions [26] in a characteristics map, ready of use in segmentation and detection [19]. These characteristics are extracted for each individual sensor, combining them later by means of neural networks, state vectors, etc, in a common decision space.

  As it is a intermediate level fusion, information from several sensors can be used, and advantage of the possibilities of each of the different sensors can be taken, but detections are presumed to be independent for each sensor. Nevertheless, as pointed in [19], the usual training process in these cases makes the addition of new sensors more difficult, as a new training process with the new characteristics from the new sensors is needed.

- High level sensor fusion

  Also called decision level fusion, it combines decisions from the different experts involved in the system. This fusion uses voting systems,

fuzzy logic and statistical methods. The final decision is made as a function of the decisions of each of the sensors and its reliability. This type of fusion is less complex, as it is based on previously established subsystems. Two different methods can be used for making classification decisions: hard decisions, that is, the optimum choice, and soft decisions, allowing some level of uncertainty that can be combined in subsequent stages of the fusion process, as in the work [22]. This level of fusion uses generally Bayesian methods [19] [26]

In this case, the purpose of the fusion process is to add reliability to the detections coming from the aforementioned subsystems, obtaining a final combination of these information. An important advantage of this kind of fusion is its scalability, as the addition of new sensors increases the whole system confidence, usually with no complexity addition. This level requires the lowest computational cost of the three levels.

Sensor fusion categorization according to the sensor configuration

Some authors categorize the sensor fusion depending on the type of configuration of the sensors involved in the fusion [27] [26], considering the diverse, non excluding possibilities, that can even be found in a hybrid way:

- Complementary sensors

  In this case, the sensors are not dependent from each other, but they complement them in order to offer a more complete information of the observed phenomenon. An example is the case of several cameras focusing on disjoint zones in an operations theater [26]. The information fusion coming from complementary sensors is simple, as the new information is just added to the preexistent.

- Competitive sensors

  Competitive sensors are those supplying independent measures of the same object, providing fault tolerance and robustness to the system. This is the case of the so called fault tolerance systems, where the compliance of the standards of service must be ensured even in the case of failure. Alternatively, and as an inferior security level system, competitive sensors allow to offer a degraded behavior in case of failure, adding robustness to the system [27].

- Cooperative sensors

  In this case, the fusion uses information from independent sensors in order to obtain results that would not be available in the case of single sensor use, such as in stereoscopic vision. This type of results are the less certain and more difficult to obtain, as they depend directly on the proper functioning of all of the sensors involved. Unlike competitive sensors, these sensors reduce precision and reliability [27].

These three categories of sensor configuration are not mutually exclusive, as diverse hybrid architectures exist, for example where multiple cameras can cover a common area in a competitive or cooperative way, while configuration would be complementary in the areas covered by just one camera.

Sensor fusion categorization according to the point of decision

Topology is a key characteristic in sensor fusion systems, that is, the way how the different nodes communicate and it function in the results delivery. In line with these criteria, some works as in [22], [28] and [29] propose the division of sensor fusion systems in centralized and distributed systems.

- Centralized systems

  In this case, all the nodes in the system send the information to a central node, where the final fusion is performed. It is important to note that some aspects of the intermediate sensor fusion may have been performed in a cooperative way in some of the nodes, as is the case when one of the sensors is a stereo camera. In centralized systems, the central node concentrates all the information from all of the sensors, providing reliability to the system, while the fault tolerance of the system is limited. An example of this type of systems if shown in [30], where a central node in a vehicle receives information from a differential GPS and inertial sensors, fusing the entire set of information with road maps, in order to obtain a more precise position than the one available using just a differential GPS.

- Distributed systems

  Systems where each node performs the fusion in a local way using information from the same node and, in some cases, adjacent nodes, are called distributed systems. A differential GPS fusing by itself information from its own sensor and the differential system is an example of distributed system. When fusing also information from inertial sensors, it would be considered a distributed multisensor system. These are fault tolerant and easily scalable systems, but the lack of global information reduces the effectiveness of the sensor fusion performed. An example of this strategy can be found in [31].

Figure 9 shows a centralized fusion of the vehicle information, with a central node in charge of the final fusion, opposed to a distributed fusion with local node fusion.

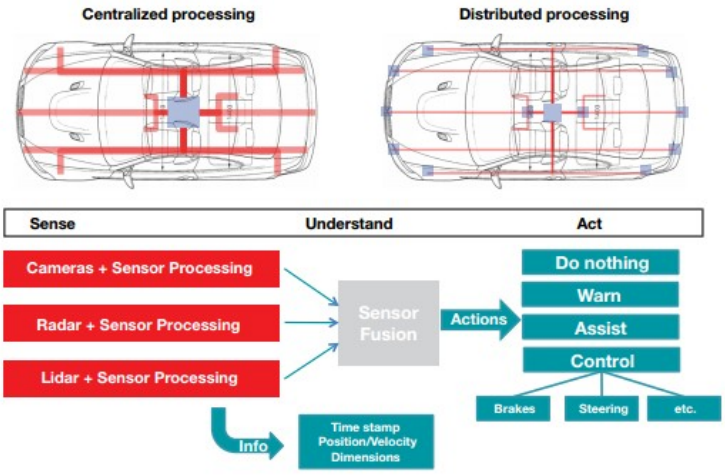


Fig. 9 Centralized vs distributed sensor fusion [32]

2.2.2 Information fusion in ADAS

The application of the aforementioned fusion architectures to ADAS systems is studied next.

Fusion in ADAS considering the level of abstraction

- Low level sensor fusion in ADAS
  Low level fusion intends to elaborate a set of information as a composition of several information set from different sources.

  A direct application of this type of sensor fusion to ADAS systems are stereo cameras, which obtain a disparity map through the independent images of two coordinated monocular cameras. The disparity map indicates the estimated distance from the sensor to every pixel in the image.

  Another example of low level sensor fusion in ADAS is shown in [30], where a fusion of inertial sensors with differential GPS is performed. The differential GPS also fuses its own information with the correction

information received from the differential system.

- Medium level sensor fusion in ADAS
  Medium level sensor fusion implies the processing of information originated by a sensor in order to extract new information. In the work [33], an obstacle detection is performed through a point cloud obtained from a laser scanner. The new information includes new characteristics, such as a list of individual obstacles, their dimensions, distances and distinctive characteristics.

- High level sensor fusion in ADAS
  High level sensor fusion combines in a final phase information obtained in every sensor in an independent way. An example of this kind of fusion applied to ADAS systems is explained in [12], where the system obtains information from a laser scanner and performs a high level fusion with the information extracted from the point cloud about obstacles in the scene and the classification of that obstacles using the 3D point cloud coordinates adaptation to the space of the image.

Fusion in ADAS considering the configuration of the sensors

- Complementary sensors in ADAS applications

  Complementary sensors are independent to each other, but the information supplied can complete the observation of the phenomenon. Lateral cameras in a vehicles are a case of complementary sensors in ADAS. These cameras present disjoint information between them, improving the perception of the scene. The addition of new cameras, laterally or backwards, increases the available information but does not imply an important increase of the system complexity.

- Competitive sensors in ADAS applications

  Sensors working in a competitive manner supply information about the same object, the same way a camera and a laser scanner facing the road perceive the same reality, but in a different way. It can provide fault tolerance if it is the same type of sensors, such as two cameras or two laser scanners, or simply increase the robustness with respect to the monosensor system, if some of them allows the system work in a degraded mode in the case of failure. The system described in en [33] shows a laser scanner and a camera working as competitive sensors, supplying information about the same reality. The failure of one of them allows a degraded mode in the system, which is still capable of sensing the environment, just with less quality than in the correct working mode of all of the sensors.

- Cooperative sensors in ADAS applications

  Cooperative sensors supply results that are not available before the information fusion. The most extended case in ADAS is the stereo camera, in which two monocular cameras with well known characteristics supply a disparity map with information about the distance from the lenses to the reality represented by every pixel in the image. This same example of cooperative sensor would be perceived as a fault tolerant competitive system if just considering the independent images of each of the monocular cameras present in a stereo pair.

Fusion in ADAS considering the point of decision

- Fusion in centralized systems for ADAS

  In this type of systems there is a central node that executes a final process of fusion of the information supplied by the sensors or intermediate fusion processes. The integration of GPS receivers, inertial sensors and

maps is shown in the work [30], where a fusion in a centralized way information from each of the sensors is performed, including information from systems external to the vehicle, such as the differential GPS correction information.

- Fusion in distributed systems for ADAS

  In distributed systems, the nodes perform the sensor fusion in a local way, supplying more elaborated information that can be exploited by higher decision levels. Systems like [10] introduce distributed fusion such as stereo cameras, or information fusion of GPS receivers with inertial sensors, performed in an autonomous way.

2.3 Sensor technology

Automobile driving requires a precise knowledge of the environment that allows the driver to be oriented and know the position of the road and of the obstacles. In ADAS, this knowledge is obtained through the information supplied by the sensors and the posterior processing of that information; so, the election of the sensors used in ADAS application is critical. The most relevant sensors in automotive applications and the different possibilities of sensor fusion between them is explained next.
Figure 10 shows a possible configuration of an ADAS, including a long distance RADAR for adaptive cruise control, LIDAR for automatic emergency braking, pedestrian detection and collision avoidance, camera for traffic signs classification, lane departure warning, lateral vision and parking assistance, short range RADAR for blind spots control, back collision warning and close frontal traffic alert. Finally, it proposes ultrasound sensors for parking assistance.

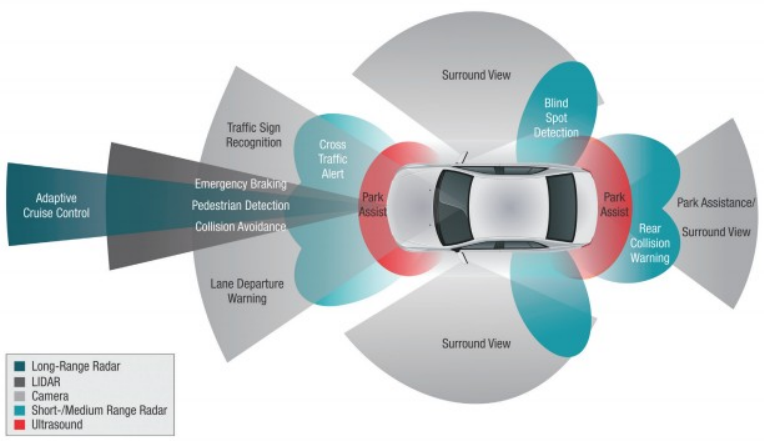


Fig. 10 ADAS sensors proposal [34]

2.3.1 Laser scanners

Laser scanners are devices widely studied in the scientific literature dealing with obstacle detection in automotive applications, specially as a complement to visible spectrum cameras, either monocular or stereo. These devices emit a variable number of laser beams, usually in making layers, and have sensors detecting the position in the three dimensional space where the beams hits an obstacle, obtaining its [x,y,z] coordinates with respect to the sensor's system of reference. The set of detection obtained in a cycle is called point cloud. Nowadays, laser scanners provide enormous amounts of information in point clouds of up to 2.2 million points per second in high-end laser scanners [35], and up to 300,000 points per second in other more affordable ones [36]. All this information must be processed searching for obstacles, usually in in a first step of the environment acquisition process that must be completed before continuing with the sensor fusion. This processing implies an important computational cost worth studying and that must be performed in every read, with immediately expiring results. In [37] an index generation is proposed to avoid the search in non-relevant sub-trees as an effort to accelerate the processing of the cloud.

In a traditional approach of the laser-camera fusion, the usual function for the laser scanners is the generation of a Region Of Interest (ROI) in the image and some help in the obstacle classification. Laser scanners represent the reality by means of a three-dimensional cloud of points, supplying information about the position and distance of the obstacles hit by the laser beam. Later, a different sensor performs a classification of the ROI using computer vision techniques [38].

Another approach is presented in [14], that argues for sensor fusion between RGB information of a camera and dense point clouds for pedestrian detection. This approach uses deformable parts classifiers trained for this purpose, associated with the parts of the point cloud corresponding to the detected object.
Laser scanners provide information not only about the position of the detected object, but about other measurements such as reflectivity, that can help to determine the kind of object being dealt with. This characteristic is used in road lines detection and modeling of the driving environment, using information provided just from a laser scanner [39–42] or as sensor fusion between camera and digital maps [43].

As a complement to obstacle detection and road characteristics definition, laser scanners allow the immediate generation of Simultaneous Localization and Mapping (SLAM) maps, adding the possibility of information fusion with road digital maps, or even with databases of images of the road and its surroundings [44, 45].

Obstacle detection is based on the detection of sets of points, called clusters, with categorizable mathematical characteristics that suggests the presence of an obstacle in a given region of the space. These clusters are usually

defined by its three-dimensional euclidean distance between the included points, sometimes modified by several parameters, such as the geometry of the laser scanner used, the distance to the obstacle and some other factors [12]. Other authors are proposing the use of the Mahalanobis distance [46] as a generalization of statistical clustering algorithms [47, 48], but these algorithms does not seem to adapt well to the particularities of the laser scanner point clouds. The Mahalanobis distance has also been studied as an extension of the K-means algorithm, trying to solve the problem of the initial estimation of the covariance matrix [49].

Before obtaining the clusters, the mathematical prerequisites can be refined, modeling the surface of the road in order to eliminate from the point cloud the points belonging to the road, that can not be considered as obstacles, or adding points to the clusters found thanks to geometric restrictions that determine that these points belong to that existing cluster even though they do not meet the mathematical conditions for cluster inclusion [33].

Figure 11 shows the clusters (red dots) extracted from the point cloud, and the ROIs (blue squares) generated for computer vision classification.

Fig. 11 Cluster extraction and ROI generation in image. Red dots represent the clusters found, and blue squares are the ROIs for bicycles search using computer vision algorithms.

2.3.2 Visible spectrum cameras

Vision is the main sense in a human being. Most of the stimuli received during a driving session come from the vision sense, so visible spectrum cameras are an essential in ADAS. Some elements in the road such as traffic signs and lights are difficult to recognize with other sensors; apart from that, cameras offer a very rich and relevant information. The reduced cost for cameras is promoting its use in ADAS applications, even though information treatment requires a high computation capacity.
Visible spectrum cameras can be divided into monocular cameras and stereo cameras.

Monocular cameras

Monocular cameras are widely used in automotive applications, as its cost is reduced and can be easily installed in many locations in the vehicle, providing information without a significant increase in the vehicle cost.

Some works as [14] use monocular RGB cameras for pedestrian classification using deformable part models and fusing the information with dense point clouds coming from a laser scanner.

Stereoscopic cameras

Stereo cameras are made up two monocular cameras mounted in a parallel way, and separated by a known and fixed distance. This position mimickes the layout of the eyes of the animals, and allows the estimation of the object's distance in the image. The distance from the lenses to every pixel in the image is represented in the disparity map, offering important advantages over monocular cameras because of the addition of a third dimension, very relevant in ADAS applications. The drawback is the important computational requirements for the disparity map creation.
Stereo cameras are, per se, a cooperative sensor fusion system, as it uses two independent sensor to obtain the depth, a new information not available previously. The work [13] takes advantage of the point clouds obtained from the disparity map in a stereo camera for road plane estimation. The same work obtains the road plane in a similar way from the laser scanner point clouds.

2.3.3 Thermal cameras

Thermal cameras, also called far infrared (FIR) cameras, are an extremely useful sensor in poor illumination conditions. Coming from a military use and severely restricted in the past, are now widespread and can be used in ADAS.

These cameras are useful not only in darkness, but also in very uneven illumination conditions, such as extreme illumination in part of the scene and poor illumination in another part. These conditions, common in tunnel exits

or sun/shadow scenes, visible spectrum cameras need to adapt the iris in order to obtain a clear image of the illuminated part of the scene, hence letting the least illuminated part in dark. Thermal cameras are able to detect obstacles such as pedestrians [11, 50] in the darkness as well as in good illumination conditions in part or in the whole scene.
Sensor fusion of thermal cameras with other image sensors require the knowledge of the intrinsic parameters of the thermal cameras in order to achieve the data alignment. Unlike visible spectrum cameras, thermal cameras calibration is not a simple task as they do not operate in the visible spectrum, and requiring special techniques. The work [51] proposes the use of a thermal metallic calibration pattern for the classic techniques in [52], while [53] offers a more sophisticated calibration, involving thermal sensors, RGB and LIDAR. The paper [54] shows a device made of two visible spectrum cameras providing stereoscopic vision, and a third infrared camera. Calibration is achieved by means of a standard calibration pattern with added heat emitters [55]. Several different calibration patterns specially designed for thermal cameras are presented in [56].
Once obtained the intrinsic parameters, it is possible to perform the sensor fusion with other vision-enabled devices such as laser scanner or depth sensing cameras as in [56], adapting the information between the systems of reference of the diverse sensors, applying the proper geometric model to each of them (pin-hole camera model) and the rotation-translation relation between sensors.

2.3.4 Ultrasonic sensors

Ultrasonic sensors suffer the drawback of being sensitive only at short distances, up to a few meters. Additionally, they experience errors at high speeds due to air pressure, making its use only recommended for parking applications and obstacle detection at low speed. Nevertheless, they are precise and accurate in these conditions, as explained in [57].

2.3.5 3D Cameras

3D cameras offer three-dimensional information of the captured scene by using several sensors. An example of these cameras are the Kinect cameras, a device initially developed for gaming that has been a breakthrough in perception for robotics and automotive applications, by making affordable technologies previously too expensive to use in many projects.
The first version of the Kinect camera was designed specifically for indoor use, so its application in ADAS was restricted to the inside of the car, for example as a driver monitoring tool as in [58]. The first version of Kinect included an infrared emitter which, together with a CMOS sensor, allowed the Kinect I camera the generation of a depth map of the environment. Additionally, a RGB camera supplied images for fusion with the 3D information aforementioned.

In 2013, the Kinect II camera was launched with improved characteristics and also capable of outdoors operation, allowing some novel ADAS applications as in [59].
The paper [60] proposes the use of the Kinect II camera by taking advantage of its RGB camera and its depth sensor, for fusion with laser scanner information. Due to the short range of operation of the Kinect II depth sensor, the use is limited to road plane estimation, in a similar way than in [13].

Although not applied directly to ADAS, [61] models the pedestrian's movements using data from the infrared camera for detection of falls in pedestrians. Standard classifiers tend to fail in pedestrian detection in poses different to bipedestation, and a fallen pedestrian might not be detected by a laser scanner, so such a system could be useful in ADAS applications.
Figure 13 shows the two Kinect camera models including their sensors.

3D cameras with Time of Flight technology, such as Kinect II, use a modulated source of light for scene illumination, and sense the reflection of that light produces by the obstacles. By computing the difference in phase between the emitted and received wave, the distance to the obstacles is obtained, as explained in [62] and as shown in figure 12. These cameras can also offer a 3D point cloud representing the scene so point cloud interpretation algorithms might apply. While Kinect II use indirect time of flight as depth sensing technology, Kinect I estimates depth based on structured light, with triangulation between infrared camera and infrared laser [63].

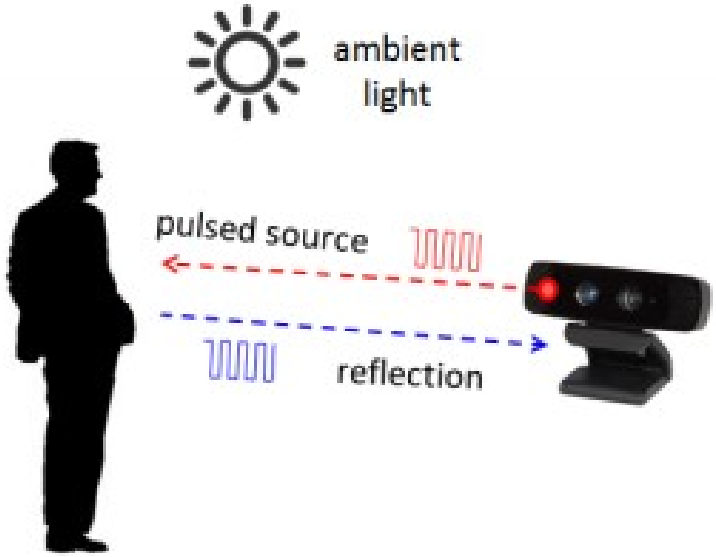
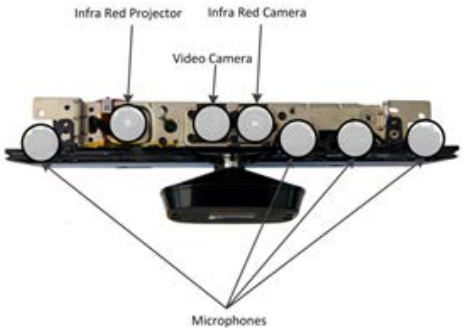


Fig. 12 Time of flight 3D camera operation. The distance to the object is measured as the difference in phase between the emitted and the received wave.

2.3.6 Radar

Radar based sensors use the *W* band of the spectrum, with a bandwidth from 15 up to 111 GHz, usually near the 77 GHz frequency. A radar sensor emits electromagnetic waves that bounce back from the obstacles in the scene, allowing the sensor to perceive the echo and extract information.



(a) Kinect I

(b) Kinect II

Fig. 13 Kinect cameras I and II characteristics.

The cost of the radar sensors is decreasing, so its installation in ADAS systems is becoming affordable [34] for applications such as adaptive cruise control, blind spot detection, emergency braking, frontal collision alert, pre-collision detection, back collision protection, and stop&go systems for distance to the preceding vehicle control [64].

- Long-range radar (LRR)

  These systems can detect obstacles located more than 100 meters away, and are usually mounted in the front of the vehicle facing forward. The frequency used is around GHz, and the main application is obstacle detection at high distances, emergency braking , pre-collision detection and stop&go.

- Short-range radar (SRR)

  Short-range radar uses frequencies around 24 GHz, and provide obstacle detection at short distance, blind spot detection, lane departure prevention and crossing traffic control.

2.4 Training and classification technologies

Obstacle classification is usually based on the extraction of relevant characteristics from a training set of samples containing the Object Of Interest (OOI), called positive samples, so a classifying system can apprehend the characteristics defining the OOI. Additionally, multiple samples not containing the OOI, called negative samples, are provided to the training system. Usually, the number of negative samples is much higher than the number of positive samples, as the common environment contains more negative objects than positive objects.
The most used characteristics in the scientific literature are Haar-like characteristics [65], Histogram of Oriented Gradients (HOG) features [66] and Local Binary Patterns (LBP) [67].
HOG features define the general shape of the OOI, while LBP tend to detect the texture of the OOI. These characteristics can be considered separately as in [18], or combine them in a single classifier as in [68], studying the structure of the OOI in a topological and features way. Haar-like features were used initially in facial detection, and it is a specially fast method, although less precise than the aforementioned alternatives. For these reasons, it is sometimes used as a fast predetection method combined with HOG features, as in [69], or as a unique feature [70], using Adaboost as learning algorithm. HOG keeps being object of study and use because of its good performance in the practice [14, 18, 71], despite the limitations noted in [71].

A possible categorization of the classifiers studies the consideration of the scene made by the classifier. If the obstacle is considered as a whole, they are called Holistic Classifiers, and if they consider the obstacles are made of several mobile parts deserving individual attention, they are called Mobile Deformable Parts Classifiers.

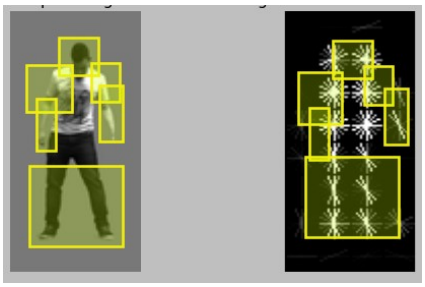
- Holistic Classifiers

  It is the most commonly used method and keeps offering excellent results in real applications, as detailed in [72] y [73].

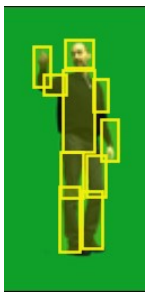
- Mobile Deformable Parts Classifiers

  These classifiers use local experts [74, 75], dedicated exclusively to the classification of some precise parts of the body. This is based on the assumption that local characteristics are easier to model than global characteristics, and also easier to generalize [76, 71]. Figure 14 shows an original image and its segmented HOG representation (figure 14a) and a final image with the detected segments (figure 14b).

  Deformable Part Model (DPM) answers the need for classification of objects with mobile parts, such as humans (specially pedestrians and cyclists) , as explained in [76]. The work [75] proposes the use of Random Forest (RF) with local experts for HOG and LBP features, using parameter optimizations in order to minimize the classification error.



(a) Original image and HOG segmented in deformable parts.



(b) Detected mobile parts

Fig. 14 image and HOG segmented in deformable parts.

Another possible categorization of the training systems considers the genesis of the training sets used.

The generation of a training set requires the compilation of a large number of images containing the OOI, and to label each of the appearances of that object in the images. Additionally, a large number of negative samples must be gathered for the training process, keeping another set of fully labeled positive samples for the classifier refining and testing process. Although variations between authors exist, a commonly used relation is 60% of the samples for classifier training, 30% of the samples for classifier refinement, and finally a 10% of the samples for testing and statistics generation about the classifier's performance.
In an attempt for avoiding the enormous effort devoted to the generation of these training and testing sample sets, some authors have studied the gathering of samples, and specially the testing, through synthetic images and virtual worlds [18, 77].

- Real world samples trained classifiers

  The generation of training datasets using real world samples is an extremely costly process in terms of time and resources. It requires the gathering of multiple samples of the OOI, located in many different environments and backgrounds, and in multiple poses, so the classifier can learn all the possible shapes that the OOI can adopt in the real world.

  Additionally, after the images collection, a labeling process of the OOI in the samples must be achieved, usually in a manual way.

If a training based in deformable mobile parts is intended, and the characteristics of these mobile parts are of interest, another individual labeling process of these parts is required in the samples set. Moreover, if not only the categorization of the OOI but also its orientation in the space or the identity of the subject are also of interest, a heavy workload is required for further labeling.

The main advantage for real world samples is that they are similar to those found by the classifier in real working conditions, so the expected results are, a priori, better. Examples of these kind of samples are the data sets KITTI from the KIT [20] and the INRIA dataset [66].

- Synthetic samples trained classifiers

  The main advantage for this samples collection method is that, once generated the virtual world or the samples generation method, the generation of training datasets, and specially of testing sets is almost automatic.

  Since this synthetic images are computer generated, the location of the OOI as well as the mobile parts that might be of interest and the identity and orientation of the subject in the training data set are software determined and provide a faster and less costly training process.

  The drawback for this kind of samples set is that they do not match exactly the characteristics of the images found in real world operation, so the expected results of the training are, a priori, worse. To avoid this problem, Domain Adaptation methods are applied using real world samples that allow to improve the required effectiveness [74, 18, 75].

(a) Synthetic image from a virtual world [18]

(b) Pedestrian images from INRIA dataset [66]

(c) Pedestrian images from LSI-CROMA

Fig. 15 Synthetic and real images for training.

These training sets can be fully virtual, such as the ones generated by virtual worlds software, or extracted from reality using background removal techniques in order to obtain the OOI ready to be inserted in a synthetic or real world background. Figure 17 shows an example of synthetic image obtained using croma techniques. Figure 17a is the extracted pedestrian inserted into a solid green background. The hog features representation shows only features from the OOI, as the background has no gradient at all. Figure 17b shows an extracted OOI inserted into a real world background. The work [78] offers a fully synthetic dataset, with full 360º, automatic depth map generation, still images and video fully labeled, making a extremely useful tool for training, testing, and domain adaptation research.

The generation of dataset for training and later classification is a very time and resources consuming task. As the best results are obtained in real operation classification when the samples used for training the classifier come from the same sensors and in the same conditions, the use of public dataset is not the best option.

The use of synthetic training techniques through virtual worlds [77, 74] can be complemented using real world images for domain adaptation as in [18]. These techniques for dataset generation offer great advantages, as provide automatic dataset generation without human intervention nor manual labeling.

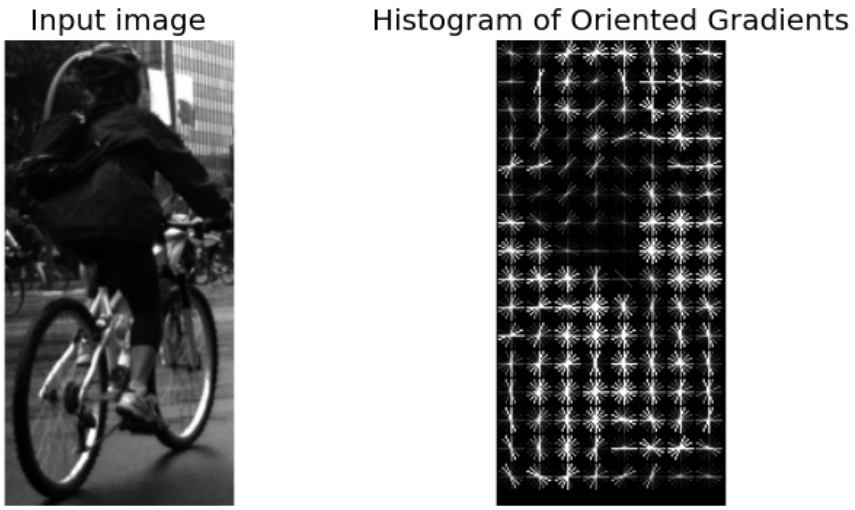


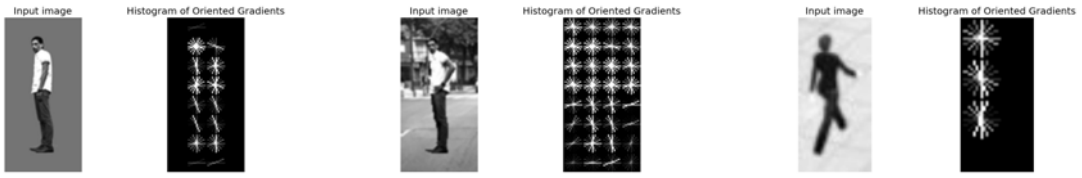
Fig. 16 Histogram of Oriented Gradients representation (HOG).

The possible loss of accuracy, in its case, can be compensated by the use of virtually infinite testing sets without cost. Several techniques for learning systems improvement exist, by expanding intraclass diversity [79], and can be applied to synthetic training data sets.

Figure 17a shows a pedestrian in a gradientless background, and its HOG representation contains only the OOI information, this is, the part of the image containing gradients. Figure 17b shows a similar pedestrian inserted in a real background, so the HOG representation includes gradients belonging to the pedestrian as well as belonging to the background, as expected in real world scenes. Figure 17c displays a virtual world pedestrian and its HOG representation.

2.5 Data alignment

Sensor fusion requires the use of data alignment algorithms between the involved sensors. An important aspect in the use of sensor fusion between



(a) Croma image (b) Synthetic image (c) Virtual world image

Fig. 17 HOG representation comparison between pure croma image, synthetic image and virtual world image.

laser scanner and visible spectrum cameras is data alignment, that is, the coordinate conversion from sensor-A's system of reference to sensor-B's system of reference when fusing information between sensor-A and sensor-B.

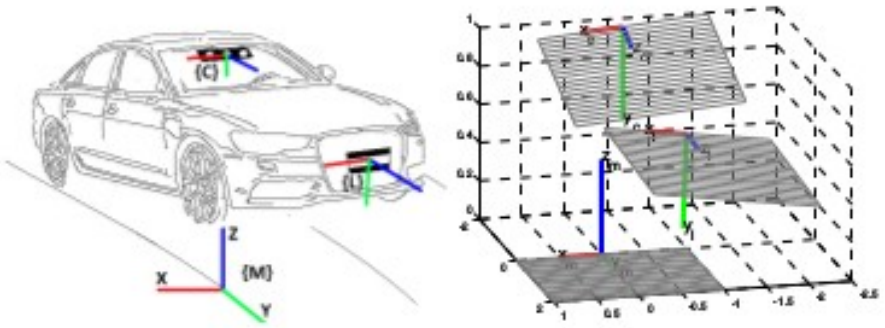


Fig. 18 Data alignment. Configuration of the array of sensors and representation of the translation and rotation between laser, camera and vehicle. A stereo camera is located in the windshield, and the laser is attached to the frontal bumper [13].

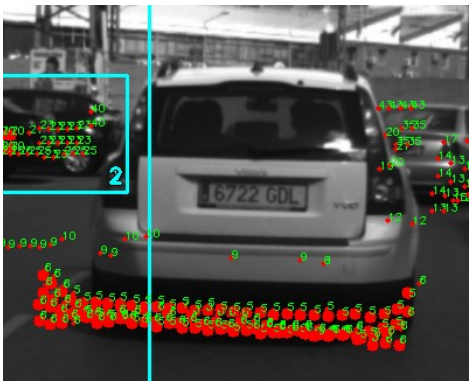
The data alignment process needs to know the intrinsic parameters of the cameras as well as extrinsic parameters (rotation angles and translations) between the fused devices.

Figure 18 shows the systems of reference of the sensors, one located in the windshield and the other located in the bumper, as well as the translations and rotations needed for data alignment.

Figure 19a displays errors in vertical and horizontal angles alignment (the image is displaced with respect to the point cloud) as well as in translation (objects look bigger for the laser than in the camera due to an error in the translation parameter in the direction of the road). Figures 19b and 19c present correct data alignment.

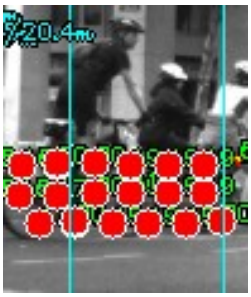
The classical approach in data alignment between camera and laser scanner requires usually a recognizable geometric shape visible to both sensors, such as triangles in the work [80] or the manual selection of meaningful points in a part of the scene common to both devices in the case of [81].

The work [13] proposes the automatic and unattended data alignment between laser scanner and stereo camera by means of the detection of the road plane and the computation of two out of the three rotation angles between the sensors, and a recognizable obstacle in the road visible for both sensors in order to compute the third rotation angle. This procedure requires a sufficiently dense point cloud representing the surface of the road, which is not always available. Works such as [60] use a Kinect II camera in order to obtain a dense point cloud allowing the extraction of the road plane and a reliable data alignment.



(a) Incorrect data alignment in rotation (the image is displaced with respect to the laser point cloud) and in translation (the point cloud looks bigger than objects in the camera as one of the translation parameters is wrong).

(b) Correct data alignment for pedestrian



(c) Correct data alignment for bicycles.

Fig. 19 Data alignment between laser scanner and camera.

2.6 Conclusion

The basics of the technologies used in the presented thesis and its state of the art have been introduced in this section.

Sensor fusion has been studied as a multidisciplinary field with civil and military applications, as well as sensor fusion architectures with its different

categorizations as a result of the source for sensor fusion.

Sensor fusion depends largely on the election of the used sensors. The combination of the sensors used in the fusion can determine the quantity and quality of the available information, and the results obtained from the fusion. Trends in sensor technology commonly used in ADAS applications have been detailed, including the fields of application for each of them and some examples of its use.

In respect of the use of the information obtained from the sensors, several training and classification technologies used in obstacle detection have been studied, emphasizing in the types of classifiers and the kind of samples used for the training process.

Finally, the essential data alignment of information coming from the sensors has been studied, reviewing the foundations as well as novel trends of research. Data alignment between sensors require the estimation of physical factors, such as the relation in orientation and translation between the sensors involved in the fusion. Safe and controlled environments such as a laboratory allow the use of geometric patterns for occasional parameter estimation, while dynamic and changing environments as in ADAS applications need unattended parameter estimation systems, capable of real time operation without human intervention, and specially without the need for technical personnel.

Chapter 3

General description

ADAS applications depend essentially on the detection and identification of the obstacles existing in the driving environment. Pedestrians and bicyclists are the most vulnerable road users, so it is crucial to work on its reliable detection.

The present thesis proposes a obstacle detection and identification system fusing information from a laser scanner and a visible spectrum camera, emphasizing in the classification of obstacles such as pedestrians, bicyclists, motorbikers and cars.

Figure 20 shows an overview of the proposed system, that will be explained later in this chapter.

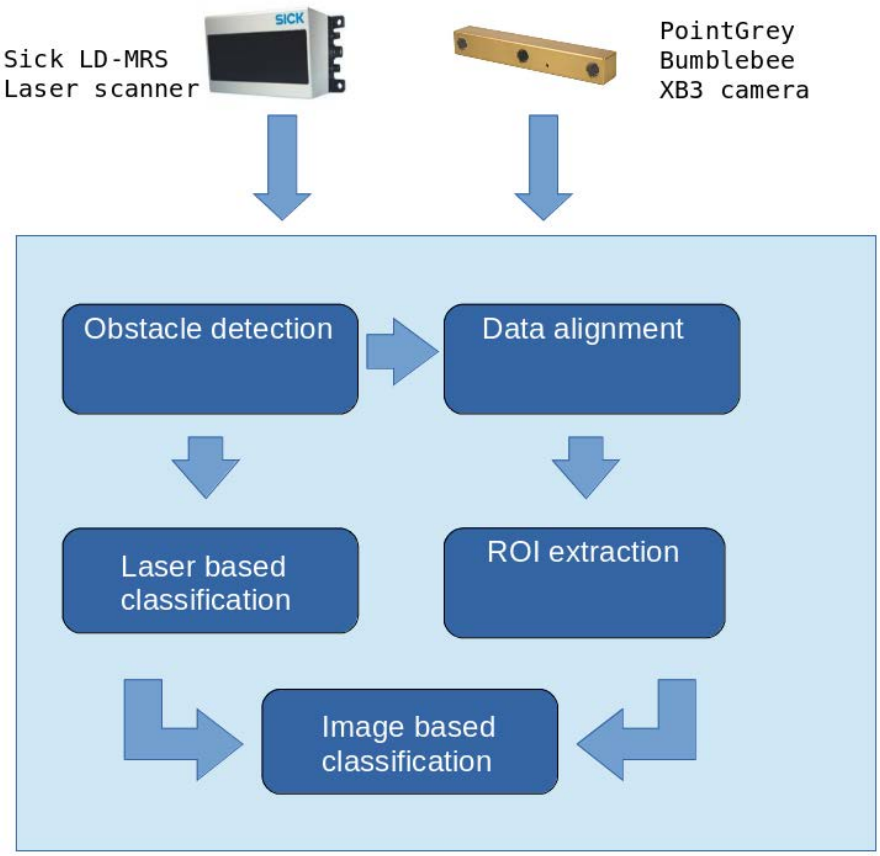


Fig. 20 System overview

3.1 IVVI 2.0: The research platform

The IVVI 2.0 research platform from Intelligent Systems Lab is described in the work [10], and is intended to offer solutions to problems related to ADAS applications. To this end, several sensors and information management systems are available. Figure 21a shows the platform and its sensors.

The sensors available in IVVI 2.0 are:

- Sick LD-MRS 4 layer laser scanner (Figure 21b)
- Bumblebee XB3 visible spectrum stereo camera (Figure 21c)

- Velodyne VLP-16 16 layer laser scanner (Figure 21d)
- Thermal camera (Figure 21e)
- Kinect II system (Figure 21f)
- GPS receiver (Figure 21g)
- Inertial measurement unit (Figure 21h)
- Frontal-lateral cameras (Figure 21i)
- SIMBA system for CAN-BUS information analysis (Figure 21j)

The sensors relevant for the present thesis will be studied in depth later in this chapter.

(a) IVVI 2.0 research platform

(b) Sick LD-MRS laser

(c) XB3 trinocular camera

(d) VLP-16 Velodyne laser

(e) Thermal camera

(f) Kinect II system

(g) GPS receiver

(h) IMU

(i) Frontal-lateral cameras

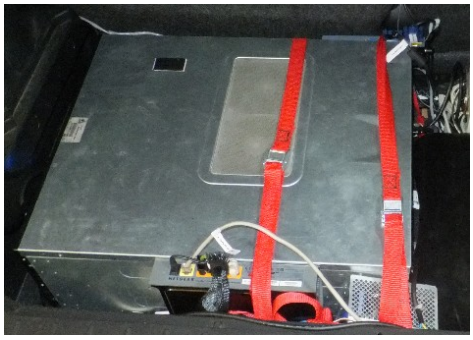
(j) SIMBA system

Fig. 21 IVVI 2.0 research platform sensors.

3.2 Information processing system: TESLA

Processing and fusion of the information received from the sensors is performed in a high-end computer called TESLA, as will be referred to in the rest of the document, and is shown in figure 22a, equipped with 2 Intel® Xeon® CPUs Processor E5-2620 (32 nm, 6 cores, 12 subprocesses, 2.00 GHz base frequency, 2.50 GHz maximum turbo frequency). The system includes two graphics cards with information processing function: NVIDIA Tesla C2075 card (Fermi GF100 chip, 448 CUDA cores, 6GB GDDR5 memory, 1.03 TFLOPS single precision, 0.52 TFLOPS double precision) and NVIDIA Tesla K40c card (Kepler GK110B chip, 2880 CUDA cores, 12GB GDDR5 memory, 4.29 TFLOPS single precision, 1.43 TFLOPS double precision).
This computer executes an Ubuntu Linux Operating System, and includes 32 GB DDR3 1333 MHz RAM, 4+2 terabytes of storage in mechanical hard disks, a 512 GB Solid State Disk (SSD) for tasks requiring high speed writing, such as sequence captures, and finally an additional 128 GB SSD for Operating System storage.
Sensor management and interaction with the sensor fusion and classification software is carried out by a ROS (Robotic Operating System) Indigo version. The standard way of working in sequence analysis consists on the synchronized capture in a .bag file of the information supplied by the sensors . This capture is performed by the rosbag utility, provided by the ROS system.
The sensors are connected to the different ports provided by TESLA for that purpose. Thermal cameras and Bumblebee XB3 are connected to TESLA's Firewire (IEEE 1394) ports; frontal-lateral cameras use USB 3.0 interface, and are connected to TESLA through a USB 3.0 hub. TESLA, SIMBA and Sick LD-MRS and Velodyne VLP-16 laser scanners are connected together through a gigabit Ethernet switch (see figure 22b). The GPS receiver and IMU are connected to TESLA through USB 2.0 ports.

Human operator system control is performed remotely through an SSH connection, or in person using a work place located in the rear seat of the IVVI 2.0, as seen in figure 22c. The driver can receive information and alerts from the system through an small display located in the dashboard as shown in figure 22d.



(a) TESLA information process system

(b) Ethernet switch for sensors and TESLA

(c) Operator work place

(d) Dashboard display

Fig. 22 TESLA information process system

3.2.1 Robotic Operating System (ROS)

ROS is an extremely flexible framework, devoted to robotic management systems. This environment provides a flexible and robust platform that solves sensor interaction and information synchronization for multiple sources with dissimilar characteristics.

One of the advantages of ROS is the seamless integration of pieces of software from several researchers, just by following certain good practices for

compatibility.

ROS basic concepts

A ROS system is made up of several programs called Nodes running independently in one or several machines, interacting and communicating through TCP/IP by means of the publishing and subscription to some data types called Topics.

In the present system, the sensors' controlling nodes are offering Topics such as the images from the central camera in the XB3 (called \stereoCamera\image) or the LD-MRS laser scanner point cloud (called \laserScanner\cloud). Some parts of ROS (`Nodes`) are sensor controllers, offering (`Publish`) these `Topics` with the meaningful names aforesaid. Other programs (`Nodes`) intended to process this information, `Subscribe` to these `Topic`, so every time that some new information is available from the sensors and is `Published` in the `Topic`, they receive it immediately as shown in figure 23. If the `Nodes` demands it, ROS is able to synchronize the `Published` information, so the `Nodes` can get information captured from the sensors in the most simultaneous possible moments.

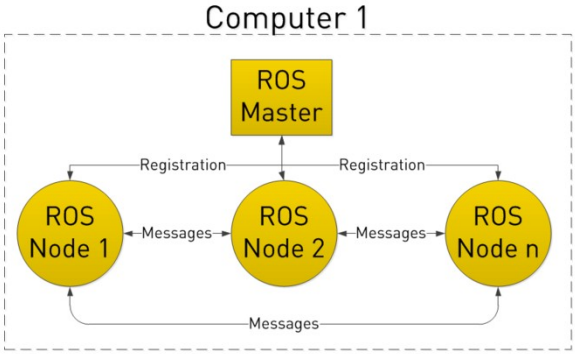


Fig. 23 ROS overview, showing ROS Master and three nodes working together [82].

3.3 Sensors

3.3.1 Sick LD-MRS 400001 laser scanner

Figure21b shows the Sick LD-MRS laser scanner mounted in the IVVI 2.0 frontal bumper. Complete technical information about this sensor can be found in the document [83].The Sick LD-MRS 400001 laser scanner is based on the Time of Flight (TOF) technology depicted in figure 24.

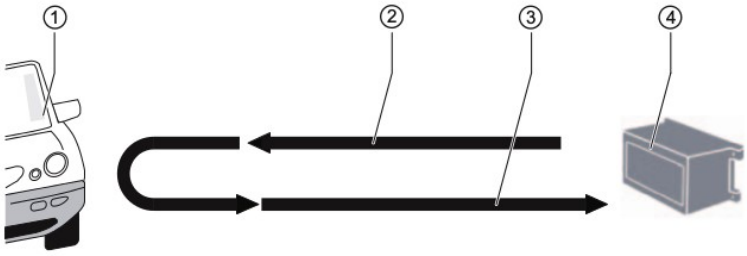


Fig. 24 Sick LD-MRS laser scanner operating mode using Time of Flight technology. 1: Detected obstacle. 2: Emitted laser pulse. 3: Laser pulse is reflected by the obstacle. 4: LD-MRS laser scanner. [83]

The Sick LD-MRS 400001 laser scanner is a device emitting four horizontal and parallel layers with a 0.º divergence between layers, with a horizontal angular resolution configurable from 0.5º up to 0.125º, depending on the scanning frequency. Scanning frequency can be selected from three values: 12.5 Hz , 25 Hz and 50 Hz.

Additionally, this device can be configured in a working mode specially useful for ADAS applications, with a variable angular resolution from 0.125º in the front, 0.25º en the frontal-lateral section, and 0.5º in the lateral parts of the field of vision, as seen in figure 25. The objective of this approach is to improve the detection in the most dangerous parts of the scene in ADAS, that is, right in front of the vehicle. The detection ability is reduced in the sides of the field of view, where an obstacle is less likely to interfere in the movement

of the vehicle. The device has been designed to offer a basic horizontal scan range of 85º in the four layers. Additionally, the two upper layers can extend the horizontal coverage up to 110º, from 50º left to -60º right. This asymmetry is due to the fact that in right lane driving, the right side of the scan tends to be more important, as it covers the border of the road, where pedestrians are more likely to approach, obstacles such as stationary vehicles can exist, etc. The reason for extending only the upper layers is that lower layers are intended for near obstacle detection, while upper layers tend to detect distant obstacles, in which the system can take more advantage of the additional information collected.

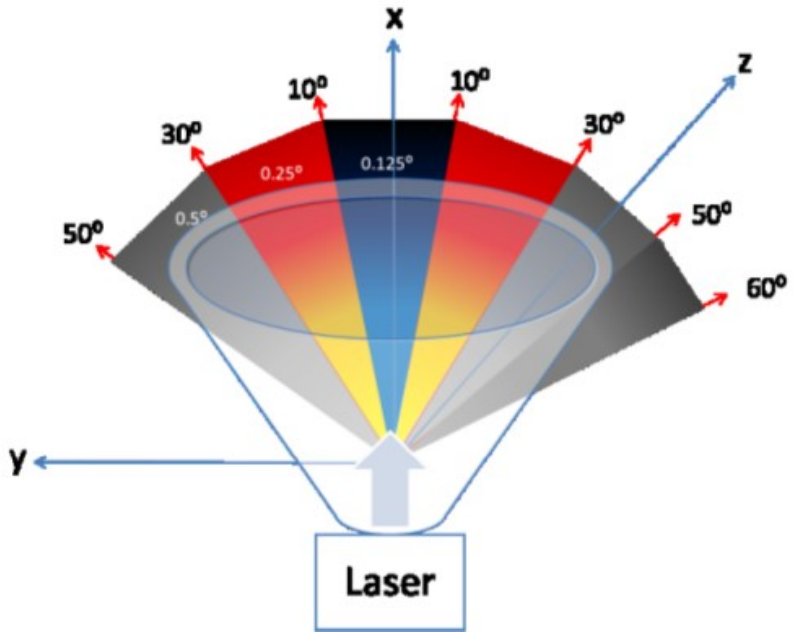


Fig. 25 LD-MRS Sick laser scanner variable angular resolutions.

The laser scanner projects four horizontal layers with a fixed vertical angular divergence of 0.8º, as shown in figure 27. The complete angular coverage is 3.2º, from -1.6º below the horizontal layer up to 1.6º beyond the horizontal layer. This horizontal angular configuration increases the ability for obstacle detection in the case of uneven road surface, as shown in figure 26.

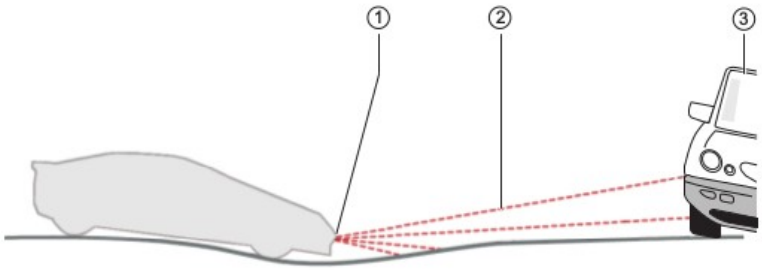


Fig. 26 Increased obstacle detection in the LD-MRS laser scanner due to the 4 layers. 1) LD-MRS laser emitter. 2) Each of the laser layers. 3) Detected obstacle[83].

Fig. 27 Sick LD-MRS laser scanner four layers vertical angular resolution [83].

3.3.2 Computer Vision System

Thermal camera for Far Infrared (FIR)

Figure 21e shows the FIR camera installed in the IVVI 2.0 platform. Several ADAS applications have been developed for pedestrian detection and classification in low or uneven illumination conditions as explained in [50]. Additionally, some studies are under way for extending FIR obstacle detection to other authors such as cars and buses.

Point Grey Bumblebee XB3 camera

The IVVI 2.0 research platform includes a low level fusion cooperative sensor such as the Bumblebee XB3 stereo camera shown in figure 21c, that includes three monocular cameras and hardware able to process the information from the cameras in order to provide the monocular images from the three cameras and the fusion of two of them as an stereo image.

Frontal-lateral cameras

Two frontal-lateral cameras are installed in the IVVI 2.0 in the sides of the front bumper, as see in in figure 21i, to complement the angle of vision of the frontal stereo camera. These cameras provide a lateral point of view, beyond the driver and able to increment the security in intersections and low visibility crossings. Another interesting capability in the IVVI 2.0 is the fusion of these cameras with the laser scanner for obstacle detection and classification from points of view inaccessible for the driver.

Kinect II System

Figure 21f shows the Kinect II system mounted in the dashboard of the IVVI 2.0 for driver monitoring. Taking advantage of its multiple abilities, a system for driver's attention control has been developed, allowing the extraction of Percentage of Eye Closing (PERCLOS), for fatigue estimation, head position statistics, driving attention to the road and rear mirrors, etc.

3.3.3 Inertial Measurement Unit (IMU)

IMUs are devices for acceleration detection, and are commonly used in robotics and aerial and terrestrial vehicles, usually in conjunction with a GPS receiver.

It is a Microstrain 3DM-GX2 IMU including triaxial accelerometer, triaxial gyroscope, triaxial magnetometer and temperature sensor, that can be considered by itself as a low level sensor fusion system [84].

The unit installed in the IVVI 2.0 is shown in figure 21h, and has been used in sensor fusion researches for driving monitoring and aggressive or erratic driving behavior modeling, as explained in [85–87], and in precision improvement for vehicle geopositioning, fusing its information with the GPS receiver [30].

3.3.4 GPS

Commonly used in ADAS applications, GPS receivers like the one shown in figure 21g are specially useful in collaboration with maps systems. The standard precision for GPS systems might not be enough for it use in ADAS, so they are complemented with differential GPS systems like the one available in the LSI, and with an IMU for vehicle position inference in environments where the GPS reception is not possible or is poor, such as tunnels or urban canyons, as explained in [30].

3.3.5 CAN-BUS reader

The work [88] describes the Sistema Integrado de Monitorización Bidireccional del Automóvil (SIMBA), shown in figure 21j. It is a monitoring system for CAN-BUS signals through the ODB2 standard port, containing information about steering wheel position, gas pedal, brake, rpm, vehicle speed, etc. Integrating all these information it is possible to model the type of driving and to detect erratic and aggressive driving, as explained in [87, 86, 85].

3.4 Information system power supply

IVVI 2.0 includes a power supply system to power both sensors and the information system when the vehicle has not access to electric power. The power supply system uses two Absorbed Glass Mat (AGM) batteries connected to an intelligent disconnection and charge system that ensures the continuous use of the vehicle, as well as the correct recharge and maintenance of the batteries. This kind of batteries deals better with charge/discharge cycles, specially the deep discharge that can damage standard batteries.
The engine battery is connected to an auxiliary battery located inside the cabin, and together feed an inverter able to provide up to 1,100 watt to the sensors and computer system. Should the charge level of the batteries ensemble reaches dangerous limits, the protecting device shown in figure 28a disconnects the engine battery from the auxiliary battery, ensuring that the vehicle is able to start the engine.
As long as the motor is running, the alternator is charging both batteries, and while the vehicle is idle in the garage, an intelligent battery maintainer (see figure 28b and 28c) connected to the standard electric supply is in charge of the battery charge management in order to optimize its service life.

3.5 Hardware and software architecture

3.5.1 Hardware architecture

System's hardware architecture consists on the involved sensors (laser scanner, see figure 21b and stereo camera, see figure 21c), the networking devices shown in figure 22b, the power supply system (figure 28) and the information processing system TESLA, the computer running the ADAS algorithms in the IVVI 2.0 platform (figure 22). The ensemble is explained in figure 29.

(a) Engine battery protecting device

(b) NOCO Genius G3500 intelligent charger

(c) Intelligent charger for both batteries

Fig. 28 Power supply intelligent system for IVVI 2.0

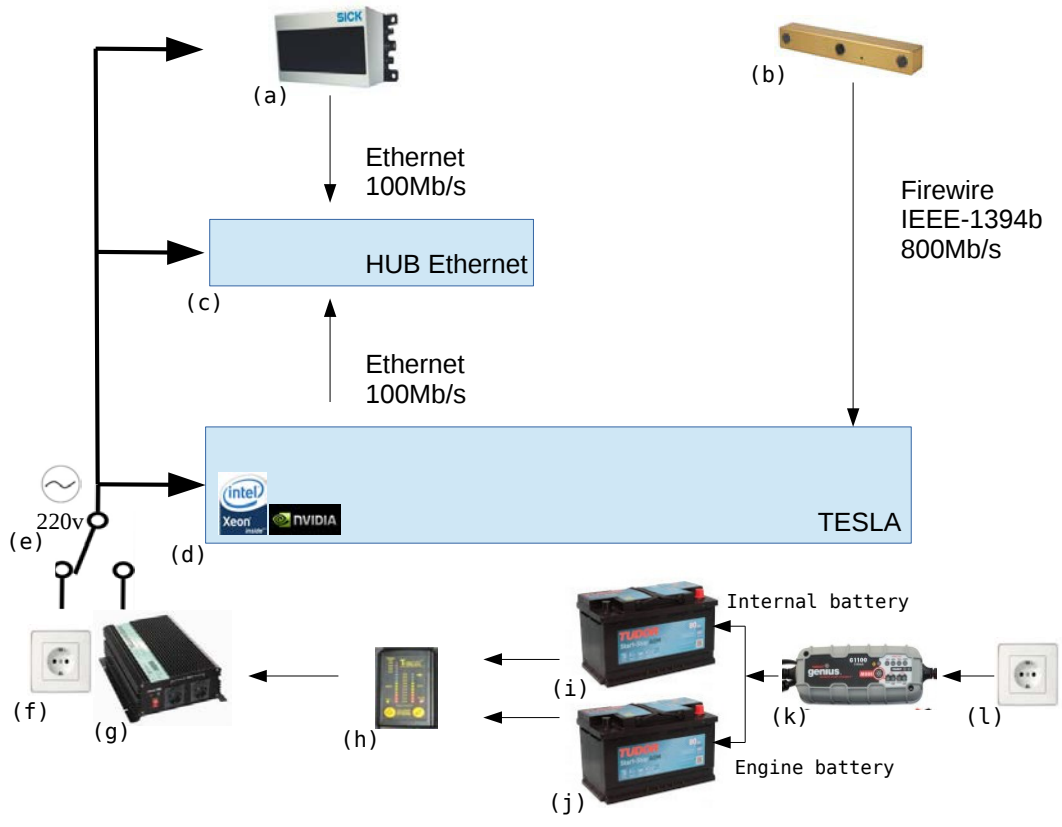


Fig. 29 System's hardware architecture. (a) Laser scanner Sick LD-MRS, (b) Bumblebee XB3 camera, (c) Interconnection Ethernet HUB, (d) TESLA information processing system, (e) Batteries/standard electric supply switch, (f) Commercial power supply, (g) Inverter from 12v DC -> 220v AC, (h) Battery protecting system, (i) Auxiliary battery for system power, (j) Engine Battery, (k) Intelligent battery charger, (l) Commercial power supply.

The laser scanner includes a Ethernet interface used for communication with TESLA through the Ethernet switch shown in figure 22b, connected also to TESLA.
These configuration provides TESLA and laser scanner interconnection. Additionally, the cable connection from the switch to the Internet provides access to TESLA from desktop computers in the LSI. This possibility is specially useful for processes requiring heavy computing capabilities, such as classifier training, whose time to completion are unacceptable in common desktop computers.

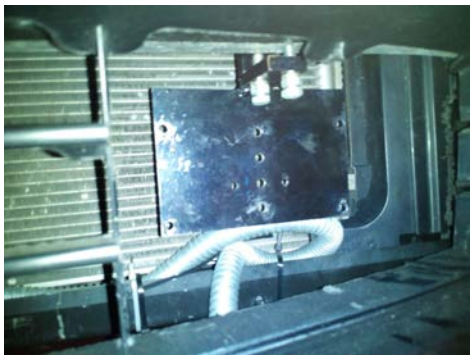
The XB3 Bumblebee camera shown in figure 21c offers a FireWire IEEE-1394b interface working at 800Mb/s, and is connected using a cable to the FireWire card installed in TESLA. This camera is installed in the inside part of the windshield using adjustable plastic fixing brackets fabricated in a 3D printer, as seen in figure 30. These brackets offer vertical angle (pitch) adjustment, keeping roll and yaw. Additionally, it is possible to extract the camera at any time if needed, remaining the brackets attached to the windshield. This feature allows the use of the camera for other purposes in the lab, keeping the translation extrinsic parameters with respect to the laser scanner, and two out of the three extrinsic rotation parameters, simplifying the extrinsic parameters calibration every time the camera is removed. The design of the plastic bracket is oriented to control the glare from the sun reflection in the bonnet, and is prepared for the installation of a matte black surface under the camera to avoid reflections from the dashboard.

Fig. 30 Bumblebee XB3 camera installation in the IVVI 2.0.

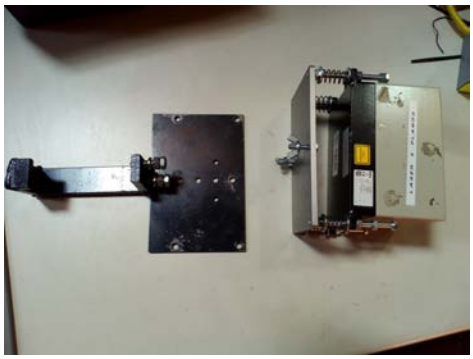
The Sick LD-MRS laser scanner is installed in the front bumper of the IVVI 2.0, using special steel brackets allowing legal driving in public roads as the ensemble does not protrudes the vehicle, as seen in figure 21a and 21b.
Figure 31 shows the laser scanner installation in the IVVI 2.0. It is a metal bracket (figure 31a) fixed to the vehicle's metal frame and offering drills (see figure 31b) for the fixing of the orientation system shown in figure 31c and 31d.
The ensemble of laser plus orientation system allows the control of laser angles in three axis, actuating the nuts provided for spring compression. The whole ensemble can be removed from the vehicle in case of necessity.

This system has been developed in the LSI with the help of the UC3M's Technical Office, and has been a great help for the project, as it provides an easy control for the laser angles with respect to the system of reference, mounted in the IVVI 2.0 or in any other structure for testing.

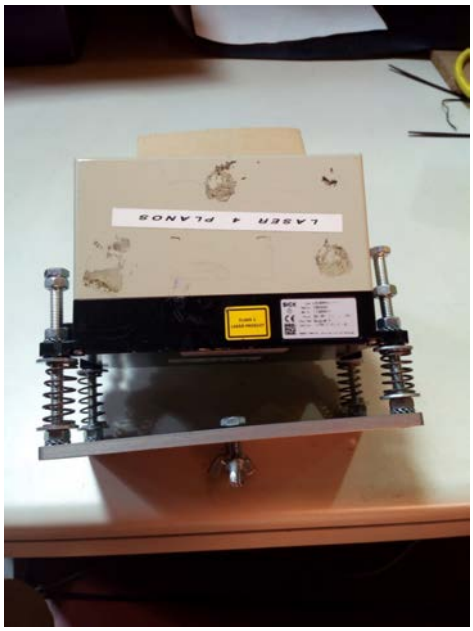
The electric and data connection for the laser are conducted to the trunk through a flexible metal pipe under the vehicle, as shown in figure 31a.



(a) Metal bracket for laser in IVVI 2.0



(b) Metal bracket and orientation system



(c) Upper view of the orientation system

(d) Side view of the orientation system

Fig. 31 Laser scanner installation in IVVI 2.0

3.5.2 Software architecture

The present system has been written in C++ language using the Qt Creator framework, taking advantage of the features provided by the aforementioned Robotic Operating System (ROS).

The nodes involved in communication and information processing are working independently, communicating through topics published by the producer nodes and subscribed by the consumer nodes. Figure 32 explains the different nodes and the philosophy of the interconnection and process, that will

be explained next. Although this process is explained sequentially for easy understanding, the system works concurrently, with a continuous message interchange related to different instants in time.

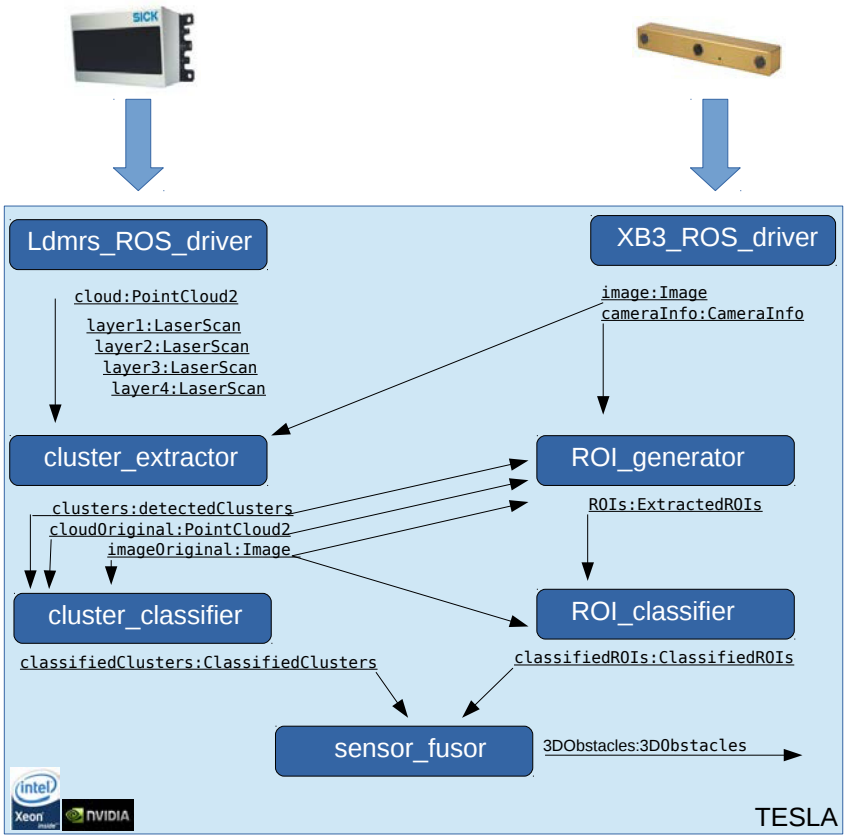


Fig. 32 System software architecture

- Independent developers have produced ROS nodes for sensor interaction, in the presented case the node `XB3_ROS_DRIVER` for the XB3 Bumblebee camera and the node `Ldmrs_ROS_driver` for the Sick LD-MRS laser. These driver controller nodes are launched at the session start and are intended to interact with the sensors' hardware, publishing as ROS topics the information provided by the sensors, like the multiple different images from the XB3 camera (topic `Image`with the image and the topic `CameraInfo` with the camera intrinsic parameters) or, in the case of the laser scanner, the full point cloud and the individual layers

(topics `Cloud` and `Layer1...Layer4`).

- The node `cluster_extractor` is in charge of the laser based obstacle detection and subscribes to the topic `cloud` of type `sensor_msgs/PointCloud` and to the node `Image` of type `sensor_msgs/Image`.
  The node `cluster_extractor` gets, using the `approximate_time` method provided by ROS, the most simultaneous messages from these topics, hence obtaining the most similar reality perceived by both sensors. This node computes the point cloud received and determines the obstacle presented in the scene. Then, it publishes the list of clusters found in the topic `clusters` and the original cloud in the topic `cloudOriginal` and the original image in the topic `imageOriginal`. The three topics are published using the same time stamp as the original image, so when the system needs later this information, it can easily collect all the matching information.

- Running in concurrence with the aforementioned node, another node called `ROI_generator` subscribes to the topic `cameraInfo`, which is generated by the node `XB3_ROS_DRIVER`, ans subscribes too to the topic coming from the node `cluster_extractor`: `clusters`, which contains the clusters corresponding to the detected obstacles in the original cloud, subscribes to the topic `imageOriginal`, which contains the original image from the camera matching the time stamp of `cloudOriginal`, and also subscribes to the `cloudOriginal` topic, which contains the original point cloud. Using data alignment techniques explained later, this node extracts the ROIs in `imagenOriginal` corresponding to the `clusters`. Once determined the ROIs, they are published in the topic `ROIs`.

- The node `cluster_classifier` collects messages from the topic `clusters` and `cloudOriginal` in order to apply the algorithms for `clusters` classification as pedestrian, car, bike, etc, and the `imageOriginal` topic in order to represent the clusters in the image in case of necessity. After performing the classification, this node publishes a message in the topic `classifiedClusters` containing the same information than the message received from, plus the classification obtained for each cluster.

- The node `ROI_classifier` subscribes to the topic `imageOriginal` and to the topic `ROIs`, so it can extract the subimages contained in the ROIs and, applying the classifiers for computer vision, determine the kind of obstacle contained in the image. Once performed the classification of every ROI, it publishes a message in the topic `classifiedROIs`, including the same information received from the topic `ROIs`, plus the classification obtained for each ROI.

- Finally, the node `sensor_fusor` performs a high level sensor fusion between the information coming from the laser (topic `classifiedClusters`) and from the camera (topic `classifiedROIs`). The results obtained from this fusion process are published in the `3DObstacles` topic.

The modular design of the nodes and the publication of all the intermediate results in the form of topics, allow the extension of the system with additional nodes taking advantage of the existing information. Alternative ROI or clustering generation using some context information, the use of alternative classifiers, linking with navigation systems only interested on the obstacles found disregarding its classification, or any other application requiring intermediate information can take advantage of the published information.

Chapter 4

Obstacle detection and classification using laser scanner

The presented work uses sensor fusion between laser scanner and computer vision for obstacle detection and classification in automotive applications, with a Sick LDMRS 4-layer Laser Scanner and a Point Grey Bumblebee XB3 trinocular camera. Laser scanner is used for primary obstacle detection and later for classification, and stereo capability from the trinocular camera is used for point cloud ground representation and data alignment parameters estimation; later, one of the cameras from the stereo camera is used as a monocular camera for image capturing. The laser scanner generates a point cloud in which the system extracts the obstacles as clusters of points. These clusters are used both for ROI generation in the images and as information for obstacle classification. The last step in the process performs further information fusion between laser and camera for a final obstacle classification based on machine learning. A database with manually labeled images and point clouds is used for SVM training and testing in the classification process.

4.1 Point Cloud clustering for obstacle detection

The first step in our system is the obstacle detection using laser generated point clouds. This is the most reliable sensor in our system, as it is not affected by illumination conditions but only by some meteorological conditions. The four layer laser sensor obtains a point cloud representing part of the reality in front of the vehicle. Obstacles are part of this reality and can be located as local concentrations of points in the point cloud that can be mathematically categorized as clusters, as seen in figure 33.

Fig. 33 Obstacle detection, represented as a cluster.

Several clustering techniques have been studied in order to obtain the highest and most reliable amount of information from the point cloud. It is important to note that obstacles to be detected will be represented by very few points in the point cloud, typically from four points to not much more than fifty depending on the distance to the vehicle, due to laser limitations. Most of the clustering strategies already available are designed for highly populated point clouds, obtained from high resolution multilayer laser scanners or stereo cameras, and do not adapt well to the studied outdoor and sparse point clouds offering limited information.

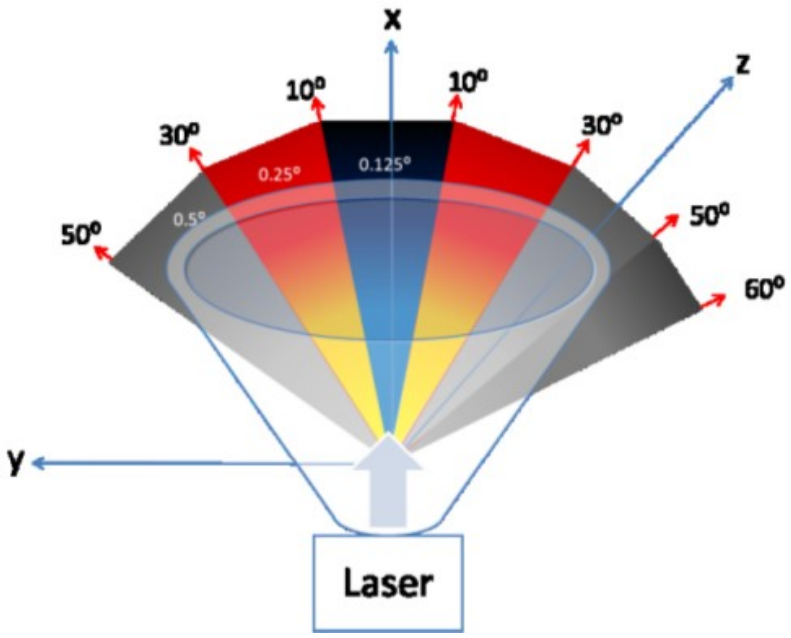


Fig. 34 Variable angular resolution in the Sick LD-MRS laser scanner.

The SICK LD-MRS laser scanner used offers several scanning frequencies with different angular resolution. The smallest frequency, 12.5 Hz, with variable angular resolution between 0.125° in front of the vehicle, 0.25° between the 10° and the 30° and 0.5° between 30° and 50° (60° in the right side of the scene) as seen in figure 34. This configuration increases the ability for long range detection in front of the vehicle, where obstacles tend to be further. For automotive applications, lower resolutions in the sides are acceptable, as the obstacles of our concern are closer than in the front and will be represented by many points even at lower resolutions. Distances between measured points in Y explain the need for adaptation of cluster threshold according to the distance from the obstacle to the laser, in order to obtain the most populated possible clusters. The meaning of the values in Table 3 are: Y_S is the width of the measured point, Y_G1 is the distance between measured points in one measurement plane, Y_G2 is the distance between measured points between two laser pulses, and X_{layer} is the height of the measured point.

Table 3 Distances between measured points at angular resolutions of 0.125 degrees

Dist (m)	Y_S	Y_G1	Y_G2	X_{layer}
10	0.014	0.029	0.007	0.139
25	0.035	0.074	0.019	0.349
50	0.069	0.148	0.039	0.698
100	0.139	0.296	0.078	1.396

Adapted Euclidean Distance and geometrically constrained clusters

In this approach, a classical Euclidean distance clustering strategy has been adopted, modulated by several parameters in order to modify the clustering behavior, such as distance from the sensor to the obstacle, geometrical constraints, allowed number of points in every cluster, etc. Additionally, some parameters used in the clustering process such as maximum distance between candidate points, are modified according to the shapes detected in the point cloud near to the cluster, to improve oblique obstacle detections. An alternative strategy has been tested, using Mahalanobis distance, as the normalized Euclidean distance from the cluster's centroid to candidate points. This method tends to obtain compact clusters and ignores increasingly further points belonging to oblique obstacles. Taking into account that our scenario will produce small clusters, that is the reason why it has been discarded. In this approach, clusters are defined as the set of points separated a certain distance, which varies as a function of several parameters, plus some points that does not meet the distance requirements, but some geometric constraints, such as belonging to the same line in the space than some of the points in the cluster.

Fig. 35 IVVI 2.0 research platform with axis represented in the image: X=laser-obstacle distance, Z=detection height, Y=horizontal deviation from the laser.

The strategy is defined as an iterative addition of points to the cluster with the following steps:

1. First point in the point cloud is taken as the first point in the cluster.
2. All the other points in the point cloud are checked to have a distance smaller than the cluster threshold $ClusterTh$

$$
\begin{aligned}
&ClusterTh = BaseTh + DistCorr(x)\\
&DistCorr(x) = \sqrt{(x*tan(\alpha_y))^2 + (x*tan(\alpha_z))^2}\\
&if \quad |arctan\left(\frac{y}{x}\right)| < 2\pi\frac{10}{360} \quad then \quad \alpha_y = 2\pi\frac{0.125}{360}\\
&if \quad 2\pi\frac{10}{360} \leq |arctan\left(\frac{y}{x}\right)| < 2\pi\frac{30}{360} \quad then \quad \alpha_y = 2\pi\frac{0.25}{360}\\
&if \quad 2\pi\frac{30}{360} \leq |arctan\left(\frac{y}{x}\right)| < 2\pi\frac{60}{360} \quad then \quad \alpha_y = 2\pi\frac{0.5}{360}
\end{aligned}
\tag{1}
$$

where BaseTh is a parameter experimentally determined as the base threshold. DistCorr(x) is a function of the x coordinate which ensures that no distance smaller than the minimum physically possible distance will be required, as seen in equation 1, and depending on the different angular resolutions seen in figure 34. DistCorr(x) is computed as the minimum distance possible between two consecutive points in z and y coordinates. α_y Represents the angle between two consecutive laser reads in horizontal (y axis) and α_z is the angle between two consecutive laser reads in vertical (z axis).

3. All the points in the point cloud are checked for cluster inclusion. The same iteration is performed for every point added to the cluster until all cross checks are performed. Then, points close to the obstacle but not belonging to the cluster are included into a temporary new point cloud together with the obtained cluster, and then lines are searched in the new cluster using RANSAC. If lines are found containing a determined minimum of points belonging to the original cluster and points not belonging to it, then these points are added to the cluster. This strategy has proven to be effective for oblique obstacles.

Figure 36 shows the result of the algorithm. Red dots are the cluster created by Euclidean Adapted distance. Blue dots are the points close to the cluster but not belonging to it. Yellow lines are 3D lines found by RANSAC, including points from the original cluster and points from the extended cluster.

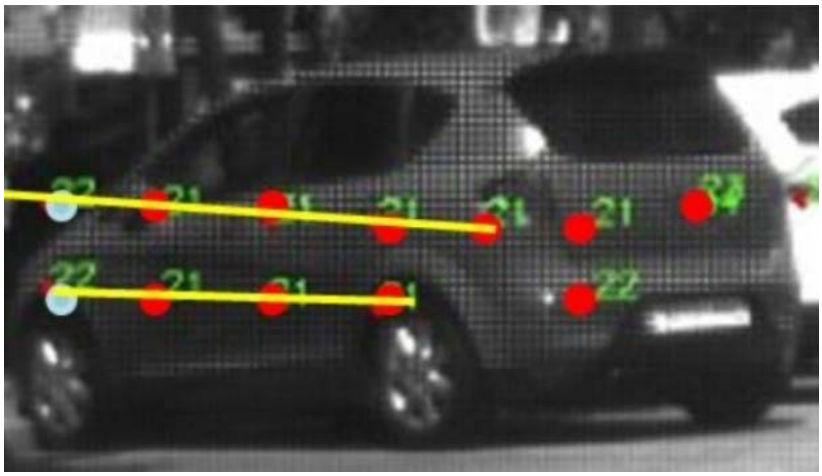


Fig. 36 Extended cluster using geometrical constraints.

Upon completion of cluster extraction, it is checked against the parameters *ClusterTolerance* for maximum width of cluster in meters, and *minClusterSize* and *maxClusterSize* for minimum and maximum number of points, respectively. These parameters are also a function of the distance to the obstacle.

The strategy is addressed to obtain the most populated clusters possible, taking into account that a low resolution multilayer laser is being used. The threshold distance must be adapted to the distance x from the laser sensor to the obstacle, as the distance between consecutive laser points grows with x. Due to laser construction limitations, the minimum distance detected in y and z in consecutive points will be greater than the initial threshold if not adapted following equation 1.

4.1.1 Ground detection and removal from point cloud

As seen in the data fusion chapter, the presented system can compute the plane corresponding to the road surface, so it is possible to remove ground plane points from the list of detected clusters. Figure 37 shows the result of the algorithm, ignoring as cluster candidates all the points located in the ground plane obtained with RANSAC. These points might match the geometrical

constraints for cluster creation as if they were obstacles, but they are ignored as also match the constraints for ground plane belonging.

Fig. 37 Cluster removal in ground plane. Points in the bottom of the image meet the geometrical constraints for cluster creation, but are omitted because of ground plane belonging.

4.2 Obstacle classification using laser information

Obstacle classification using laser information is a challenging task, as the amount of available information is very scarce. The presented thesis uses different levels of complexity depending on the stage of classification, explained in the next sections.

The first step in obstacle classification for this thesis is laser scanner based classification between Vulnerable Road Users (VRU), that is, pedestrians and bicyclists, and vehicles, namely cars and motorbikes. Figure 38 shows the process. After initial laser based classification, ROIs are extracted and an

image based classification is performed for pedestrian or bicycle detection in case of VRU, and cars or motorcycles in case of Vehicles.

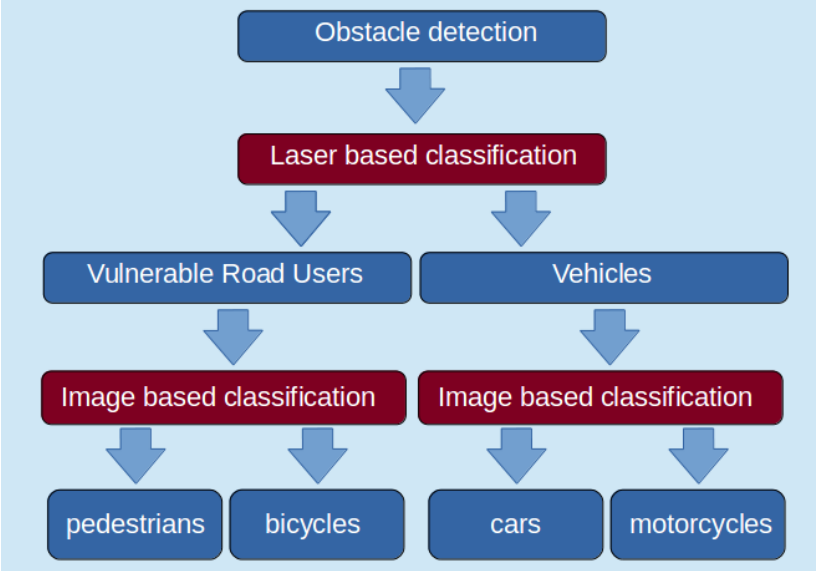
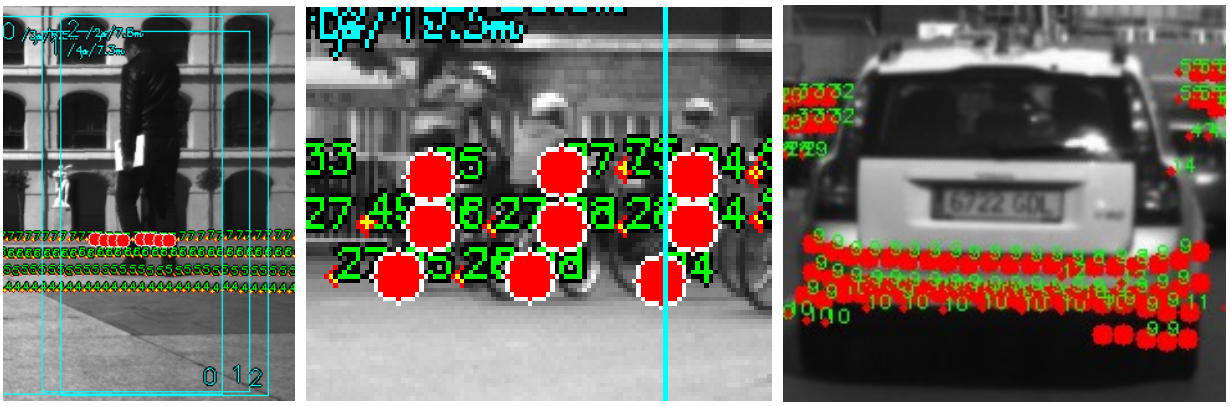


Fig. 38 Obstacle classification process. Initial obstacle detection, Laser classifies between VRU and vehicles and generates ROIs for image classification. CV classifies VRU between pedestrians and vehicles, and vehicles between cars and motorcycles.

4.2.1 Morphological classification

Some morphological constrains can be applied depending on the specific expected shapes of the Objects Of Interest. In the case of pedestrians, (figure 39a) the constraints are based on the model of dressed human body [19] as an ellipse of 60cm x 50cm. Bicycles, from a laser point of view, are considered as pedestrians, because the structure of the vehicles is very seldom detected, as shown in figure 39b. Cars are considered as rectangular objects with regular faces and dimensions varying from 1.5 up to 10 meters (figure 39c).



(a) Pedestrian (b) Bicycle (c) Car

Fig. 39 Morphological characteristics of different cluster representations of objects of interest.

Figure 40 shows the distribution of the cluster size in meters in a set of study. Positive clusters are pedestrians, obtained from the LSI-CROMA recording with the LD-MRS laser scanner. Negative (non pedestrian) samples are cars, bicycles, motorbikes and diverse objects in sideways and roads, such as walls, trees, poles, mailboxes, etc, obtained from diverse recordings in the IVVI 2.0 platform using the same laser scanner.

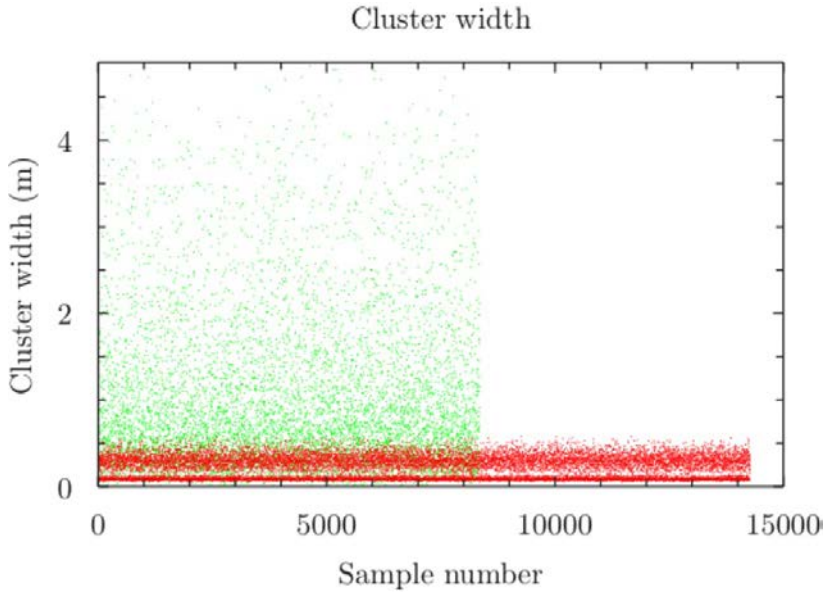


Fig. 40 Distribution of cluster width in pedestrian/no pedestrian obstacles. Red dots are pedestrian clusters, green dots are not-pedestrian clusters.

Figure 41 shows the distribution of the cluster size in meters in a set of study. Negative (non car) samples are pedestrians, bicycles, motorbikes and diverse objects in sideways and roads, such as walls, trees, poles, mailboxes, etc. It is interesting to note that most of the obstacles found not being a car are pedestrian size, a small proportion of obstacles car size, and a bigger proportion of high sized obstacles, usually representing walls. The maximum size of the cluster considered has been set to 6 meters. Car samples are obtained from recordings using the IVVI 2.0 platform in parking lots, pedestrians for negative samples are obtained from the LSI-CROMA recording session, and other obstacles are extracted from recordings using the IVVI 2.0 in urban and road scenarios.

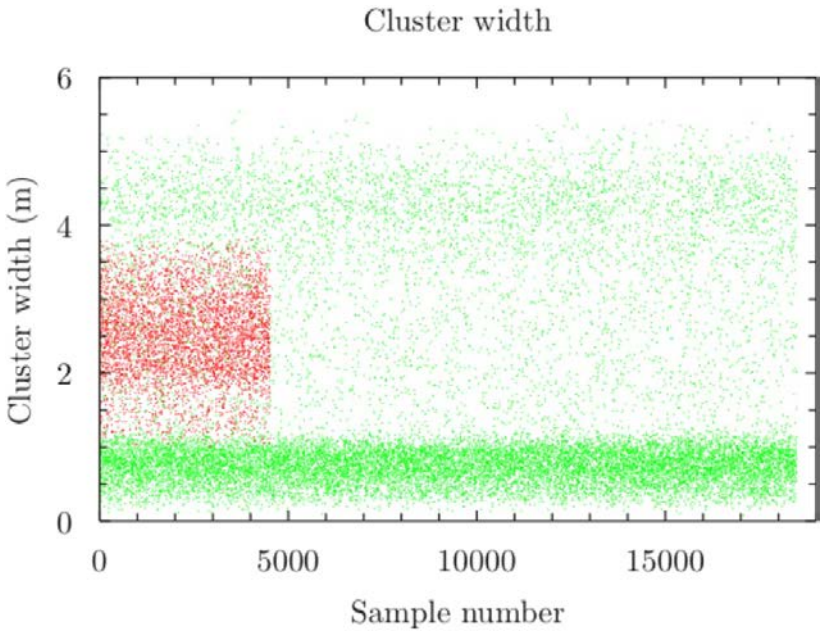


Fig. 41 Distribution of cluster width in car/no car obstacles. Red dots are car clusters, green dots are not-car clusters.

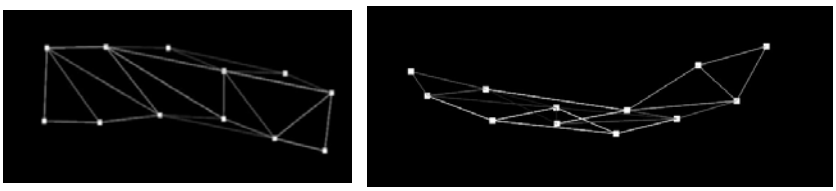
4.2.2 SVM classification

SVM classification is performed using the SVM implementation from the Computer Vision OpenCV library. SVM algorithm was developed by Vapnik and Cortes in [89] and is widely used in machine learning as a classification

method. SVM computes features from the positive and negative samples used for training. In images, HOG features and LBP features are common, but laser point clouds are not suitable per se for SVM classification. A number of mathematical features have been defined in order to extract from the point clouds the required information about the shape ans characteristics of the detected obstacle, and to keep the information in a fixed size regardless the size of the point cloud, as required for SVM training and classification.

Laser scanner clusters feature vector

Clusters detected in laser scanner generated point clouds are used to determine a Region of Interest in the image where we can perform obstacle classification applying Computer Vision and Artificial Intelligence techniques, but can also be used for obstacle classification without image support [12]. Clusters are converted into a mesh structure by Delaunay triangulation in order to reconstruct the shape of the obstacle and to extract relevant features according to the 3d shape of the cluster, as seen in figure 42. The mesh can be represented from any point of view; figure 42a represents the view from the laser, and 42b represents a aerial view of a point cloud.



(a) Frontal view (b) Aerial view

Fig. 42 Mesh representation of a cluster.

These obstacles are detected by the system as clusters, which have some characteristics suitable for further SVM training following the process outlined in figure 43. Clusters obtained from the test sequences are stored and manually labeled using the corresponding images for training. These clusters

are manually labeled as frontal view, back view, side view, frontal oblique view and back oblique view.

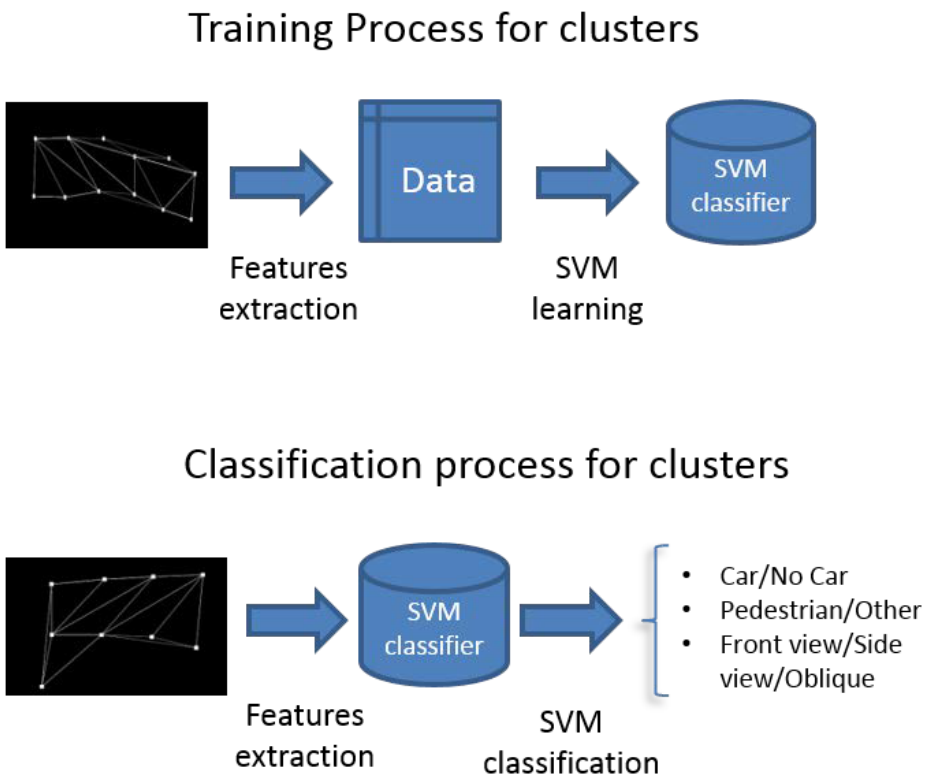


Fig. 43 SVM learning process for clusters: Training and classification.

Previous works as [14] have considered 2D point clouds for classification, but the present work is intended to extract features from a 3D point cloud, in an effort to maximize the use of the available information. Some of the features considered are described in Table 4.

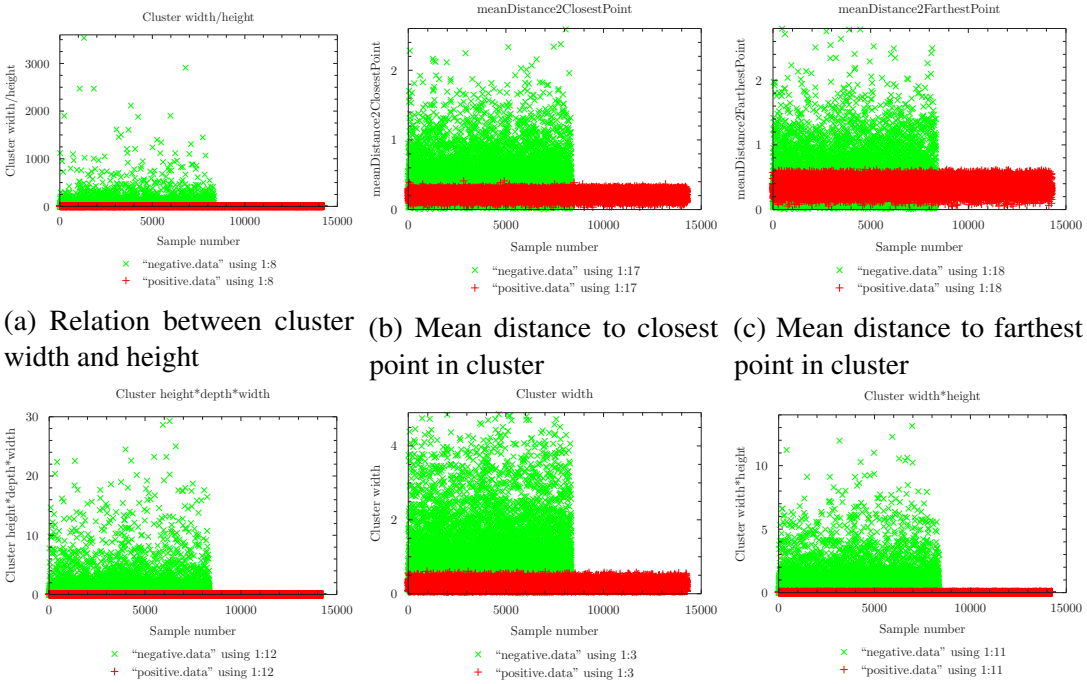
Table 4 Some of the features considered for cluster classification.

Feature	Meaning
Concentration	Normalized mean distance to the centroid 3D
Y-Z concentration	Normalized mean distance to the centroid excluding x
X-Z concentration	Normalized mean distance to the centroid excluding y
X-Y concentration	Normalized mean distance to the centroid excluding z
Flatness	Normalized mean distance to the most populated plane found in the cluster
Sphericity	Normalized mean distance to the most populated sphere
Cubicity	Measures how far are the planes containing the mesh triangles from being the same plane or from being perpendicular
Triangularity	Measures the uniformity of the triangles composing the mesh bye the relation between sides' lengths
Average deviation	Average deviation from the median in x, y, z

A study of the relevance of every feature considered has been performed, using a set of training of 14,000 clusters representing a pedestrian and 8,400 clusters representing several kind of non-pedestrian obstacles. It is important to use only features that help to differentiate between positive and negative existence of the Object Of Interest. A similar study has been performed for car, bicycles and motorcycles clusters database used for classifier training, in order to select the appropriated features for each kind of obstacle.

In figure 44, several good features are studied. The horizontal axis indicates the number of sample considered, and the vertical axis represents the magnitude of the feature. Red crosses represent the value of the positive samples, while green crosses are the values of the negative samples. The features describing well the difference between positive and negative samples present a high concentration of positive magnitudes, very different from the negative magnitudes, as shown in figure 44e, *cluster width*, with the positive values very concentrated near zero, and negative magnitudes concentrated higher

than zero. The rest of the examples in figure 44 show highly differentiated values for positive and negative samples.



(a) Relation between cluster width and height

(b) Mean distance to closest point in cluster

(c) Mean distance to farthest point in cluster

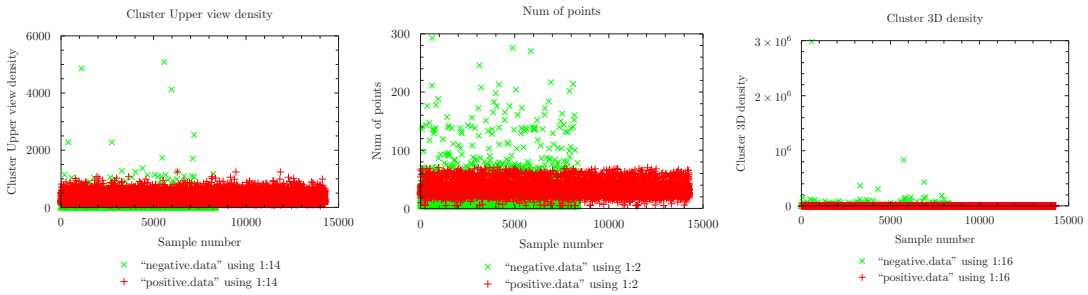
(d) Cluster height * depth * width

(e) Cluster width

(f) Cluster width * height

Fig. 44 Distribution of the values of a feature describing well a cluster characteristic.

Figure 45 represent statistics for features describing poorly the difference between positive and negative samples. Figure 45a shows that the feature *Cluster density from upper view* does not define well the difference between positive and negative samples, as most of the positive and negative values for that particular feature are coincident.



(a) Cluster density from upper view (b) Number of points in cluster (c) Cluster 3D density

Fig. 45 Distribution of the values of a feature describing poorly a cluster characteristic.

Chapter 5

Obstacle detection and classification using computer vision

After the initial laser-based obstacle detection and classification stage, obstacles represented as clusters in the laser scanner Point Cloud are translated into Regions Of Interest (ROI) in the image, where Objects Of Interest (OOI) are searched using Computer Vision algorithms, as will be shown in the present chapter. Although public datasets are available for pedestrians and cars, like INRIA, ETH, TUD-Brussels, Daimler, Daimler stereo, Caltech-USA and KITTI, experience shows that best results for classification are achieved when the same camera and in the same position is used for training and for classification. Having this goal in mind, several datasets have been created for LSI using the XB3 camera.

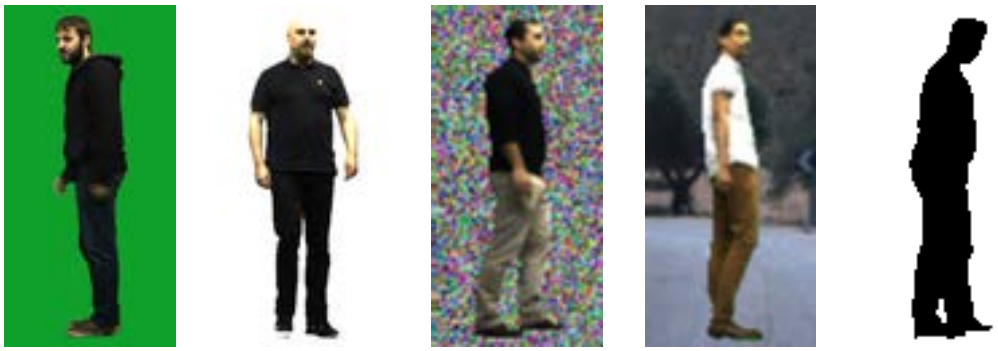
5.1 LSI Datasets

Using computer vision for obstacle classification requires a dataset of positive and negative samples of the Objects Of Interest (OOI). Several public dataset are available, but better results are expected if a dataset obtained from the same sensors, located in the same position of the vehicle is used. For this reason,

image datasets for pedestrian and bicycles have been created at Intelligent Systems Lab (LSI) as part of the present thesis.

5.1.1 LSI-CROMA pedestrian training set

The intention in the making LSI-CROMA is to evaluate the possibility of using synthetic datasets created from extracted pedestrians inserted in different backgrounds, as a source for machine learning training in pedestrian detection applications. Subsequent addition of small sets of samples will be tested in order to check the impact in the accuracy of the pedestrian detection.
A synthetic pedestrian training set, called LSI-CROMA, has been created at Intelligent Systems Lab (LSI) for research and testing purposes. LSI-croma is a training set for computer vision based pedestrian classification in images. Several sets of images with different manipulations are included for testing purposes. A synthetic test set has also been included with annotations of the position of the inserted pedestrians and its identity and pose.
Each training set contains almost 9,000 positive samples (mirrored images are not included, so another 9,000 samples can be obtained just by mirroring the originals)
The training set includes 18 different pedestrians wearing different clothes, recorded from every possible angle. From the whole set of 18 pedestrians, 10 have been devoted for training, and 8 for testing.



(a) Green background (b) White background (c) Inserted noise (d) Inserted background (e) Black and white mask

Fig. 46 Example of the images provided in 64x128, 128x128 and 128x256 pixels.

Image manipulation

Four versions of the images with inserted background or noise are supplied, depending on the treatment applied to the subject and to the background, as seen in figure 47.

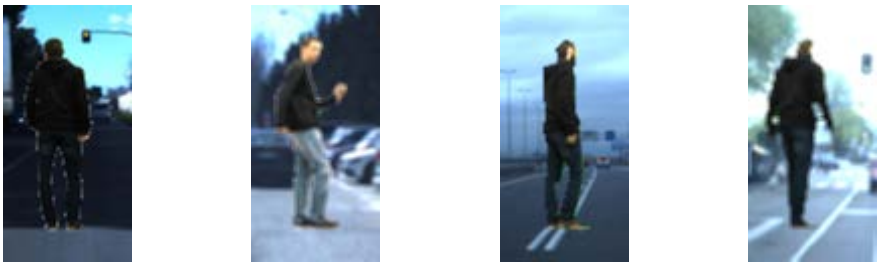
1. Blur2x2 : A 2x2 kernel Gaussian Filter has been applied to both the pedestrian and the background for blurring. These images still present a small trace of the white croma around the pedestrian (figure 47a).

2. Blur3x3:A 3x3 kernel Gaussian Filter has been applied to both the pedestrian and the background for blurring. These images still present a small trace of the white croma around the pedestrian (figure 47b).

3. Eroded2x2: An erosion using a 2x2 kernel has been applied to the pedestrian, then the pedestrian is inserted into the background and then a 2x2 kernel Gaussian Filter has been applied to both the pedestrian and the background for blurring. These images do not keep any trace of the white croma around the pedestrian (figure 47c).

4. Eroded3x3: An erosion using a 3x3 kernel has been applied to the pedestrian, then the pedestrian is inserted into the background and then

a 3x3 kernel Gaussian Filter has been applied to both the pedestrian and the background for blurring. These images do not keep any trace of the white croma around the pedestrian (figure 47d).



(a) Blur2x2 (b) Blur3x3 (c) Eroded2x2 (d) Eroded3x3

Fig. 47 Samples of the four different versions provided, in 64x128, 128x128 and 128x256 resolutions

Neutral backgrounds for pedestrian insertion

Almost 500 varied background images are provided with road, country and urban scenes. Several resolutions are also provided in the backgrounds set, although each scene has only one resolution. An example of the background set can be seen in figure 48

Fig. 48 Sample of the background set in LSI-CROMA

Testing sets

Every available background in the set has been treated for three random pedestrian insertions in random locations of the image as seen in figure 49. The test images have been annotated with the coordinates of the square surrounding the pedestrian and the name of the file corresponding to the pedestrian. The annotation file is in the same folder and its name is the name of the .png file with a ".txt" suffix. An example of an annotation file is shown in table 5.

Table 5 Example of the contents of an annotation file.

Xsup	**Ysup**	**Xinf**	**Yinf**	**Pedestrian file name**
180	338	308	594	pedestrian3256.png
456	321	584	577	pedestrian3473.png
822	381	950	637	pedestrian0236.png

These test files are treated the same way the images are, so there are four available set of images (blur2x2, blur 3x3, eroded2x2, eroded3x3).

Fig. 49 Test image blurred with a 2x2 kernel and annotation file. The annotation file would contain:
157 406 285 662 LSICromaGreen0094.png
528 460 656 716 LSICromaGreen0055.png
780 345 908 601 LSICromaGreen0186.png

5.1.2 LSI-CROMA making

The complete recording of images for LSI-CROMA was performed at Universidad Carlos III de Madrid facilities. 18 different pedestrians were recorded from every possible point of view, before a green croma screen. The pedestrians mimicked the normal walking gestures of a pedestrian, while turning around slowly without moving away from a mark in the ground. The recording lasts two minutes for each pedestrian and delivers around 400 images with every possible angle and walking gesture.
The individual images are then processed for background color homogeneity, as the original croma picture does not offer a pure green background color. An example of the processed image can be seen in figure 50c. These images are then searched from left to right, right to left, top to bottom and bottom to left, in order to find the position of the pedestrian in the image, as seen in figure 50b. This cropped image is then resized and padded with 8 pixels on top and bottom, and the appropriated padding in the sides to keep the aspect

ratio and get a 64x128 pixels image, as shown in figure 50c. In order to keep the aspect ratio and place the pedestrian always in the same position of the training samples, eight pixels of background are always allowed on top and bottom of the sample, re dimensioning the crop to 112 (i.e. 128-16) pixels height and filling with background the rest of the sample until the 128x64 pixels size following algorithm (2).
Similar computation is performed for 64x64 and 128x256 pixel images, and for white background instead of green background generation.

$$\begin{aligned} upBottomMargin &= 8 \\ newHeight &= \frac{originalHeight * 128}{64 - upBottomMargin * 2} \\ newWidth &= newHeight/2 \\ newYSup &= ySup - |\frac{originalHeight - newHeight}{2}| \\ newXYSup &= xSup - |\frac{originalWidth - newWidth}{2}| \end{aligned} \tag{2}$$

At this time, top left point of the image, width and height of the crop are obtained. Crop is performed using these parameters, and then image is resized to 64x128 pixels.

(a) Original image after green background homogenization.

(b) Image after cropping.

(c) Final image after padding addition.

Fig. 50 Image processing from original to final.

In order to extract the background for synthetic image generation, random parts of the background files with the appropriated size are selected, and only the pedestrian from the green set is inserted, after the following processing:

As seen previously, a different set of images is created for each of the following pedestrian processing:
Four versions of the images with inserted background or noise are supplied, depending on the treatment applied to the subject and to the background, as seen in figure 47.

1. Blur2x2 : A 2x2 kernel Gaussian Filter has been applied to both the pedestrian and the background for blurring. These images still present a small trace of the white croma around the pedestrian (figure 47a).
2. Blur3x3:A 3x3 kernel Gaussian Filter has been applied to both the pedestrian and the background for blurring. These images still present a small trace of the white croma around the pedestrian (figure 47b).
3. Eroded2x2: An erosion using a 2x2 kernel has been applied to the pedestrian, then the pedestrian is inserted into the background and then a 2x2 kernel Gaussian Filter has been applied to both the pedestrian and the background for blurring. These images do not keep any trace of the white croma around the pedestrian (figure 47c).
4. Eroded3x3: An erosion using a 3x3 kernel has been applied to the pedestrian, then the pedestrian is inserted into the background and then a 3x3 kernel Gaussian Filter has been applied to both the pedestrian and the background for blurring. These images do not keep any trace of the white croma around the pedestrian (figure 47d).

Additional sets are created with random noise background, as seen in figure 46c and a black and white silhouette set as shown in figure 46e.

5.2 LSI-BICYCLES

Several hours of cyclists images were recorded using both laser scanner and stereo camera. At a later stage, sensor information were fused in order to align data, and all the images from the recording were stored. While the aspect ratio for pedestrian samples is considered to be vertical (64x128 pixels) and aspect ratio for car samples is usually considered horizontal (128x64 pixels), the aspect ratio for bicycles varies from frontal and lateral view and has been set to 64x64 pixels.

Bicycles in the images have been manually labeled, cropping the image as tight as possible to the bicycle. In order to keep the aspect ratio and place the bicycle always in the same position of the training samples, eight pixels of background are always allowed on top and bottom of the sample, re dimensioning the crop to 48 pixels height and filling with background the rest of the sample until the 64x64 pixels size following algorithm 3.

$$
\begin{aligned}
&upBottomMargin = 8 \\
&newHeight = \frac{originalHeight * 64}{64 - upBottomMargin * 2} \\
&newWidth = newHeight \\
&newYSup = ySup - |\frac{originalHeight - newHeight}{2}| \\
&newXYSup = xSup - |\frac{originalWidth - newWidth}{2}|
\end{aligned}
\tag{3}
$$

At this time, top left point of the image, width and height of the crop are obtained. Crop is performed using these parameters, and then image is resized to 64x64 pixels.

(a) Original image. 640x480 pixels.

(b) Intermediate bicycle tight cropping. 130x265 pixels.

(c) Final padded training sample. 64x64 pixels.

Fig. 51 Original bicycle image and cropping process

More than five thousand positive samples of bicycles from every point of view have been selected for system training.

5.3 System Training

In every of the training sets, the system is trained initially using exclusively the positive samples. Once computed the classifier, an iterative procedure known as bootstrapping is executed, in which a set of images not containing any OOI (cars, pedestrians or bicycles, depending the case) is searched for false detections of OOI . Each one of these false positives is then added to the negative samples and the system is retrained until a certain threshold of false positives is reached. In our system, more than 5,000 positive images of bicycles and more than 27,000 negative images are used. Table 6 shows the figures for the three datasets generated.

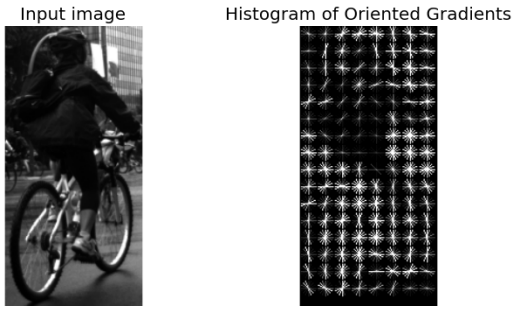
Table 6 Number of samples per dataset generated.

Sample/OOI	Cars	Pedestrians	bicycles
Positive samples	3,000	9,900	5,000
Negative samples	10,000	30,681	27,000

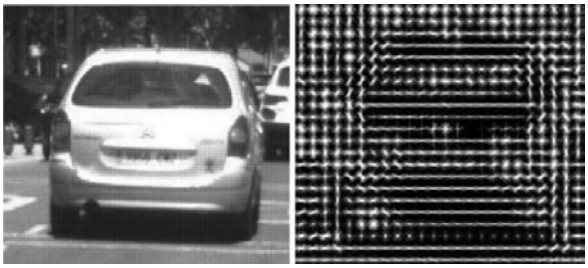
In the presented work, classification has been performed using four different approaches, depending on the descriptors considered in the image and the classification algorithm.

Image Features

1. HOG Descriptor. HOG is a holistic descriptor that captures the general shape of the image through the information about the gradients and its orientation. HOG is well suited for human classifiers and is commonly used in CV. As cyclist detection involves human body detection, HOG is likely to be a good choice as descriptor, even though it requires a high computational effort. Figure 52 shows original images of a bicycle and a car and the resultant HOG descriptors. Figure 53 shows the global HOG descriptor extracted from the training for positive and negative images in the bicycle dataset.



(a) Histogram of Oriented Gradients representation (HOG) in a bicycle image. Left, original image. Right, HOG representation.



(b) Histogram of Oriented Gradients representation (HOG) in a car image. Left, original image. Right, HOG representation.

Fig. 52 Histogram of Oriented Gradients representation (HOG)

2. Local Binary Patterns (LBP) Descriptor. LBP is a descriptor widely used in Computer Vision that describes the texture of the image and is invariant to monotonic changes in the gray level and to translation. In the presented work, LBP is combined with HOG in in order to improve detection performance.

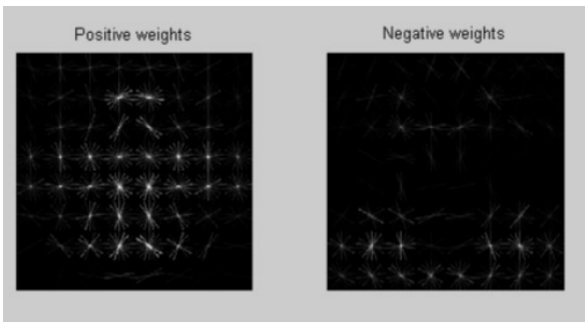


Fig. 53 Global HOG descriptor obtained after training.

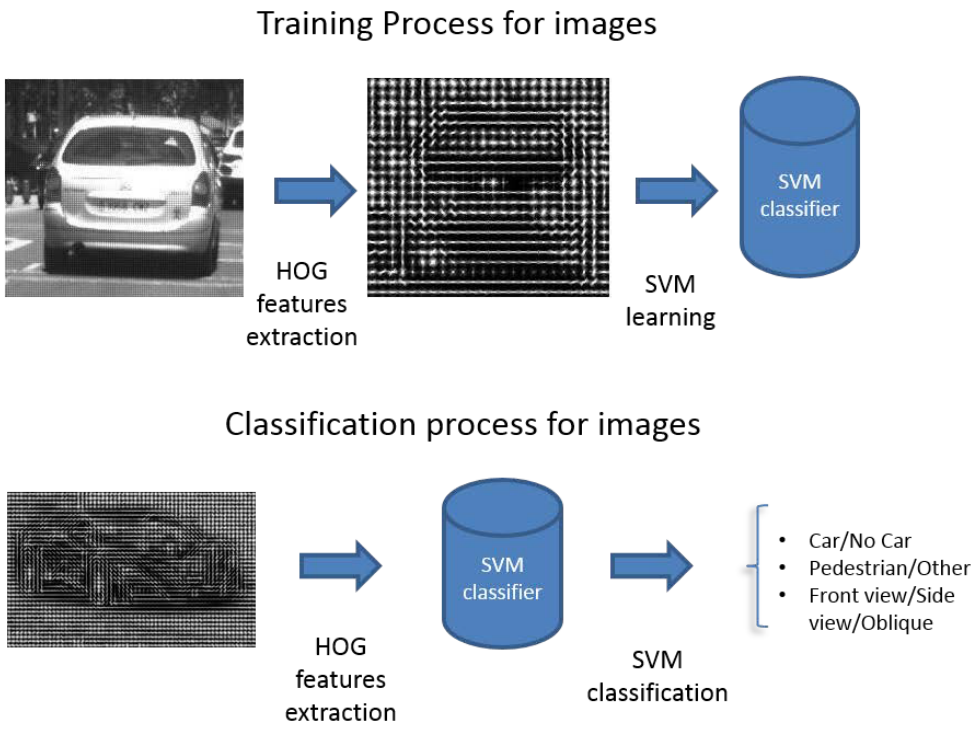


Fig. 54 SVM learning process for images: Training and classification.

Classification algorithm. Support Vector Machines

Support Vector Machines (SVM) is a supervised classification method that intends to find a hyperplane able to separate a set of samples into two different subsets, positive and negative. SVM is a discriminative learning method, classifying the samples without the use of probability density functions in the classes, unlike the generative models [90].

The classifier frontier is obtained from a representative training set of samples in the form of a n-dimensional vector of characteristics and a label indicating the class (+1,-1) of the particular sample.

SVM is a linear classifier, so the searched solution will be a line, a plane or a hyperplane, depending on the dimensions of the data to classify. Only a limited subset of the samples, called support vectors, will be taken into account for the generation of the frontier. These support vectors will be

chosen so that the distance, called margin, between the planes containing them in each of the classes is maximum. The maximum margin is the solution searched, and represents the maximum portion of the space free of support vectors. The middle plane of this margin, called frontier, is the solution of the SVM. In figure 55, support vectors of the green class and from the red class are absent between the dashed lines, so this is the margin. Blue line is the frontier, this is, the solution for the SVM.

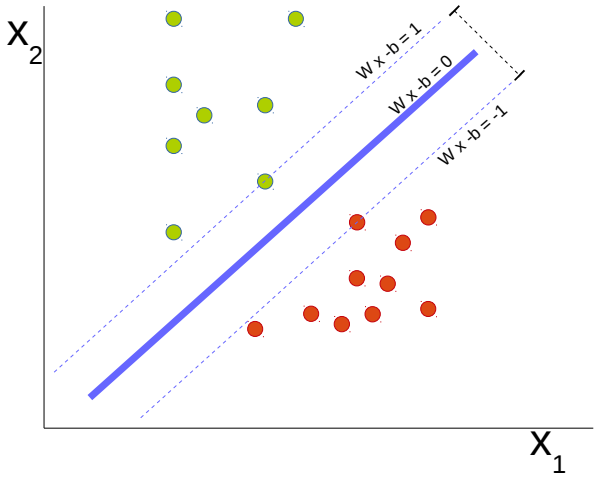


Fig. 55 SVM application. The margin is the portion between the dashed lines, and the frontier is the blue line.

A problem to avoid is overfitting. This is, finding an optimal solution for the particular sets used for training, but at the expense of generalization. The solution will be good for the training set, but will fail for other sets. A graphical example is shown in figure 56, where the solution is very fitted to this particular set. The use of support vectors guarantees a solution not determined by samples outside of the margins delimited.

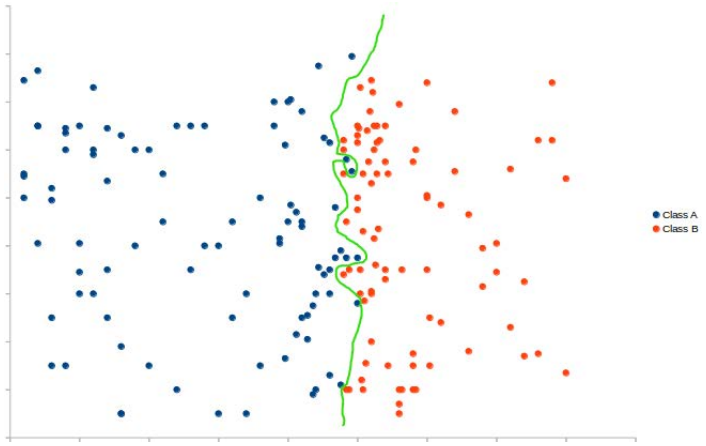


Fig. 56 Overfitting. The frontier is too fitted to the set and might not generalize well for other sets.

Most of the real sets are not perfectly lineally separable into two classes, as some overlapping samples exists between them. Soft margin allows a relaxation of the constraints, so some of the samples might be located inside the margin or even in the space belonging to the opposite class. Clearance parameters adjust the permissiveness to outliers not meeting the constraints. In figure 57 an example is shown. some red samples are inside the margin or even after the frontier. Some green samples are inside the margin and some even after the opposite margin. Nevertheless, the classifier can work and is not overfitted.

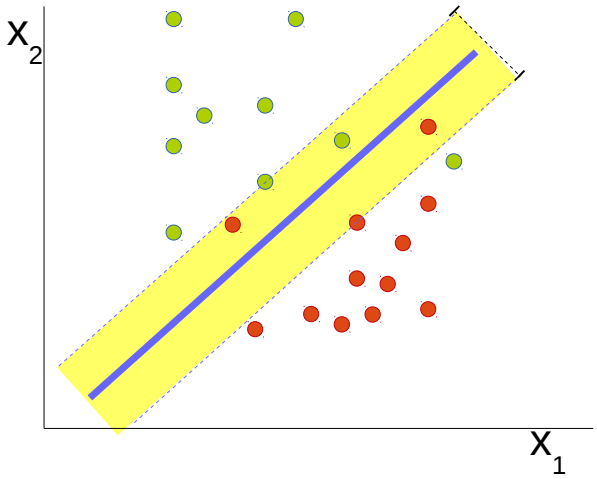


Fig. 57 Soft margin. Some samples are located outside or the delimited margins.

5.4 Results

Several training and test sets have been used in order to exploit the capabilities of the LSI-CROMA set and compare performances.
The set used are:

- LSI-Croma.

  9,800 Pedestrian images extracted from a croma background, inserted behind a real world urban, road of urban background. LSI-croma is described early in this thesis. Figure 58a shows an example of the images in this dataset.

- INRIA pedestrian dataset.

  2,415 Pedestrian images provided by INRIA. Figure 58b shows an example of the images in this dataset.

- Kitti pedestrian dataset.

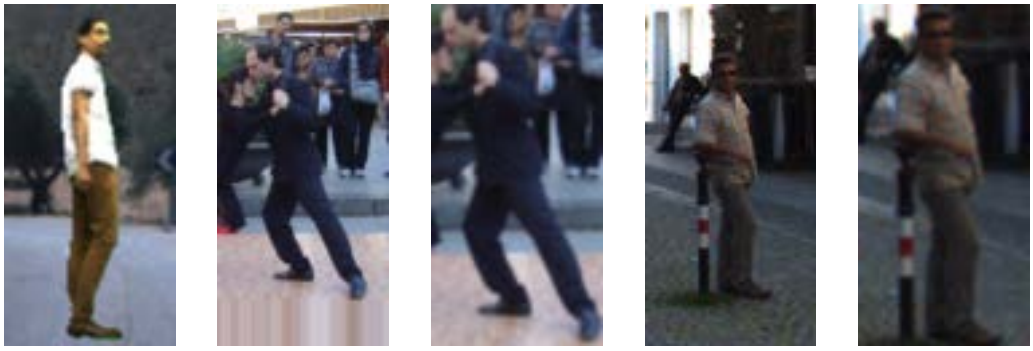
  2,010 Pedestrian images provided by the KIT in the Kitti pedestrian dataset. Figure 58d shows an example of the images in this dataset.

- INRIA+Croma dataset.

  This dataset includes the 2,415 images in the Inria dataset, enriched with 2,711 pedestrian images from the Croma dataset for a total of 5,126 images. Experiments were made using the Inria images in its original state and resized to fit the same proportions than in the Croma dataset. An example of the resized images is shown in figure 58c

- Kitti+Croma dataset.

  This dataset includes the 2,010 images in the Kitti dataset, enriched with 2,711 pedestrian images from the Croma dataset for a total of 4,721 images. Experiments were made using the Kitti images in its original state and resized to fit the same proportions than in the Croma dataset. An example of the resized images is shown in figure 58e.



(a) Example of Croma pedestrian dataset

(b) Example of Inria pedestrian dataset

(c) Example of Inria pedestrian dataset resized to fit Croma proportions

(d) Example of Kitti pedestrian dataset

(e) Example of Kitti pedestrian dataset resized to fit Croma proportions

Fig. 58 Example of the pedestrian datasets used for training and testing in the thesis

The procedure followed for testing is as follows:

1. A classifier is extracted for each of the aforementioned training sets, using 70% of the positive samples for training, and keeping the rest for testing purposes. The negative samples are the provided by the dataset maker.

2. In an iterative process, predictions looking for positive detections are made in the remaining 30% of the positive samples allocated as a testing set, and search for false positives is performed in a set of images not containing any pedestrian. In each of the iterations, the confidence parameter is changed from a set of 15 values, from 0.0 to 3.0, in order to extract information about the evolution of the classifier and to determine the best confidence value for real use. This full process is performed in each of the testing datasets and for each of the classifiers obtained in the previous step.

3. Data from predictions are classified into True Positives, False Positives, True Negatives and False Negatives (see figure 59), and then the performance statistics are computed for each of the training sets and classifiers.

4. Performance statistics are compiled and graphics are generated.

The performance parameters used are derived from the confusion matrix shown in figure 59 and are explained next:

- Accuracy/Threshold charts.

  The Threshold indicator represents the certainty of the detection, and Accuracy is a performance indicator computed as

$$Accuracy = \frac{TruePositive+TrueNegative}{TruePositive+TrueNegative+FalsePositive+FalseNegative}$$

- Precision/Recall charts

  The precision/recall charts indicate the compromise between Precision and Recall and its evolution.

  - Precision. Is the ratio between the number of true detections over the number of positive annotations.

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{4}$$

  - Recall. Is the ratio between the number of True Positives over the number of negative annotations.

$$Recall = \frac{TruePositives}{FalsePositives + FalseNegatives} \tag{5}$$

- ROC curve

  ROC curves model the detections of true positives and false negatives, and has better statistical foundations than other measures. Y axis represents the True Positive Rate or Recall (see equation 5), while X axis is computed as

$$FalsePositiveRate = 1 - \frac{TrueNegative}{FalsePositive + TrueNegative} \tag{6}$$

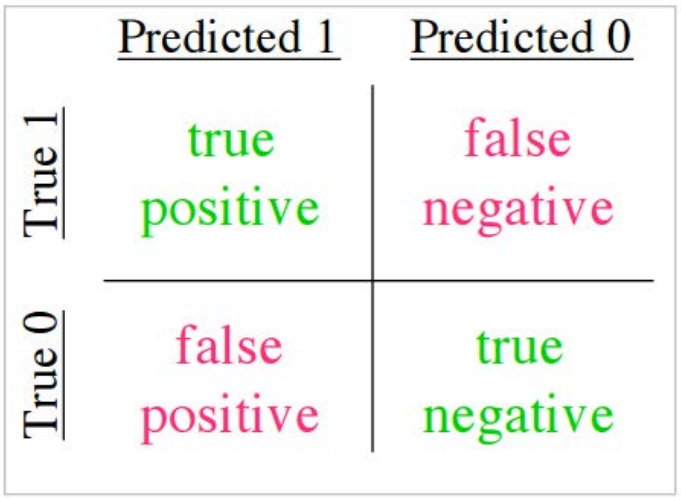


Fig. 59 Confusion Matrix

The data obtained are explained in the following graphs.

Figure 60 displays the evolution of the accuracy with different values of threshold of detection, detecting in the Croma testing set. Predictions made with a classifier in the same training set are usually expected to be accurate, as in this case. As the dataset used is synthetically generated, unexpected results could be obtained for predictions from real world classifiers, but results are coherent, being Croma, Kitti and Kitti+Croma the best classifiers for this testing dataset.

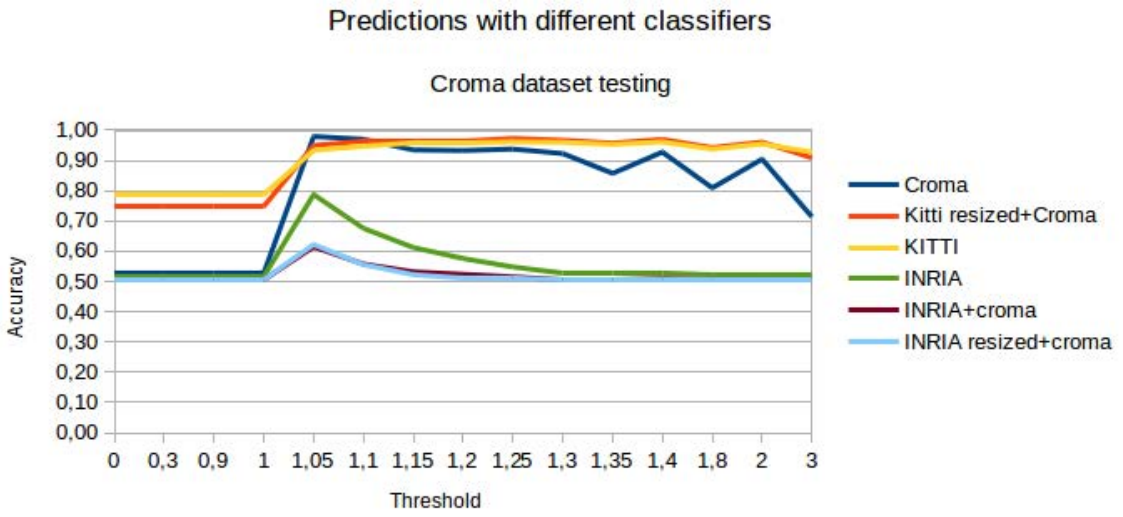


Fig. 60

Figure 61 illustrates predictions with different classifiers in the Croma testing dataset. The Croma, Kitti and Croma+Kitti classifiers obtain the best results.

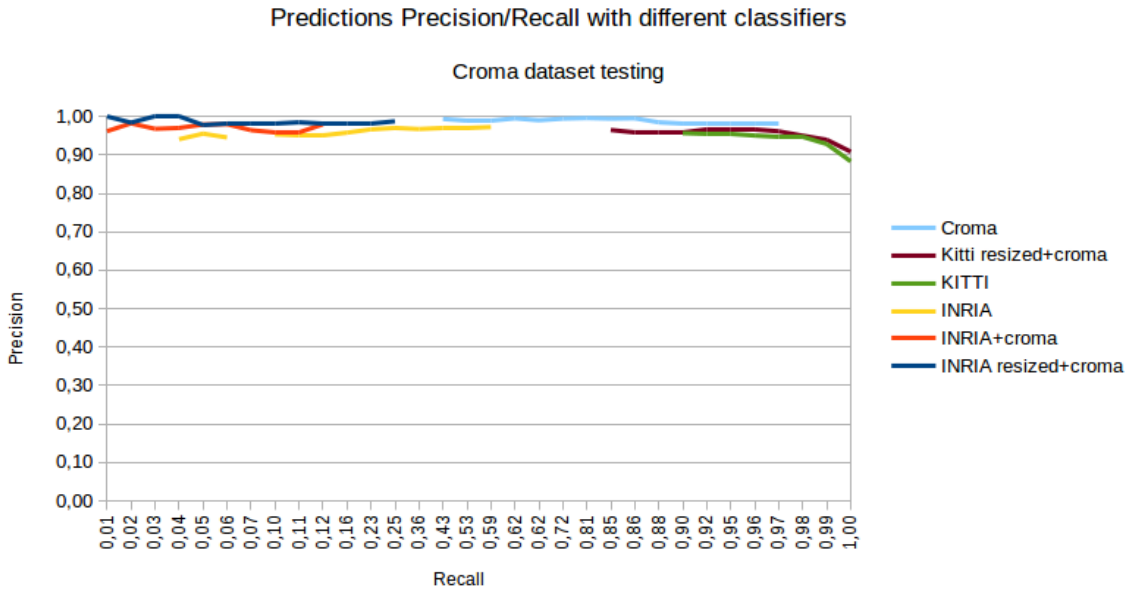


Fig. 61

Figure 62 illustrates the ROC curve for predictions with different classifiers in the Croma testing dataset. The Croma, and Kitti and Croma+Kitti classifiers obtain the best results, as in the previous cases.

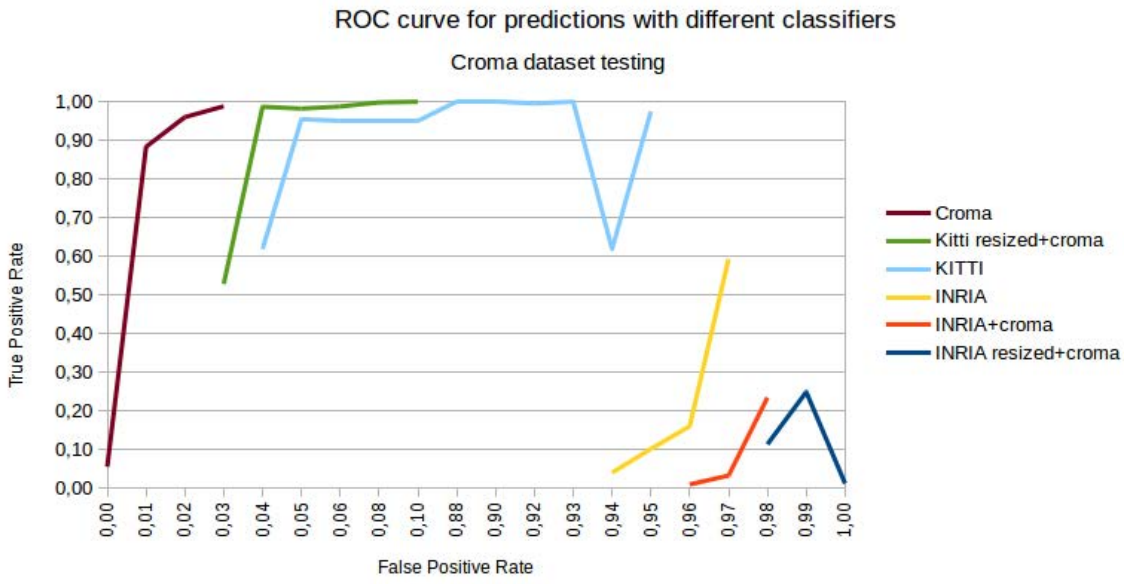


Fig. 62

Figure 63 shows the precision/recall results for predictions with different classifiers in the Inria testing dataset. Although precision is good for all of the classifiers, the precision/recall compromise is better for the Kitti+Croma classifier, followed by the pure Croma classifier. The Inria classifier is probably penalized by the small training set used.

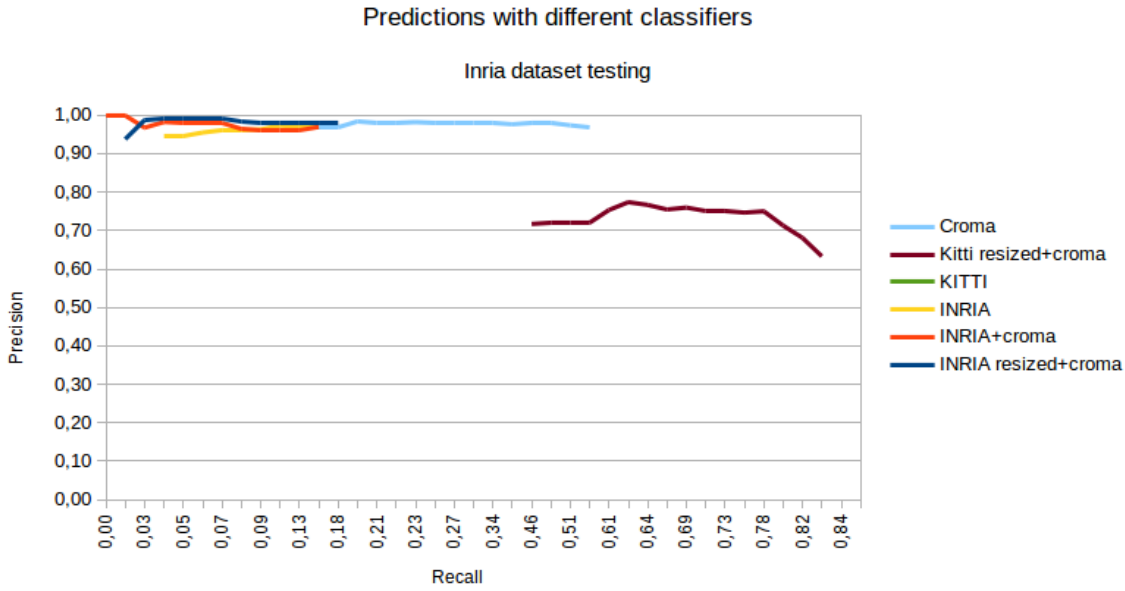


Fig. 63

Figure 64 displays the accuracy/threshold evolution. The Inria classifier obtains good results in its own testing dataset, but the Croma classifiers gets better results even when predicting in a real world testing set.

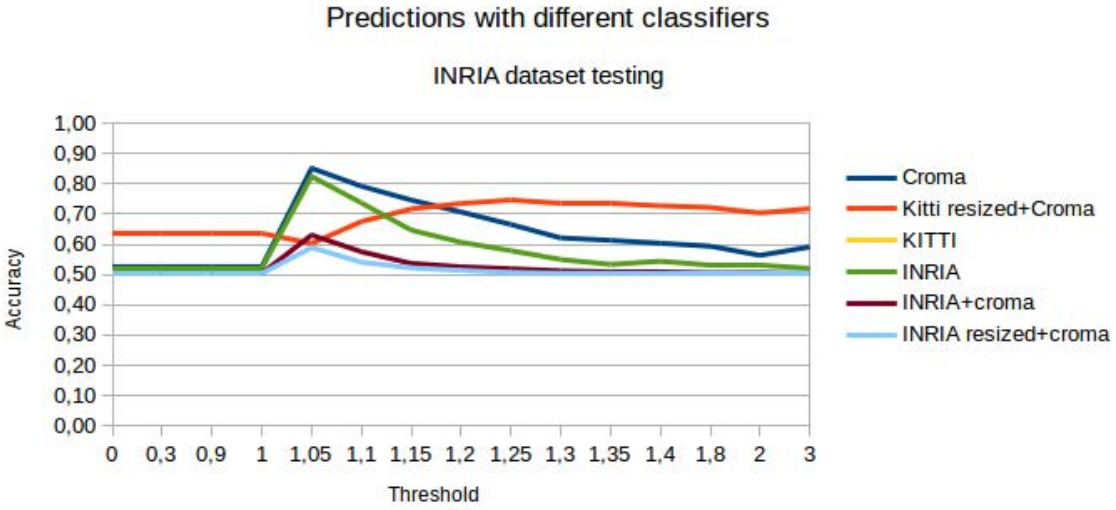


Fig. 64

Figure 65 represents the ROC curve for classifications in the Inria testing dataset. The Croma classifier obtains the best FPR results, but TPR is not as good as Kitti and Kitti+Croma classifiers, which have a better overall behavior.

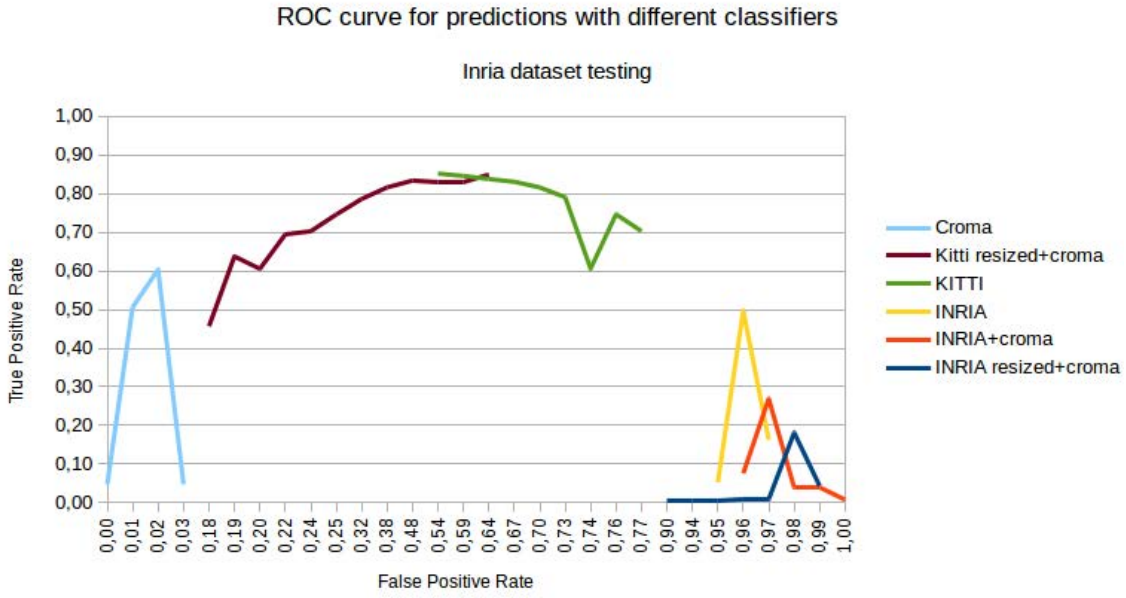


Fig. 65

Figure 66 shows the Accuracy/Threshold evolution for predictions in the Kitti testing dataset. The Kitti and Kitti+Croma classifiers obtain the bests results, followed by the Croma classifier.

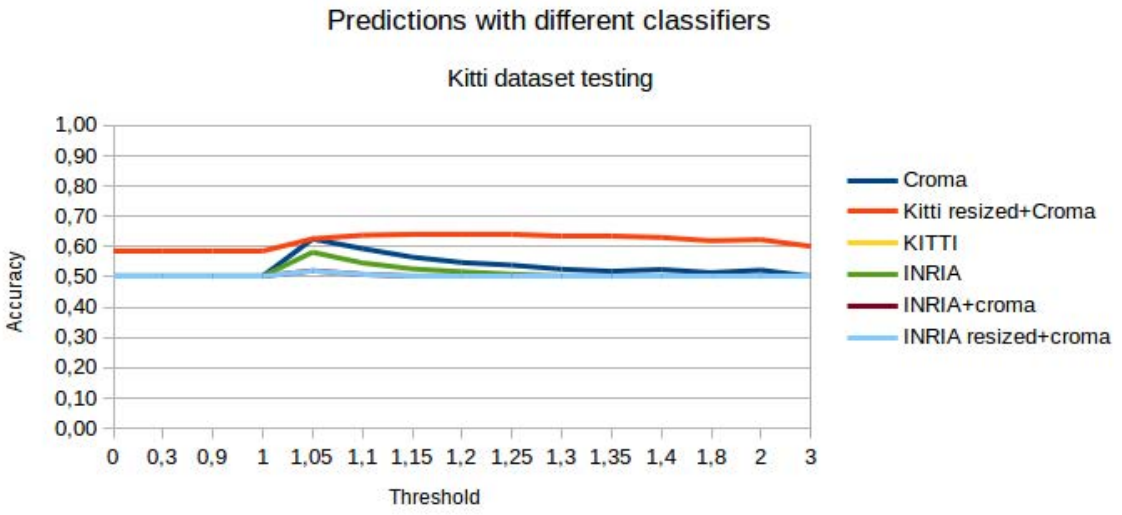


Fig. 66

Figure 67 represents the precision/recall results for predictions in the Kitti testing dataset. The Kitti classifier obtains good results, but the Kitti+Croma classifiers improves its results due to the domain adaptation process. In this case, the Croma classifier obtains good precision results but recall is worse than Kitti and Kitti+Croma classifiers.

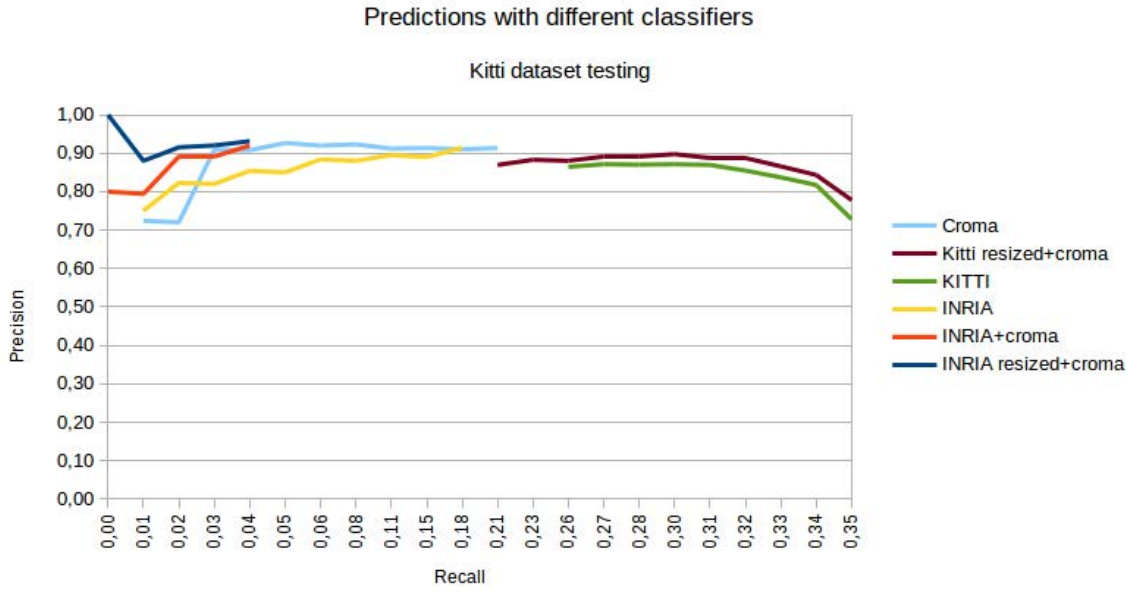


Fig. 67

Figure 68 displays the ROC curves for predictions in the Kitti testing dataset. The Kitti+Croma classifier obtains the best compromise FPR/TPR, followed by the pure Croma classifier and the Kitti classifier.

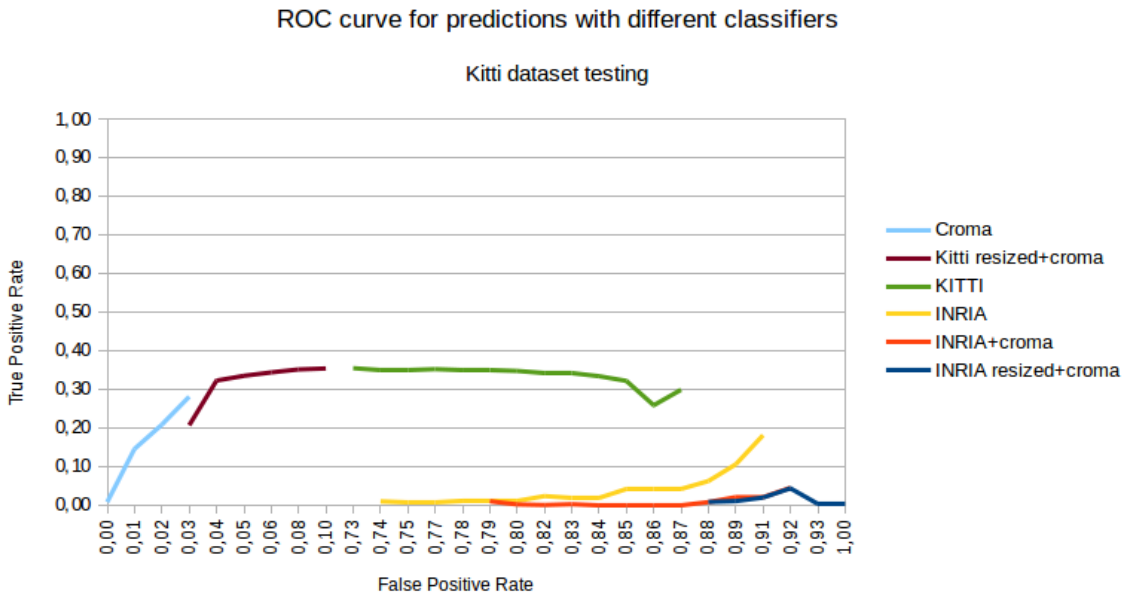


Fig. 68

5.4.1 Conclusions

The classifier generated using the pure Croma training dataset has obtained remarkably good results in the prediction tests against its own testing set, and against real world testing set like Inria and Kitti. These statistics and the experiences performed in the IVVI 2.0 platform backs the use of the synthetic Croma classifier for pedestrian detection.

Domain adaptation techniques in mixed real-world and synthetic classifiers like in the case of the Kitti+Croma classifier can improve the results of the pure real-world classifiers.

The best results are achieved in all cases for a $PredictonThreshold = 1.05$, so this is the Threshold used in the real world IVVI 2.0 testing.

The generation of the Croma training dataset is justified by the results using pure Croma classifiers and by the improved results when enriching real-world training datasets using domain adaptation techniques.

The poor results obtained by the Inria classifier are probably due to the small extension of the Inria training dataset and to the peculiarities of this dataset, with numerous occlusions, incomplete Objects Of Interest, and often uncommon scenarios (ski slopes, groups of pedestrians, dancing courts, etc). The different results obtained for tests with different classifiers in diverse testing datasets suggests that real use applications require a testing phase for the selection of the most adequate classifier for the particular kind of images used for prediction.
The results obtained depend on the internal behavior of the LibSVM library. Even though a very low value for the level of confidence parameter supplied to the detect() function should provide a very high number of false positive detections, and a very high value for the level of confidence parameter should provide a very high number of false negative detections, the 100% and 0% values are very rarely reached. This behavior is due to internal constraints of the library, which probably ignores values of confidence beyond certain limits even if externally imposed.

5.4.2 Bicycle detection

Results for bicycles can be seen in figure 69. At the time of the making, no other datasets were available for comparison. SVM classification using HOG features were used for statistics, with a labeled set for performance

measurement. The performance parameters used are FFPI (False Positives Per Image) and TE (Error rate).

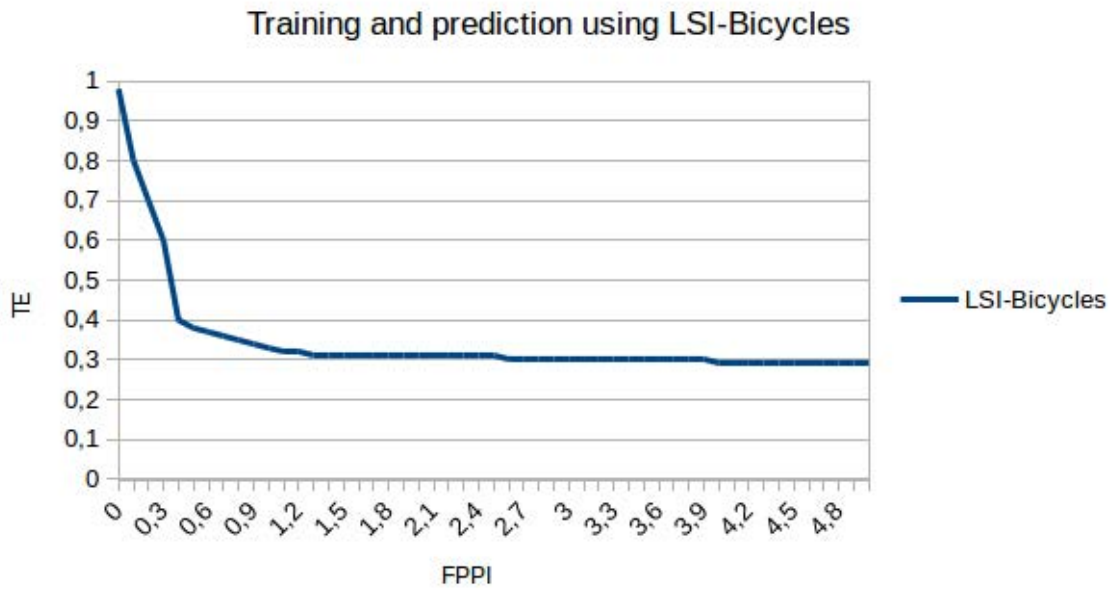


Fig. 69 Results for bicycle prediction using SVM

A vehicle detection algorithm was also used on this work, however it was not developed within the scope of the thesis. This algorithm is used together with the approach presented in this section for the complete detection and classification of obstacles in road environments, and is based on Haar Like features is described in [91].

Chapter 6

Sensor fusion

The goal of the work is the extraction of information from the combination of several sources. These sources present different characteristics in terms of orientation and location of the associated sensors, data rate and timing, and type of reality sensing.
This section is intended to explain the processes required in order to synchronize, align and finally fuse the information obtained from the sensors in the previous stages.

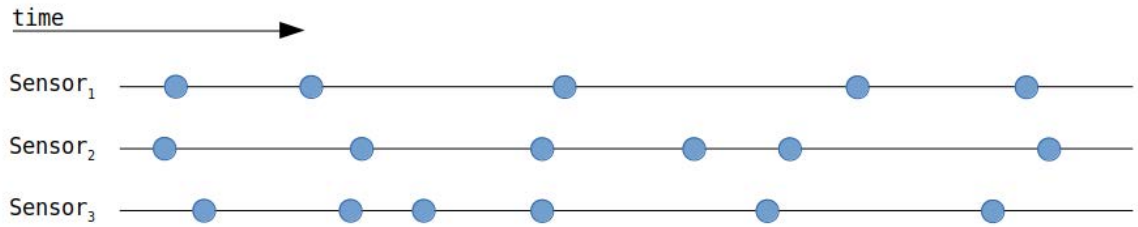
6.1 Data alignment

Different sensors have different field of view and different characteristics, but the reality they sense must match in order to get useful information from them. We must find the relation between each of the sensors and the world, and then the relation between the sensors. The extrinsic parameters of the sensors are rotation and translation. Determining the translation with respect to the reference point of the vehicle is a laborious task, but usually is done just once and is relatively easy, accurate, and small errors does not affect significantly to the precision of the system. Rotation, in the other hand, is more difficult to measure and is more prone to involuntary changes.

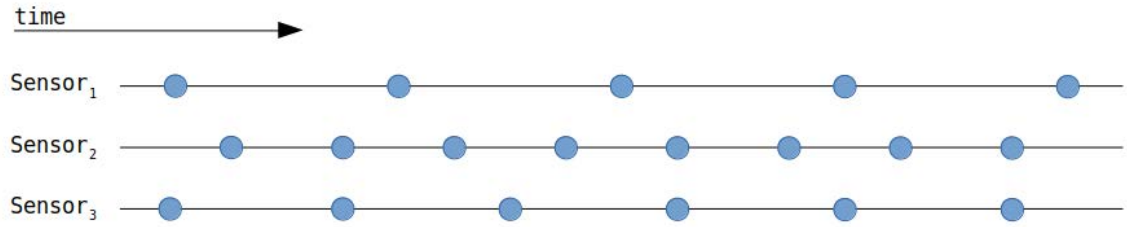
The simultaneous use of several sensor is common in Advanced Driving Assistance Systems (ADAS), in order to obtain higher reliability in the algorithms while detecting, for example, pedestrians, vehicles, or even the topology of the road we are driving on. Therefore, the need for synchronization of several sources with a common reference system arises. This is the case that this work deals with, three-dimensional data captured from a stereo camera and a laser scanner.
Another problem involved in a multisensor system is time alignment. Different sensors usually offer different data frequency and, unless they are physically synchronized, provide information at different times. Time alignment has to deal with this problem.

6.1.1 Time alignment

The sensors involved in the present sensor fusion problem deliver their data at variable rates, depending on the configuration. Different driving situations might need different data rates; so, a restricting solution focused on just the most favorable case is not acceptable. Figure 70 shows the problem to solve: Several sensors each of them with variable data rate (figure 70a) and several sensors with fixed frequency but different between sensors (figure 70b), which is the most common case. Blue dots represent incoming messages from each sensor.



(a) Sensors deliver information with variable frequency



(b) Sensors deliver information with different but fixed frequency

Fig. 70 Time synchronization problem. Sensors deliver information at different rates and different times.

As explained earlier, the Robotic Operating System (ROS) is used in this thesis for sensor management. Messages in ROS contain a time stamp keeping the time of the message generation from the sensor or from an intermediate node.
This environment offers two different time synchronizing techniques for multiple sensors [92]:

- ExactTime Policy

  ExactTime policy only matches messages having the exact same time stamp. The associated callback function will be executed upon the reception of all the matching messages using the ExactTime policy.

- ApproximateTime Policy

  ExactTime policy matches messages with the exact same time stamp, but this behavior might not be right for some situations. ApproximateTime

policy takes some control parameters to match messages with different timestamps, as will be explained.

A topic-specific queue is designed for messages inclusion as they arrive.

Being t the last published set of messages to match, and t_+1 the next one to be created:

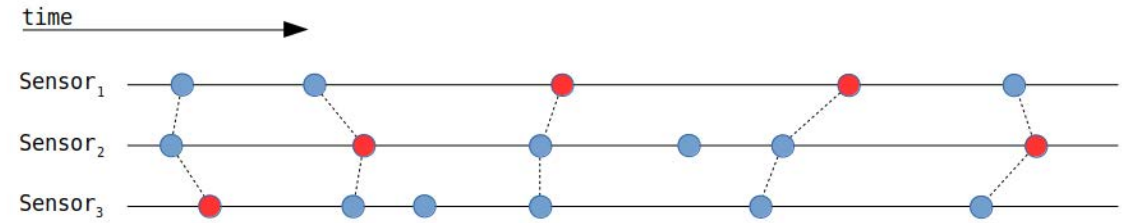
When a new set t is published, every message older than the previously message existing in the aforementioned queue is discarded.

Once every topic-specific queue contains at least one message, the pivot will be chosen as the latest message between the heads of the queues.

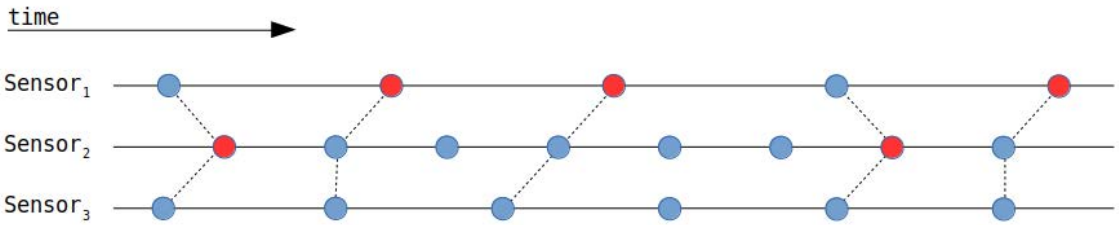
An iterative process is started then:

Find the pivot and a first valid candidate set. Search the queues until empty for a better candidate. Search t_{+1}, t_{+2} ... messages to prove that the chosen candidate is the best.

According to the explained Approximated time policy, the example in figure 70 would be synchronized in the way shown in figure 71, being the blue dots the incoming messages from each sensor, and the red dots the chosen pivot for each synchronization. Dashed lines represent the messages chosen for each synchronization. Messages not included in any synchronization are lost in this particular process, but might be included in other synchronization process executed by other nodes in ROS.



(a) Synchronization for variable frequency



(b) Synchronization for different but fixed frequency

Fig. 71 Synchronization using the ROS ApproximateTime policy. Red dots are the chosen pivots for each synchronization, which is represented by dashed lines.

As shown in figure 72, ApproximateTime is the policy used with the messages coming directly from the nodes associated to the sensors, while messages generated from other nodes will use the ExactTime policy, having all of those messages the exact same time stamp than the original image message from the camera.

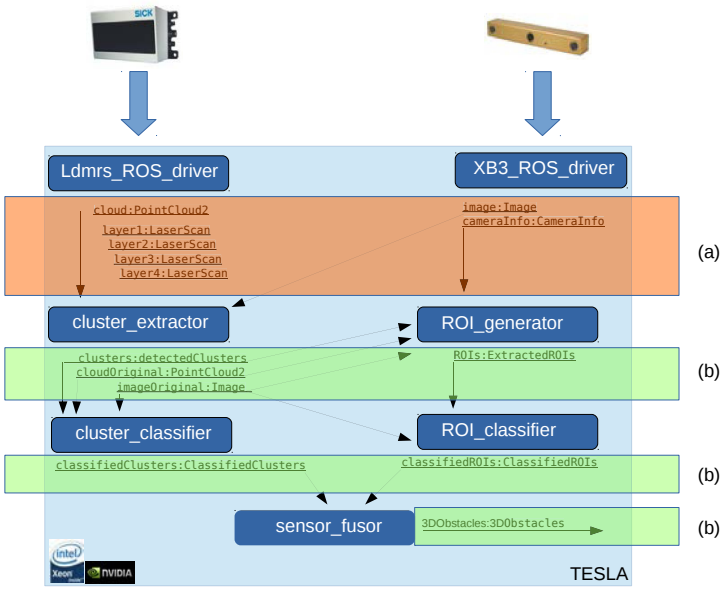


Fig. 72 Time synchronization in the system. (a) Messages from the sensors use Approximate-Time Policy, (b) node-generated messages use ExactTime Policy.

6.1.2 Location and Orientation alignment

Sensor fusion requires the use of data alignment algorithms to convert from the systems of coordinates of the different sensors to a common one, so data from different sources can be compared and more information can be extracted.

Location and orientation alignment requires a previous knowledge of some rotation and translation parameters related to all of the sensors. These parameters can be obtained using attended calibration algorithms involving fixed geometric patterns which does not allow for a fast and unattended calibration, or use innovative algorithms like the one explained in the next section, a novel method for automatic and unattended extrinsic parameters calibration without the need of any special shape that can be performed at virtually any time during a driving session.

Automatic Laser And Camera Extrinsic Calibration for Data Fusion Using Road Plane

The presented system is based on data fusion between several sensors, which acquire different physical phenomena. Thus, each of these sensors has its own system of reference, and extrinsic parameters between sensors system of reference must be estimated in order to perform the data alignment. To achieve the necessary alignment, rotation and translation between sensors must be estimated. Some methods have already been proposed by other authors, involving chessboards or specific patterns detectable by all of the sensors involved in the fusion. This is cumbersome and requires driver implication or some help from others, needs specific and stationary environment and to be performed manually again in case of change of orientation or translation between sensors.

Extrinsic parameters estimation is key for data alignment. A mobile system such as a vehicle can suffer changes in sensor orientation or position, so it is important to implement a method for extrinsic parameters estimation in an unattended and convenient way.

The solution presented obtains a point cloud (PC) from both the laser scanner and the stereo camera. These point clouds must present a significant amount of points belonging to a flat surface in front of the vehicle, in this specific application it is the road. Applying to both point clouds the RANSAC [93] algorithm to the detection of planes in the space, we get a unitary vector normal to the surface for every sensor. These vectors determine the two orientation angles for the sensor (pitch and roll), and the height from the plane found is obtained from the projection of the origin of the point cloud in the plane of the road. The last orientation angle (yaw) is obtained by obtaining the projection of the distances to the obstacles within the road, calculating their angle signatures and correlating among the different clouds of points from the different sensors.

Extrinsic parameters estimation from a point cloud

The process starts from the three-dimensional reconstruction, based on a PC, of the environment with respect to the camera system reference {c}. This plane can be represented in the world system reference {m}, attached at ground plane, as seen in figure 73.

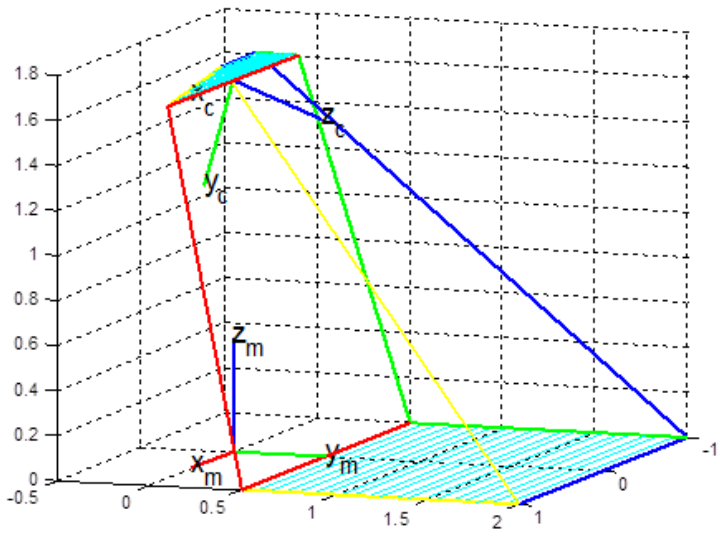


Fig. 73 Framework of the images system: Camera {c}, World {m}

Using the M-estimator-SAmple-Consensus (MSAC) algorithm [94], applied to plane detection in the space, it is possible to generate a $[a,b,c,d]$ vector defining the plane as the most populated plane in the PC.

$$\pi_{(x)} : ax_c + by_c + cz_c + d = 0 \tag{7}$$

In its Hessian form, equation 7 can be written as:

$$\pi_{(x)} : \underset{n}{\rightarrow} \cdot \underset{p}{\rightarrow} = h \tag{8}$$

From (8) we deduct that the vector $\vec{n}$ is normal to the $\pi_{(x)}$ plane we found, as the projection on $\vec{n}$ of any point located in the plane always generates a fixed

distance. This distance is the minimal from the plane to the PC's origin of coordinates, thus the height h of the sensor.

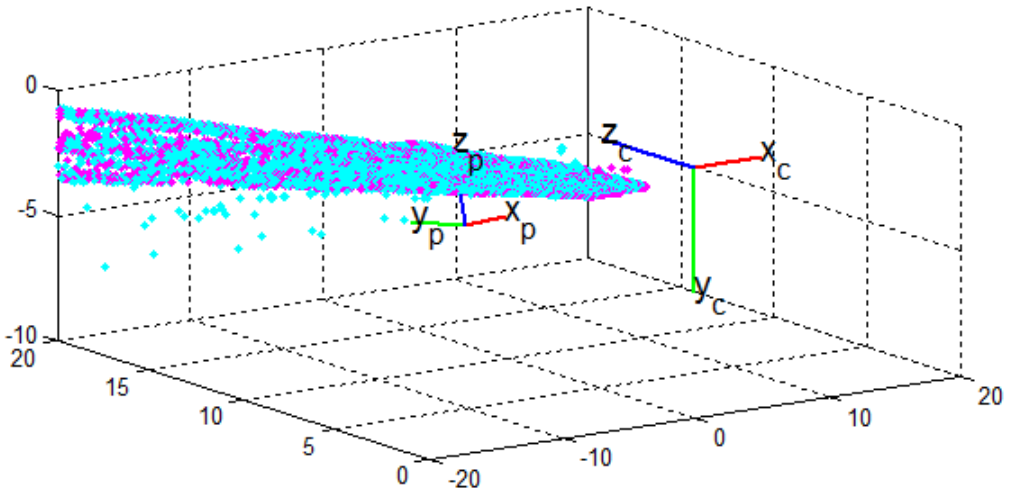


Fig. 74 Base obtained on the road plane {p}. Cyan points are inliers and magenta points are its projections on the plane.

The direction of the $\vec{n}$ vector is then defined as the cross-product between u_p and v_p, which corresponds to the direction of Z_p. Figure 74 depicts the base generated on the road plane, system reference p.

1. Rotation extraction with respect to the camera on X and Y'

   The vector $c_{\omega_p} = [c_{\omega_{px}} c_{\omega_{py}} c_{\omega_{pz}}]^T$ normal to the road plane as seen from the camera in Figure 75, correspondent to is related to the orientation of the axis of the direction of the camera according to (9).

$$c_{\omega_p} = Rot_x(\frac{\pi}{2}) * Rot_x(-\alpha) * Rot_y(-\sigma) * p_{\omega_p} \tag{9}$$

   The rotation on $Z^{''}$ is not taken into account, as its application does not change the road plane Z_p axis with respect to the axis Z_c of the camera.

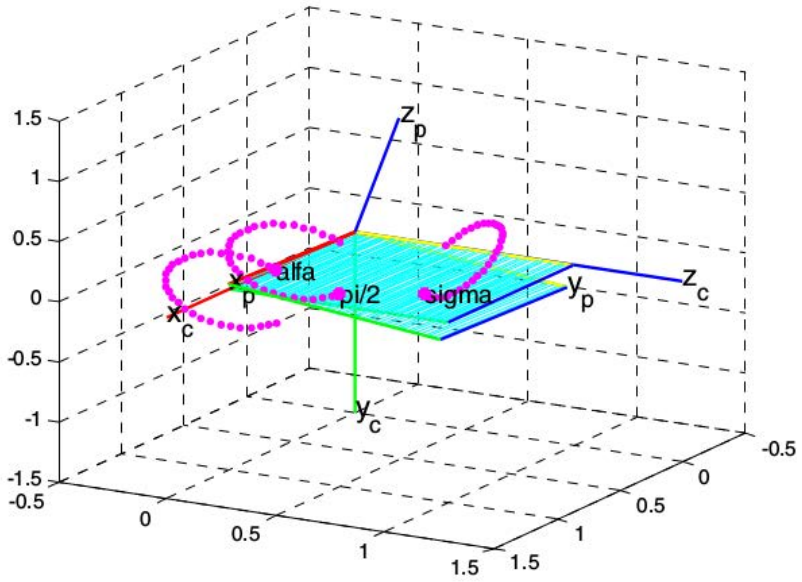


Fig. 75 Road plane rotation respect to camera reference system

Expanding (9):

$$c_\omega = {}^{c}R_p * {}^{p}\omega \tag{10}$$

$$\begin{bmatrix} c_{\omega_x} \\ c_{\omega_y} \\ c_{\omega_z} \end{bmatrix} = \begin{bmatrix} c\sigma & 0 & -s\sigma \\ -c\alpha s\sigma & s\alpha & -c\alpha c\sigma \\ s\alpha s\sigma & c\alpha & s\alpha c\sigma \end{bmatrix} * \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{11}$$

$$\sigma = sin^{-1}(-c_{\omega_x}) \tag{12}$$

$$\alpha = cos^{-1}(-\frac{c_{\omega_y}}{c\sigma}) \tag{13}$$

Last, relating (7) and (8), sensor height is computed as

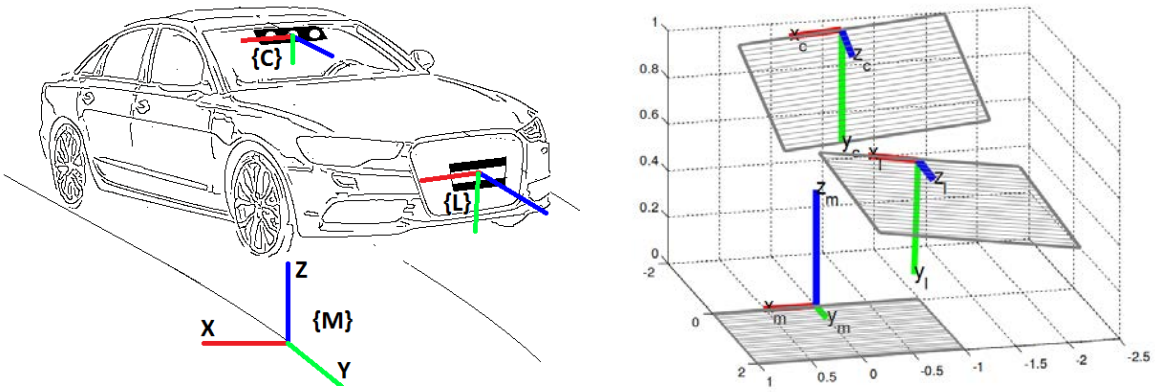
$$h = -d \tag{14}$$

2. Algorithm optimization for large PC

Two strategies were implemented to accelerate the MSAC algorithm, RANSAC variant. One consists on eliminating all the points allegedly not located in the ground plane, this is, in the upper half of the image, as far as it is a PC obtained from a stereo reconstruction. The second one consists on extracting a uniform sample of the point cloud in order to get a limited amount of data representing the environment, thus the PC can be processed in real time, less than 100ms.

Data alignment between two three-dimensional capture sensors

The problem to solve then is finding the spatial position of a sensor reference systemc (for the stereo-camera) respect to a sensor reference systeml (for the laser), both attached to a mobile system i.e. the vehicle. Each sensor has its own position, and its Field of View (FOV) to see the road as the mobile system moves on, as shown in figure 76.



(a) Sensor array in vehicle.

(b) Sensor array systems of reference.

Fig. 76 Sensor array configuration, stereo camera located in the windshield, and laser scanner located in the bumper.

The next step is based on the point clouds PC_c and PC_l , captured from each of the sensors. Each cloud is transformed until the most populated plane matches the $X_m = Y_m$ plane. In each transform ${}^{M}T_c$ and ${}^{M}T_l$, the respective extrinsic parameters $\begin{bmatrix} \alpha_c & \beta c & hc \end{bmatrix}$ and $\begin{bmatrix} \alpha_l & \beta l & hl \end{bmatrix}$ are estimated.

Figure 77 shows PC from each sensor, green-red are inliers-outliers respectively and purple-blue are inliers-outliers.
Transforms from c and l into m are presented in (7) and (8).

$$ {}^{l}P = {}^{l}T_{m} \quad {}^{m}T_{c} \quad {}^{c}P \tag{15} $$

$$ {}^{l}T_{c} = {}^{l}T_{m} \quad {}^{m}T_{c} \tag{16} $$

$$ \begin{aligned} D_z(h)R_z(\gamma)R_y(\beta)R_x(\alpha) \quad &= \\ R_x(-\alpha_l)R_y(-\beta_l) \quad &\cdot \\ R_z(-\gamma_{lc}Dyz)([-dx_{lc}-dy_{lc}-h_{lc}]) \quad &\cdot \\ R_y(\beta_c)R_x(\alpha_c) \end{aligned} \tag{17} $$

$$ \begin{bmatrix} c\beta c\gamma & s\alpha s\beta c\gamma + c\alpha s\gamma & -c\alpha s\beta c\gamma + s\alpha s\gamma & 0 \\ -c\beta s\gamma & -s\alpha s\beta s\gamma + c\alpha c\gamma & -c\alpha s\beta s\gamma + s\alpha c\gamma & 0 \\ s\beta & -s\alpha c\beta & c\alpha c\beta & -h \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} & T_{13} & T_{14} \\ T_{21} & T_{22} & T_{23} & T_{24} \\ T_{31} & T_{32} & T_{33} & T_{34} \\ T_{41} & T_{42} & T_{43} & T_{44} \end{bmatrix} \tag{18} $$

Decomposition matrix ${}^{l}T_{c}$ through world reference system m and its relationship to the global transformation element is shown in (18).
As the plane alignment lets free the rotation angle around Z_m, a rotation of ${}^{m}PC_{c}$ is made with respect to ${}^{m}PC_{l}$ in an angle γ_{cl}, assuming that the translation of the sensors in the x_m i.e. dx_{cl} and y_m i.e. dy_{cl} axis is known.
As the point clouds are different, in order to find the γ_{cl} angle, it is necessary to adjust the data by looking for the highest similarity. To do so, outliers from ${}^{m}PC_{l}$ farther than 10 meters from the Z_m axis are removed, obtaining ${}^{m}PC_{out_l}$. Then, the minimal and maximal distance to Z_m, ${}^{m}PC_{out_l} : [d_{min}d_{max}]_{\perp z_m}$ are calculated among the filtered points, then the minimal and maximal distance

along the Z_m, ${}^{m}PC_{out_l} : [h_{min}h_{max}]_{z_m}$. Using the computed boundaries in cylindrical coordinates, information from ${}^{m}PC_c$ is filtered in order to extract the ${}^{m}PC_{out_c}$ cloud. Both filtered clouds are shown on the figure 78b, ${}^{m}PC_{out_l}$ in blue and ${}^{m}PC_{out_c}$ in green. Figure 78a shows the same point clouds before being filtered.

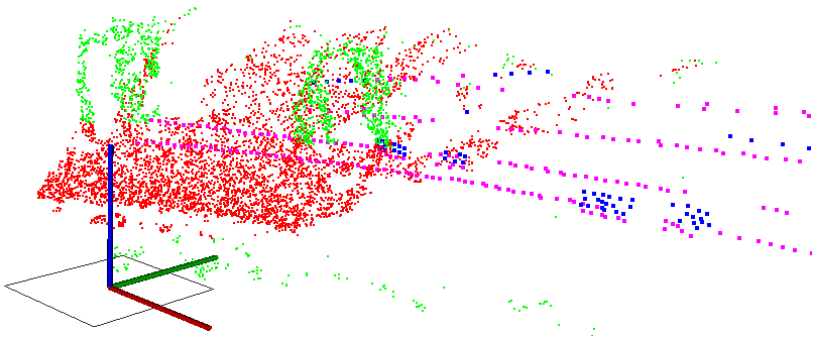
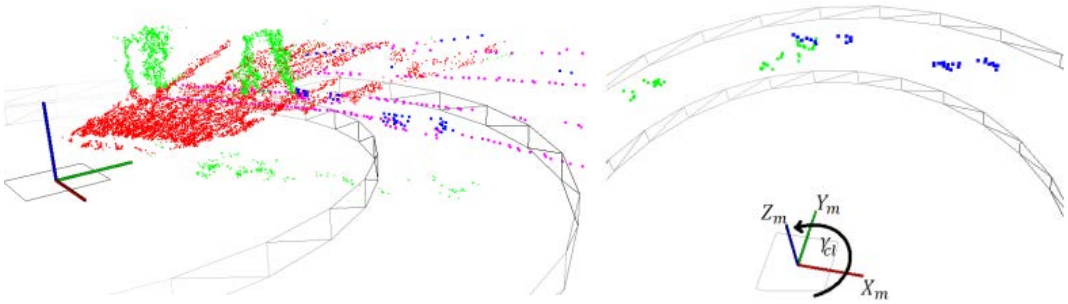


Fig. 77 Alignment and segmentation of the road plane in point clouds PCL_c and PCL_l green-red are PCL_c inliers-outliers and purple-blue are PCL_l inliers- and outliers.



(a) Unfiltered clouds.

(b) Filtered clouds.

Fig. 78 ${}^{m}PCL_{out_l}$ and ${}^{m}PCL_{out_c}$ in the Region of Interest $[d_{min}d_{max}]_{z_m}$ and $[h_{min}h_{max}]_{z_m}$

The final procedure for finding γ_{cl} consists on obtaining the projections from each cloud $Proj_{xy}({}^{m}PC_{out_c})$ and $Proj_{xy}({}^{m}PC_{out_l})$, or ${}^{xy}PC^{xy}_{out_c}PC_{out_l}$. To do so, the Z coordinate is eliminated from the definition of every point, as shown in figure 79.

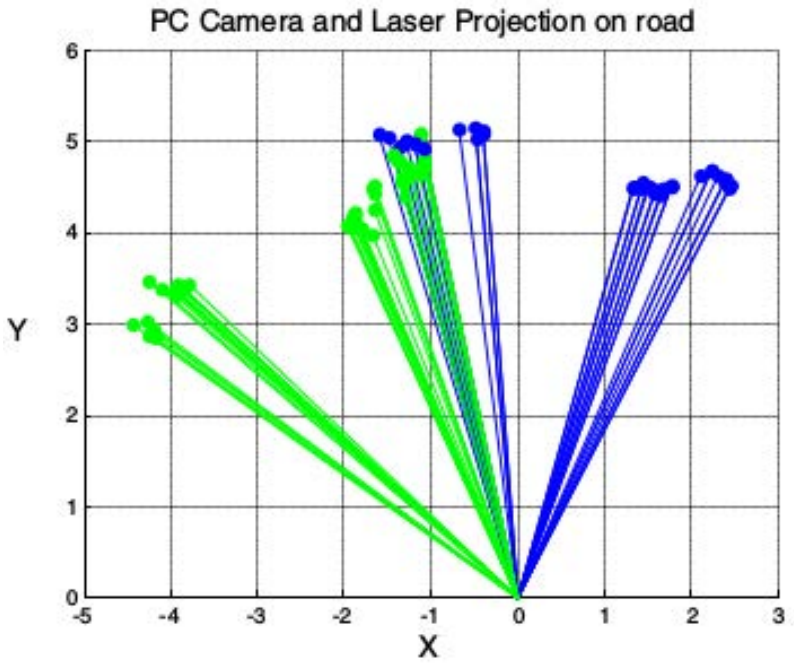


Fig. 79 PC camera and PC laser projections

Next, the signature form each projection $signature(PC_{outc_{xy}})$ i.e. s_{outc} and $signature(PC_{outl_{xy}})$ or s_{outl} are obtained. To do so, the bi-dimensional PC is translated into polar coordinates as magnitude and angle of a vector S, meaning the angle of every point the position n, where the magnitude S of the point will be stored, as shown in figure 80.

$$s(n) = s(ang(PC(i)_{xy})) = mag(PC(i)_{xy}) \tag{19}$$

Next, the correlation between both signatures is found.

$$E(m) = \sum_{n} (S_{out_c}(m-n) - S_{out_l}(n))^2 \tag{20}$$

Figure 81 shows the correlation between profile signatures for PC, laser and camera projection.

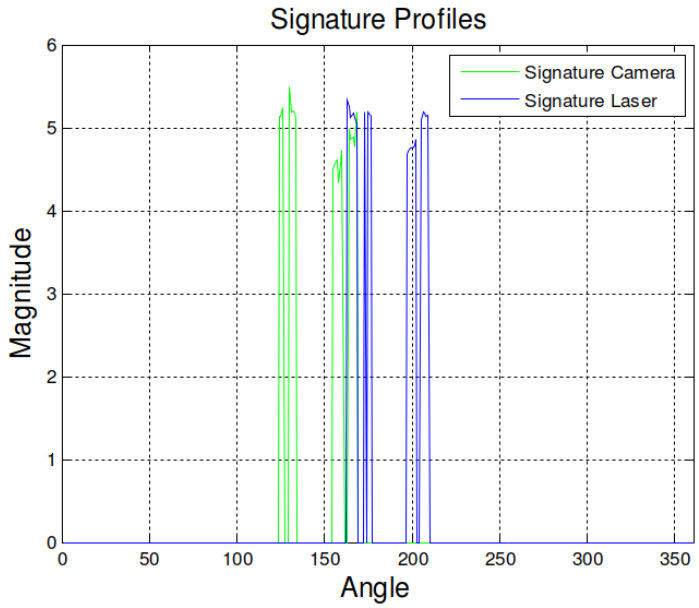


Fig. 80 Profile signatures for PC, camera and laser projections in XY.

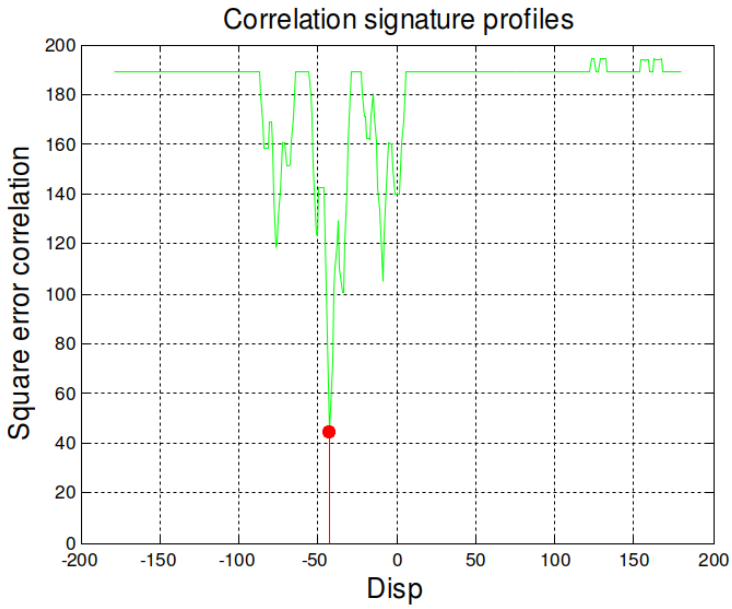


Fig. 81 Correlation between profile signatures on camera and laser PC projections.

The minimal value of E is found on a m^* shifting, whose values matches the rotation γ_{cl} that was searched.

$$\gamma_{cl} = m^* : min(E(m))_{\lfloor m^*} \tag{21}$$

After finding γ_{cl}, rotation angles between sensors can be obtained, and sensor height is computed from (18). So,

$$\begin{aligned} -c\beta s\gamma &= T_{12} \\ s\beta &= T_{13} \\ -s\alpha c\beta &= T_{23} \\ c\alpha c\beta &= T_{33} \\ h &= h_{cl} \\ \beta &= sin^{-1}(T_{13}) \\ \alpha = cos^{-1}(\frac{T_{33}}{c\beta}), \alpha &= sin^{-1}(-\frac{T_{23}}{c\beta}) \\ \gamma &= sin^{-1}(-\frac{T_{12}}{c\beta}) \end{aligned} \tag{22}$$

Data alignment testing

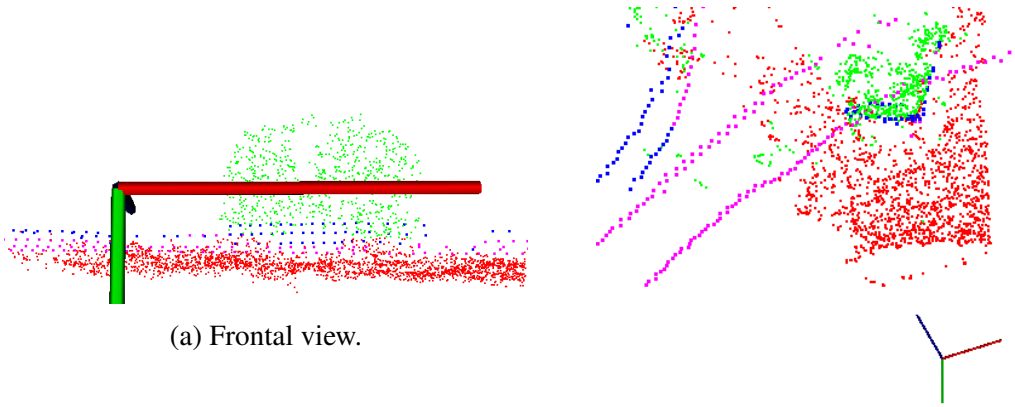
Tests were performed for checking the accuracy of the relative extrinsic parameters estimation algorithm. To do so, the first tests use a single frame, and later the algorithm is tested in a recorded sequence.

- Relative extrinsic parameters measurement

  Absolute extrinsic parameters measurement for each sensor camera and laser, and relative measurement from camera to laser, have been done from the configuration presented in figure 76 and considering a scene like the one in figure 82.

Fig. 82 Laser projection on the image.

The result of the ground plane detection and the subsequent alignment in the Z axis is shown in figure 83.



(a) Frontal view.

(b) Upper view.

Fig. 83 Ground plane alignment for the point clouds obtained from a stereo camera and a laser scanner.

The experiment is now repeated, this time for a sequence of images and laser captures similar to the depicted in figure 83 Data are shown in Table 7.

Table 7 Extrinsic parameters measured from a sequence of synchronized images and laser captures.

Param/Sensor	Camera		Laser		Camera-Laser	
	Mean	**Std dev**	**Mean**	**Std dev**	**Mean**	**Std dev**
High	1.27	0.007	0.267	0.04	-0.937	0.048
Pitch	-7,3	0.06	-2.22	0.29	-5.14	0.29
Roll	1.08	0.18	-0.71	0.23	4.12	0.33
Yaw	-4.39	0.34	0	0	2.34	0.34

- Testing conclusions

  The estimation method for extrinsic parameters based on the road plane detection from a point cloud shows a pitch angle difference respect to the ground truth of 0.5 degrees and 0.4 degrees for roll angle. Furthermore, the reference sensor (IMU) exhibits the same standard deviation of 0.08 degrees for the pitch angle than the proposed measurement method. As far as the roll angle is concerned, the IMU sensor shows a standard deviation of 0.28 against 0.12 in the proposed method. After the transformation of the point clouds to the road reference system, i.e., the planes estimated for each point cloud coming from each sensor are coplanar and superimposed, the final alignment produced good results. This alignment was achieved by rotating the camera point cloud until the objects, out of road, matched in both point clouds. In a static sequence, standard deviation of the rotation with respect to the Z axis of the road is 0.33 degrees. Relative extrinsic estimation between camera and laser was tested by projecting the laser on the image and checking quantitatively the match, as seen in figure 82.

6.2 Camera and laser information fusion

In the presented thesis, a Sick LDMRS 4-layer Laser Scanner and a camera are used. Laser scanner for primary obstacle detection and later for classification, and stereo capability from the camera is used for point cloud ground representation and data alignment parameters estimation; later one of the cameras from the trinocular camera is used as a monocular camera for image capturing.

As explained in figure 84, the laser scanner generates a point cloud in which the system extracts the obstacles as clusters of points. These clusters are used both for ROI generation in the images and as information for obstacle classification. The extracted ROIs in the image are processed for obstacle classification using AI methods applied to Computer Vision. The last step in the process performs further information fusion between laser and camera for a final obstacle classification based on machine learning. A database with manually labeled images and point clouds is used for SVM training and testing in the classification process.

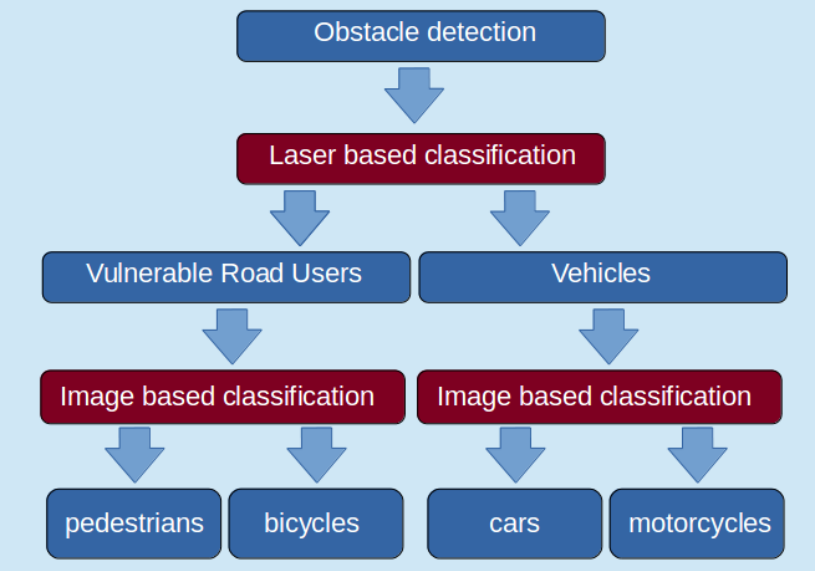


Fig. 84 Obstacle classification process. Initial obstacle detection, Laser classifies between VRU and vehicles and generates ROIs for image classification. CV classifies VRU between pedestrians and vehicles, and vehicles between cars and motorcycles.

Once all the calibration parameters, i.e. roll, pitch, yaw and x,y,z translations between sensors have been computed, the system is able to translate from laser coordinates into camera coordinates in the image for obstacle classification using Computer Vision. The conversion between laser and image coordinate systems can be performed as in equation 23 where T represents the translation vector and R the rotation matrix between sensors.

$$
\begin{aligned}
\begin{bmatrix} x \\ y \\ z \end{bmatrix} &= R\left(\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} + T\right) \\
R &= \begin{bmatrix} cos(\delta) & 0 & sin(\delta) \\ 0 & 1 & 0 \\ -sin(\delta) & 0 & cos(\delta) \end{bmatrix} \cdot \\
&\quad \begin{bmatrix} 1 & 0 & 1 \\ 0 & cos(\phi) & -sin(\phi) \\ 0 & sin(\phi) & cos(\phi) \end{bmatrix} \cdot \\
&\quad \begin{bmatrix} cos(\theta) & -sin(\theta) & 0 \\ sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
T &= \begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix}
\end{aligned}
\tag{23}
$$

6.2.1 Results of the sensor fusion

An overview of the performance measurement strategy of the system is presented next to illustrate the making of the performance statistics.

A SQLite database is created for every experiment, containing exhaustive data about detections and sensor readings. Intermediate images, point clouds, clusters and measurements are stored in disk for later research and performance metering, and the database stores paths to images and point clouds files, as well as information about the detections. Every detected cluster in the session generates a row in the database containing, among other information, the fields shown in table 8 in order to store all the available information about

the session. Several fields existing but not shown are provided for future compatibility with other sensors present in the IVVI 2.0 platform.

Table 8 Fields in the table containing information about each detection and classification in the system

Name	Type	Explanation
original	TEXT	Name of original image in disk
crop	TEXT	Name of crop file corresponding to detection in original image
cloud	TEXT	Name of point cloud file corresponding to the detection
imageObstacle	TEXT	Type of obstacle detected as image
cloudObstacle	TEXT	Type of obstacle detected as point cloud
fusionObstacle	TEXT	Type of obstacle detected as sensor fusion
width	INT	Width in pixels of the window in image containing the obstacle
height	INT	Height in pixels of the window in image containing the obstacle
xSup	INT	Up-left X coordinate of the candidate window
ySup	INT	Y coordinate of the candidate window
cropDir	TEXT	Folder in disk containing the crop image file
cloudDir	TEXT	Folder in disk containing the point cloud file
originalDir	TEXT	Folder in disk containing the original image file
detectionXsup	INT	Up-left X coord. of the detection in the cropped image window
detectionYsup	INT	Up-left Y coord. of the detection in the cropped image
detectionXinf	INT	Low-right X coord. detection in the cropped image
detectionYinf	INT	Low-right X coord. of the detection in the cropped image
imageThreshold	REAL	Confidence of the classification in image
cloudThreshold	REAL	Confidence of the classification in point cloud
fusionThreshold	REAL	Confidence of the classification in sensor fusion
distance	REAL	Distance to the obstacle measured in the point cloud

Sliding Window improvement using laser sensor fusion.

Computer vision classification is performed only in the ROIs selected by the laser obstacle detection. This initial stage of the sensor fusion provides significant savings in computational effort compared to classification using the whole image. OOI classification using CV uses the Sliding Window (SW) technique, as seen in figure 85. Initially (figure 85a), the smallest SW is

extracted from the image to search for OOI on it. The SW is moved through the image in Δx horizontal leaps until the right end of the image, then a Δy vertical leap and again the horizontal iteration from left end to right end, as shown in figure 85b, until all the image is covered. In the next iteration, the full image is scaled by a fixed factor, as seen in figure 86. Applying the same SW size to this smaller image, figure 85c is obtained, with a bigger relation between the SW size in relation with the whole image. Eventually, the maximum SW size fitting in the image is obtained for searching, as in figure 85d.

(a) Initial sliding window

(b) Leap Δx and Δy moving the sliding window

(c) Medium size sliding window

(d) Big size sliding window

(e) Initial window size using sliding window improvement.

Fig. 85 Sliding window improvement using laser sensor fusion.

Fig. 86 Pyramid reduction.

The use of ROIs obtained from prior laser obstacle detection allows to optimize the SW generation, knowing in advance the physical characteristics of

the OOI. Laser obstacle detection provides the 3D location of the obstacle, including the distance from the sensor to the obstacle. Assuming that the obstacle detected will start from the ground, and applying some padding around the expected dimension of the object (for pedestrians, a recommended configuration is 1 meter below the ground level, 1 meter above an estimated 2 meter height of the pedestrian, and 1 meter on each side). Figure 87 explains these paddings. Big red dots are the cluster representing the obstacle. The blue square labeled as "3" (meaning the fourth obstacle detected on the scene) represents the ROI to extract, and is composed adding 1 meter below the expected ground level, 1 meter on top of the expected 2 meters pedestrian maximum height, 1 meter left to the left-most point in the cluster, and 1 meter right to the right-most point in the cluster. The portion of the image containing that 3D region of the space is computed using the Distance to Obstacle and equation 23. These paddings are suitable for pedestrians, but the same technique is applied for ROI generation when looking for other actors, such as bicycles, motorbikes or cars. If the system is trying to classify more than one actor, several ROIs are generated, each of them with the appropriated paddings for the OI. These actor-specific ROIs are then searched using the appropriated classifier using CV techniques.

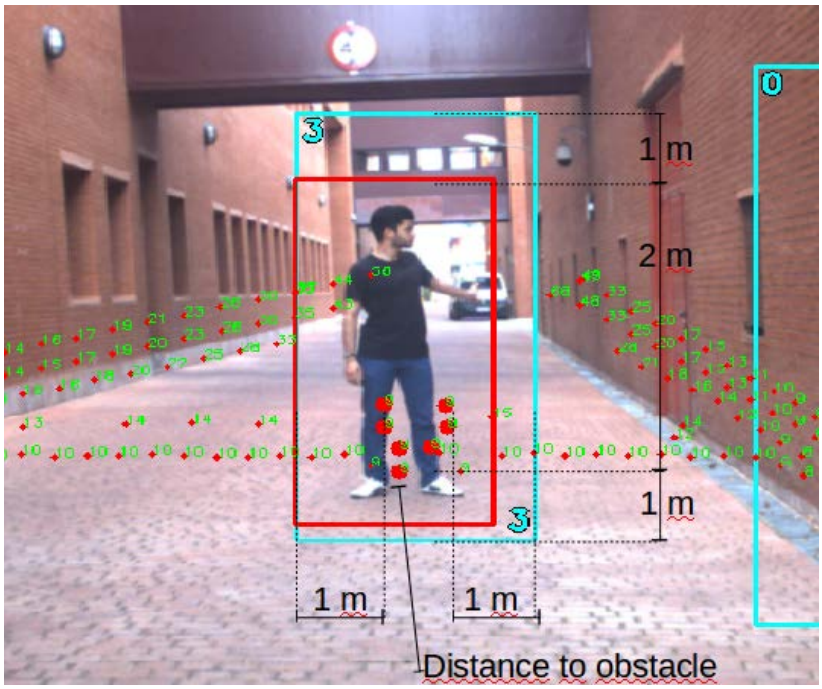


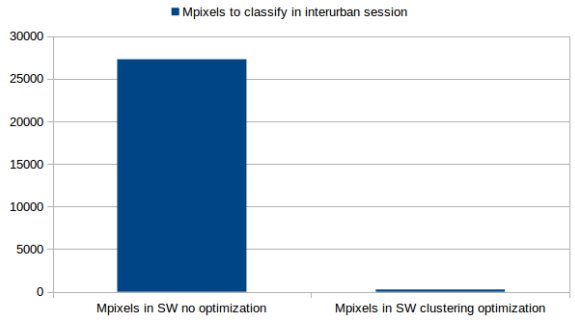
Fig. 87 Pyramid reduction.

Applying equation 23 to the 3D coordinates obtained from the laser obstacle detection and the aforementioned padding, the smallest and greatest possible dimensions of the OOI in the ROI cropped are obtained, so the SW initial size can be adjusted in order to be big enough to contain the OOI and avoid the smaller windows, as shown in figure 85e.

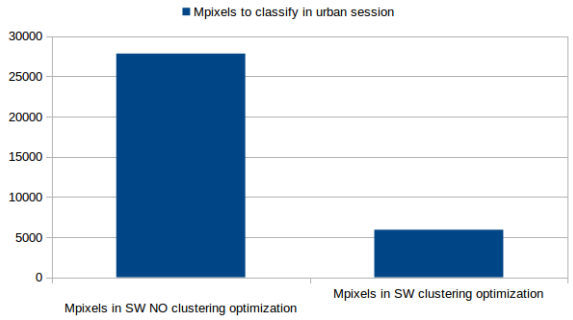
Without the use of sensor fusion improvement for SW, a standard configuration of 1024x768 pixels image, 128x64 pixels SW, Δx and Δy of 32 pixels and a pyramid reduction rate of 1.5, requires 2,038 classifications in 128x64 pixels windows for a full search.

The savings on using sensor fusion improvement for SW are very variable, depending on the environment and, specially, the number and size of the obstacles found. Close obstacles create bigger ROIs than the distant ones, so urban environment tends to provide more frequent and bigger ROIs than interurban. The clustering algorithm configuration also affects greatly to the number and size of the ROIs extracted. A study of the statistics of several

session captures is represented in figures 88 and 89. Figure 88 shows a comparison between the number of Mpixels that the CV classification process has to compute in pure SW algorithm compared to clustering optimized SW. Figure 89 shows the relation between the amount of 128x64 pixels windows that the CV classification process has to compute in pure SW algorithm compared to clustering optimized SW.

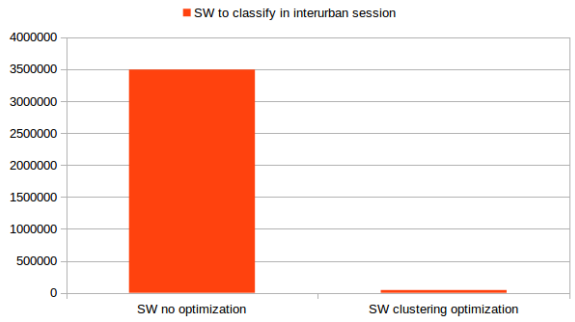


(a) Mpixels to classify in a interurban session.

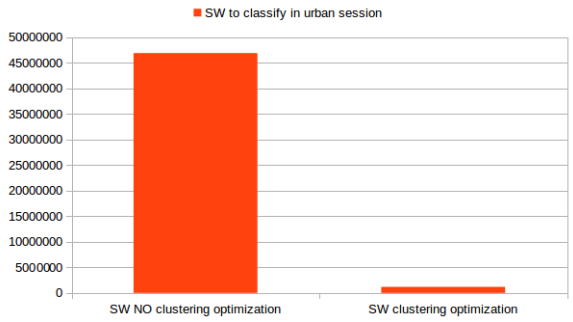


(b) Mpixels to classify in a urban session.

Fig. 88 Sliding window performance improvement using sensor fusion.

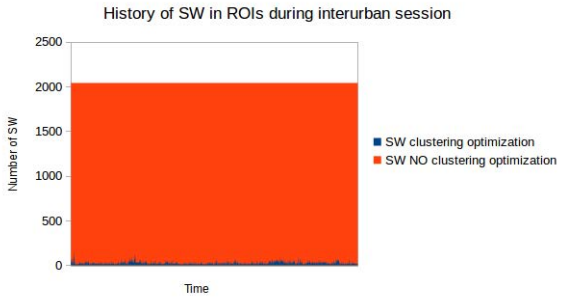


(a) Number of SW to classify in a interurban session.

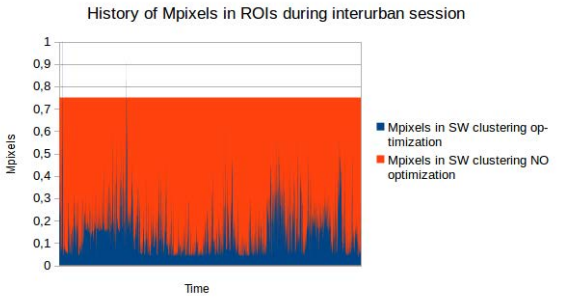


(b) Number of SW to classify in an urban session.

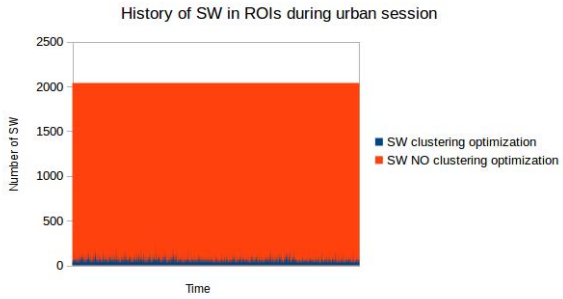
Fig. 89 Sliding window performance improvement using sensor fusion.



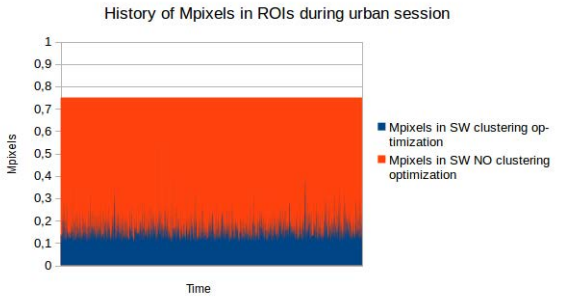
(a) History of SW to classify in a interurban session.



(b) History of Mpixels to classify in an interurban session.



(c) History of SW to classify classify in an urban session.



(d) History of Mpixels to classify in an urban session.

Fig. 90 Sliding window performance improvement using sensor fusion.

Window Overlapping

Local concentrations of clusters representing obstacles can determine overlapping ROIs for CV classification. If the distances from the sensor to the obstacles responsible for the clusters creation are similar and the obstacles are located close to each other, the overlapping windows can be merged in order to reduce the number of ROIs. This is, for example, the case of a pedestrian standing near to a traffic sign or a wall as in figure 91, where a pedestrian detection overlaps with the adjacent wall detection.

Fig. 91 ROI overlapping due to close detections (clusters are displaced to the left for better displaying).

Two rectangles are supposed to be equivalent if equation 24 is less than 0.5. In the case of the figure 91, as the distance to the detected clusters are very similar and the Overlapping Rate is less than 0.5, these ROIs are not considered individually, but merged into the union of both ROIs, obtaining great savings in computational effort, specially in urban areas where OOI tend to be located close to other obstacles.

$$OverlappingRate = \frac{Area(A \bigcap B)}{Area(A \bigcup B)} \tag{24}$$

Obstacle classifications using sensor fusion

Statistics for obstacle classification using sensor fusion have been obtained using labeled urban sessions and the database (see table 8) associated to every capture in the thesis. The sessions have been manually labeled for pedestrians,

cars, bicycles and motorbikes with the prerequisite that only OOI detectable by clustering must be labeled; i.e. laser occluded OOI and farther than the maximum detectable distance are not labeled. With the selected clustering distance and minimum number of points for clustering, pedestrians can be laser detected up to 25 meters following equation 1. With the intrinsic optical characteristics of the camera, a pedestrian located 25 meters away is captured at 40x65 pixels, far below the minimum CV-classifiable pedestrian of 64x128 pixels without the ROI magnification method used (The ROI is scaled x2 in order to be able to detect OOI smaller then the minimum detectable size). Bicyclists are not labeled as pedestrians. A database containing the labeled positions of the pedestrians in every image of the session is created for easier manipulation of the information.

Figure 92 illustrates the final sensor fusion classification process, in this case only for pedestrians. Blue squares are ROIs produced by laser cluster detections, which are represented by big red dots. These ROIs are searched using SVM for pedestrians, and the results are fused with cluster classification. The detected pedestrian is then surrounded by a red square.

Fig. 92 Example of pedestrian classification. Blue squares are obstacle detections, big red dots are clusters, red squares are sensor fusion pedestrian positive classifications.

Some special cases can be seen in figure 92; figure 93a shows a pedestrian detection and classification apparently with two different clusters, one on the chest and another in the feet. The cluster detection associated with this pedestrian is the one in the feet; the cluster in the chest corresponds with some other obstacle in the background, and the representation on top of this pedestrian is due to the different point of view of camera and laser. From the left side of the bumper, the laser can sense obstacles not visible for the camera, so the representation of these obstacles will occur in the part of the image associated to the cluster 3D converted to image coordinates, regardless the contents of the image. Figure 93b shows the opposite situation: The pedestrian in the foreground is occluding for the laser the pedestrian in the background, but not for the camera. In this case, the camera can see the pedestrian but it will not generate a cluster nor a ROI because the laser can not see her. Figure 93c shows a total occlusion for laser and partial for camera. The closest pedestrian is generating the cluster and the ROI, and the farthest

pedestrian will be included in the ROI, so it might be classified as a pedestrian by the SVM CV classifier, as well as the real detection.

(a) Cluster detection apparently over a pedestrian. This effect is produced by the different point of view of camera and laser.

(b) A pedestrian is occluded by another pedestrian for laser detection, so a ROI is not generated.

(c) A pedestrian is occluded by another pedestrian, both for laser detection and for CV classification.

Fig. 93 Special cases in detections.

The statistics obtained for this particular session are shown in figure 94 and 95. 542 pedestrians have been labeled in 235 urban environment 1024x768 pixels images from a sequence recorded in the IVVI 2.0 platform. In the session, 6,364 obstacle detections (cluster creations, implying clustering classification and ROI creation for CV classification) were computed. As seen in figure

94, 92% of the cluster classifications over the total labeled pedestrians are True Positives (TP), 11% of the cluster classifications over the total obstacle detections are False Positives (FP), and 7% of the cluster classifications over the total labeled pedestrians are False Negatives (FN). When using SVM classification of the whole image looking for pedestrians, 95% of the image classifications over the total labeled pedestrians are TP, 16% of the image classifications over the total total labeled pedestrians are FP, and 5% of the image classifications over the total labeled pedestrians are FN. When SVM classification is performed over the whole image, more FP are found because there are more shapes in the image similar to a pedestrian that would not appear in the laser generated ROIs. When data fusion is performed with laser ROI generation, cluster classification and CV classification, 94% of the image classifications over the total labeled pedestrians are TP, 4% of the image classifications over the total total labeled pedestrians are FP, and 5% of the image classifications over the total labeled pedestrians are FN. FP decrease in the case of sensor fusion because of the ROI generation, avoiding the search in the whole image.

These statistics demonstrate that laser and camera sensor fusion achieve a True Positive rate similar to the obtained with individual sensor classification (ISC), an improved False Positive rate with respect to ISC, and a False Negative rate similar to the lowest in the ISC case. Figures are improved using sensor fusion, with a highly significant lower computational cost, as shown in figures 88, 89 and 90.

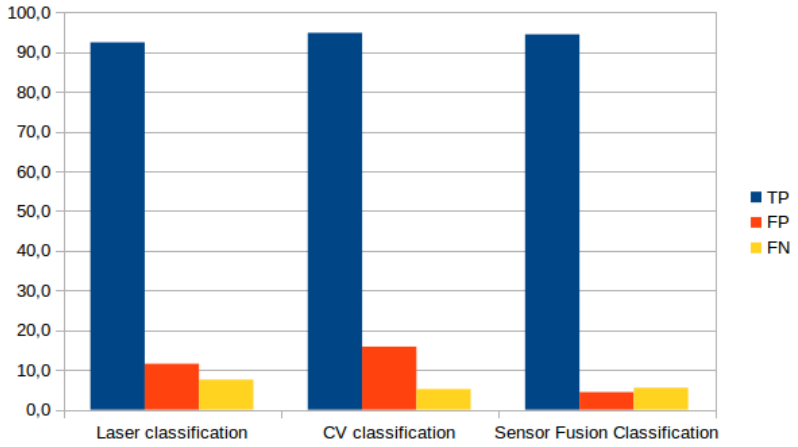


Fig. 94 Session classification statistics by type of classification method.

Figure 94 offers the same data, from the TP, FP and FN grouping perspective.

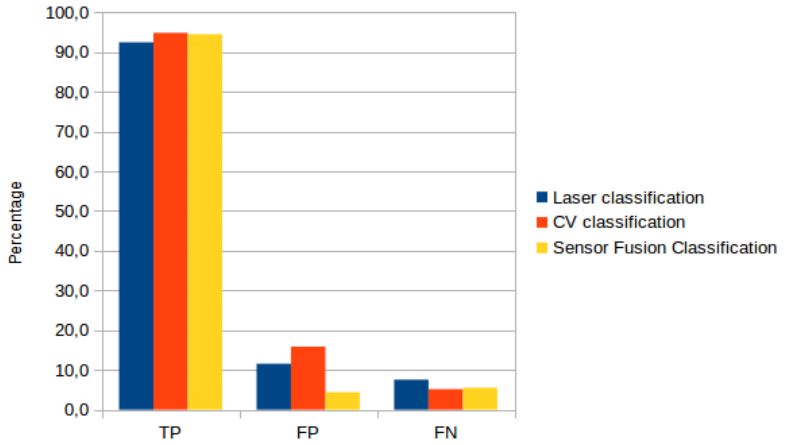


Fig. 95 Session classification statistics by TP, FP and FN.

Additionally, data from sequences without the particular OOI have been added to the previous data in order to study the distribution of the classification depending on the distance. Clusters are classified as OOI only when the confidence reaches 0.6.

As expected, accuracy is lower as the distance to the obstacle increases, because fewer points are included in the clusters and the information associated to the cluster is more restricted. Figure 96 shows that the confidence in TP cluster classification decreases with the distance. Figure 97 also shows that

FP cluster classification increases with the distance. Figure 98 shows a more even distribution of the true negatives, with less confidence in the classification associated to higher distances. Figure 99 shows also the the distance adversely affects the ability of the system to classify correctly the obstacles, with the lowest confidence rates in FN located in the farthest obstacles.

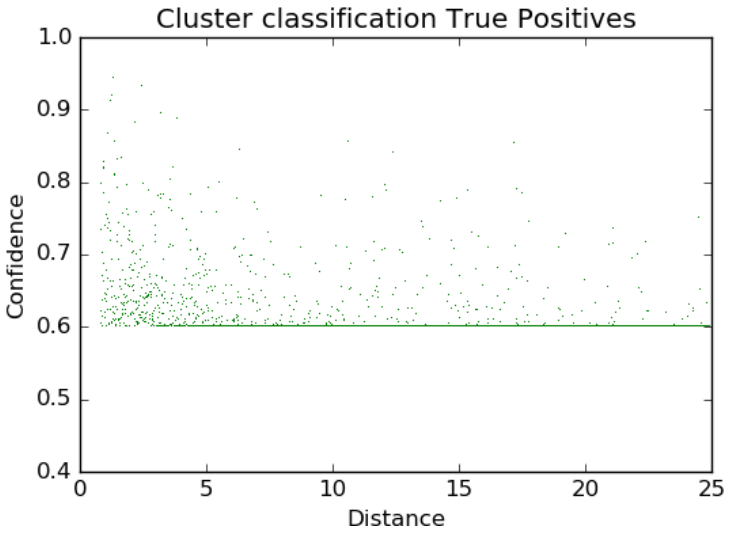


Fig. 96 Pedestrian True Positive detection using Cluster classification.

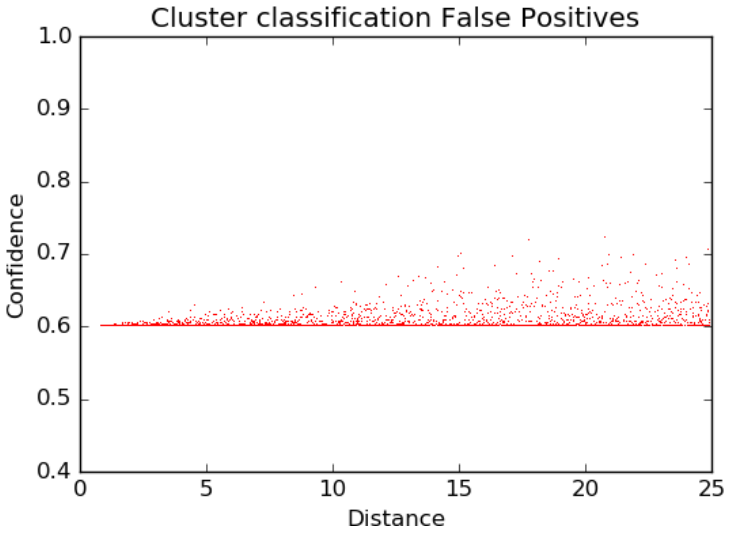


Fig. 97 Pedestrian False Positive detection using Cluster classification.

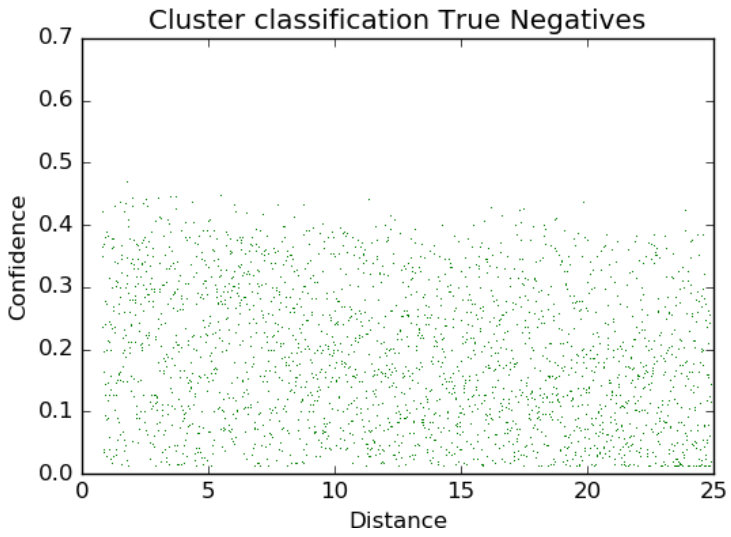


Fig. 98 Pedestrian True Negative detection using Cluster classification.

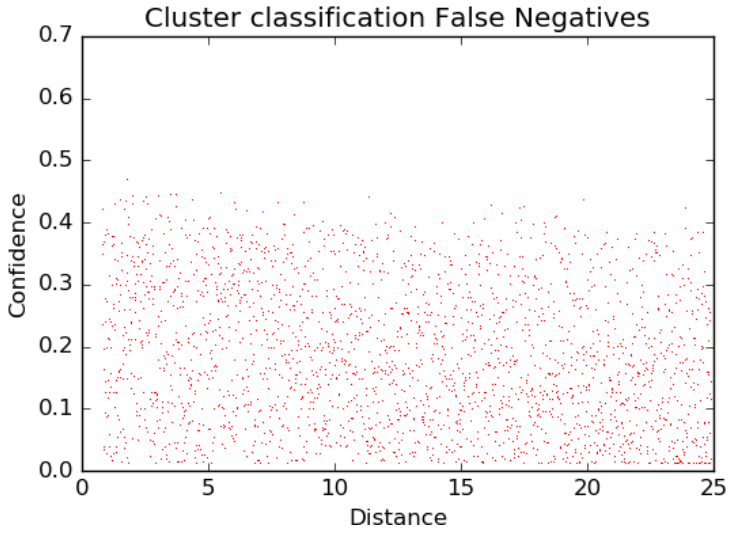


Fig. 99 Pedestrian False Negative detection using Cluster classification.

Chapter 7

Conclusions

7.1 Conclusions

The purpose of this thesis is the development of a sensor fusion system for laser scanner and camera, taking advantage of the strengths of each sensor and overcoming the weaknesses by the use of sensor fusion. The provided statistics demonstrate that a robust and reliable system has been developed using the sensors available at the time of the beginning of the thesis. The application is a good starting point for new researches using new affordable and powerful laser scanners available in the market during the last year of the thesis. The cluster classification research can be applied to new and more complete sensors, obtaining better and promising results.

7.2 Contributions

The presented thesis provides several contributions for different aspects of the ADAS research, introducing opportunities for future works:

- **The extrinsic parameters estimation system** for sensors providing point clouds, allowing data alignment in virtually any time without human intervention and without the need for a special shape, just a flat surface common to both sensors, such as the road. Extrinsic parameter

estimation is problem to solve in every sensor fusion system, and the presented algorithm is a step forward for automatic and unattended estimation in changing environments like real world from previous works such as [95] and [96].

- **The intelligent clustering algorithm for obstacle detection** overcomes the problem associated to multilayer laser scanner offering limited information about the detected obstacles. Adaptable euclidean distance and geometric addition of new points to existent clusters using RANSAC allows the creation of richer and more reliable representations of the obstacles than previous obstacle detection algorithms such as in [19]. Ground detection and removal from point cloud also improves the speed of data management and provides better clustering, providing clearer information about the detected obstacles.

- **The CROMA-LSI dataset** for use in domain adaptation systems and enriching real world datasets. The research in this area might improve the results of existing datasets and facilitate the creation of new classifiers for objects of interest difficult to acquire in real world. This dataset will be public, as well as the pedestrian clustering dataset used for pedestrian laser classification. Bicycles images dataset will also be public and free to use for research. Other works such as [78] and [18] use synthetic images for domain adaptation, while LSI-CROMA is using real world images for real world applications.

- **The classification algorithm for clusters** representing obstacles in the point cloud, using 3D and morphological information for obstacle classification. This research provides new possibilities to laser and camera sensor fusion, allowing high levels of reliability in bad illumination conditions, beyond the usual laser obstacle detection. Previous researches such as [14] and [19] provided obstacle detection and classification

for 2D point clouds; this thesis has extended cluster classification to three-dimensional point clouds.

- **The improved sliding window algorithm** involving sensor fusion and information from the cluster representing the obstacle, allowing a remarkable improvement (see figures 88, 89 and 90d) in the calculation effort for computer vision classification of the obstacle in the Region of Interest. This improvement allows a real time obstacle detection in real environments.

7.3 Future works

The research in the presented thesis uncovered several limitations in the systems and sensors used. The effort devoted to the research can be very profitable in future researches. The research lines that can be continued from the thesis could be:

- **Clustering extraction and classification using new laser sensors**. New multilayer laser scanners, recently available in the laboratory, offer crucial improvements in the quality of the information to be obtained from the obstacles, both for precise detection and for classification.
- **Domain adaptation and classification testing** using the CROMA-LSI dataset and the synthetic samples. New and improved CROMA datasets can be generated, and virtual worlds could be used for new classification algorithms and testing.
- **Novel classification techniques such as Deep Learning and CNN** could be used both for cluster classification and for image classification.
- **Stereo vision for point cloud generation** and merging with point cloud and depth information from the laser, and information fusion from

visible image and depth information could be used for improved obstacle extraction and classification.

- **Autonomous driving** The iCab (Intelligent Campus Automobile) platform from the Intelligent Systems Lab can use algorithms developed in this thesis for obstacle detection and classification in autonomous roaming in the UC3M campus in a near future.

References

[1] Eurostat. Transport accident statistics. http://ec.europa.eu/eurostat/statistics-explained/index.php/Transport_accident_statistics/, 2015. [Online; accessed 19-July-2016].

[2] The league of american biclysts. BICYCLE COMMUTING DATA. http://www.bikeleague.org/commutingdata, 2016. [Online; accessed 19-July-2016].

[3] NHTSA. NHTSA Traffic Safety Facts - Bicyclists and Other Cyclists. https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812282, 2016. [Online; accessed 19-July-2016].

[4] National Highway Traffic Safety Administration. Preliminary Statement of Policy Concerning Automated Vehicles. http://orfe.princeton.edu/~alaink/SmartDrivingCars/Automated_Vehicles_Policy.pdf, 2012. [Online; accessed 19-July-2016].

[5] Alain L. Kornhauser. National Highway Traffic Safety Administration statement of Policy Re: Automated Vehicles. http://orfe.princeton.edu/~alaink/SmartDrivingCars/CommentOnNHTSA_PrelimStatement.pdf, 2012. [Online; accessed 19-July-2016].

[6] IHS Makrit. ADAS- Current and Future Perspectives. https://www.ihs.com/pdf/IHS-ADAS-Current-and-Future-Perspectives_227834110913052332.pdf, 2015. [Online; accessed 19-July-2016].

[7] Don Sherman. Semi-Autonomous Cars Compared! Tesla Model S vs. BMW 750i, Infiniti Q50S, and Mercedes-Benz S65 AMG. http://www.caranddriver.com/features/semi-autonomous-cars-compared-tesla-vs-bmw-mercedes-and-infiniti-feature, 2016. [Online; accessed 19-July-2016].

[8] Google. Google Self-Driving Car Project. https://www.google.com/selfdrivingcar/, 2016. [Online; accessed 19-July-2016].

[9] Google. Google Self-Driving Car Project Monthly Reports. https://www.google.com/selfdrivingcar/reports/, 2016. [Online; accessed 19-July-2016].

[10] D. Martín, F. García, B. Musleh, D. Olmeda, G. Peláez, P. Marín, A. Ponz, C. Rodríguez, A. Al-Kaff, A. De La Escalera, and J. M. Armingol. IVVI 2.0: An intelligent vehicle based on computational perception. *Expert Systems with Applications*, 41(17):7927–7944, 2014.

[11] Daniel Olmeda, Cristiano Premebida, Urbano Nunes, Jose Maria Armingol, and Arturo De La Escalera. Pedestrian detection in far infrared images. *Integrated Computer-Aided Engineering*, 20(4):347–360, 2013.

[12] Aurelio Ponz, C. H. Rodríguez-Garavito, Fernando García, Philip Lenz, Christoph Stiller, and J. M. Armingol. Automatic obstacle classification using laser and camera fusion. In *Proceedings of the 1st International Conference on Vehicle Technology and Intelligent Transport Systems*, pages 19–24, 2015.

[13] Cesar H. Rodriguez Garavito, Aurelio Ponz, Fernando Garcia, David Martin, Arturo de la Escalera, and Jose M. Armingol. Automatic Laser And Camera Extrinsic Calibration for Data Fusion Using Road Plane. In *Proc. IEEE International Conference on Information Fusion (FUSION)*, 2014.

[14] Cristiano Premebida, Joao Carreira, Jorge Batista, and Urbano Nunes. Pedestrian detection combining RGB and dense LIDAR data. In *IEEE International Conference on Intelligent Robots and Systems*, pages 4112–4117, 2014.

[15] C Premebida and U Nunes. Fusing LIDAR, camera and semantic information: A context-based approach for pedestrian detection. *International Journal of Robotics Research*, 32(3):371–384, 2013.

[16] R. Fernandes, C. Premebida, P. Peixoto, D. Wolf, and U. Nunes. Road Detection Using High Resolution LIDAR. *2014 IEEE Vehicle Power and Propulsion Conference (VPPC)*, pages 1–6, 2014.

[17] Kiyosumi Kidono, Takeo Miyasaka, Akihiro Watanabe, Takashi Naito, and Jun Miura. Pedestrian recognition using high-definition LIDAR. In *IEEE Intelligent Vehicles Symposium, Proceedings*, pages 405–410, 2011.

[18] D Vazquez, A M Lopez, J Marin, D Ponsa, and D Geroimo. Virtual and Real World Adaptation for Pedestrian Detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(4):797–809, 2014.

[19] Fernando Garcia. *Data fusion architecture for intelligent vehicles*. PhD thesis, Universidad Carlos III de Madrid, 2012.

[20] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012.

[21] Frank Moosmann and Christoph Stiller. Velodyne SLAM. In *IEEE Intelligent Vehicles Symposium, Proceedings*, pages 393–398, 2011.

[22] an Steinberg and Cl Bowman. *Handbook of multisensor data fusion*. CRC Press LLC, 2001.

[23] Henrik Bostrom, Sten F Andler, Marcus Brohede, Ronnie Johansson, Alexander Karlsson, Joeri Van Laere, Lars Niklasson, Maria Nilsson, Anne Persson, and Tom Ziemke. On the Definition of Information Fusion as a Field of Research. *IKI Technical Reports*, pages 1–8, 2007.

[24] Wilfried Elmenreich. An introduction to sensor fusion. *Austria: Vienna University Of Technology*, 1(February):1–28, 2002.

[25] C A Fowler. Comments on the Cost and Performance of Military Systems. *Aerospace and Electronic Systems, IEEE Transactions on*, AES-15(1):2–10, 1979.

[26] Federico Castanedo. A review of data fusion techniques. *TheScientificWorldJournal*, 2013:704504, 2013.

[27] H. F. Durrant-Whyte. Sensor Models and Multisensor Integration. *The International Journal of Robotics Research*, 7(6):97–113, 1988.

[28] D L David L Hall, Senior Member, and James Llinas. An introduction to multisensor data fusion. *Proceedings of the IEEE*, 85(1):6–23, 1997.

[29] H. B. Mitchell. *Data fusion: Concepts and ideas*. 2012.

[30] Enrique David Martí, David Martín, Jesús García, Artur de la Escalera, José Manuel Molina, and José María Armingol. Context-aided sensor fusion for enhanced urban navigation. *Sensors (Switzerland)*, 12(12):16802–16837, 2012.

[31] F. Garcia, A. Ponz, D. Martín, J. M. Armingol, and A. De La Escalera. Laser Scanner and Computer Vision fusion for pedestrian detection in road environments. *RIAI - Revista Iberoamericana de Automatica e Informatica Industrial*, 12(2):218–229, 2015.

[32] Texas Instruments. Paving the way to self-driving cars with advanced drver assistance systems. http://www.ti.com/lit/wp/sszy019/sszy019.pdf, 2015. [Online; accessed 19-July-2016].

[33] *Laser scanner and camera fusion for automatic obstacle classification in ADAS application*, Communications in Computer and Information Science, Switzerland, 2015. Springer.

[34] Texas Instruments. Advanced Driver Assistance (ADAS) Solutions Guide. http://www.ti.com/lit/sl/slyy044a/slyy044a.pdf, 2015. [Online; accessed 19-July-2016].

[35] Velodyne. Hdl-64ehigh definition real-time 3d lidar, 2016. original document from Velodyne.

[36] Velodyne. Velodyne lidar puck, 2016. original document from Velodyne.

[37] J Behley, V Steinhage, and A B Cremers. Efficient radius neighbor search in three-dimensional point clouds. *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 3625–3630, 2015.

[38] F. Garcia, A. Ponz, D. Martin, J. M. Armingol, and A. De La Escalera. Fusion de Escaner Laser y Vision por Computador para la Deteccion de Peatones en Entornos Viarios. *RIAI - Revista Iberoamericana de Automatica e Informatica Industrial*, 12(2):218–229, 2015.

[39] Jorg Kibbel, Winfried Justus, and Kay Furstenberg. Lane estimation and departure warning using multilayer laserscanner. In *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, volume 2005, pages 777–781, 2005.

[40] Jan Sparbert, Klaus Dietmayer, and Daniel Streller. Lane detection and street type classification using laser range images. In *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, pages 454–459, 2001.

[41] K. Takagi, K. Morikawa, T. Ogawa, and M. Saburi. Road Environment Recognition Using On-vehicle LIDAR. *2006 IEEE Intelligent Vehicles Symposium*, pages 120–125, 2006.

[42] T. Ogawa and K. Takagi. Lane Recognition Using On-vehicle LIDAR. In *Intelligent Vehicles Symposium*, pages 540–545, 2006.

[43] Manolis Tsogas, Nikos Floudas, Panagiotis Lytrivis, Angelos Amditis, and Aris Polychronopoulos. Combined lane and road attributes extraction by fusing data from digital map, laser scanner and camera. *Information Fusion*, 12(1):28–36, 2011.

[44] Chieh Chih Wang, Charles Thorpe, and Arne Suppe. LIDAR-based detection and tracking of moving objects from a ground vehicle at high speeds. In *IEEE Intelligent Vehicles Symposium, Proceedings*, pages 416–421, 2003.

[45] Michael Montemerlo, Sebastian Thrun, Daphne Koller, Ben Wegbreit, and Others. FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *Aaai/iaai*, pages 593–598, 2002.

[46] R. De Maesschalck, D. Jouan-Rimbaud, and D.L. L Massart. The Mahalanobis distance. *Chemometrics and Intelligent Laboratory Systems*, 50(1):1–18, 2000.

[47] Nicolas Picard and Avner Bar-Hen. A Criterion Based on the Mahalanobis Distance for Cluster Analysis with Subsampling. *Journal of Classification*, 29(1):23–49, 2012.

[48] Shiming Xiang, Feiping Nie, and Changshui Zhang. Learning a Mahalanobis distance metric for data clustering and classification. *Pattern Recognition*, 41(12):3600–3612, 2008.

[49] Igor Melnykov and Volodymyr Melnykov. On K-means algorithm with the use of mahalanobis distances. *Statistics and Probability Letters*, 84(1):88–95, 2014.

[50] Daniel Olmeda. *Pedestrian Detection in Far Infrared Images*. PhD thesis, Universidad Carlos III de Madrid, 2013.

[51] Boguslaw Wiecek Dariusz Rzeszotarski. Calibration for 3D Reconstruction of Thermal Images . In *9th International Conference on Quantitative InfraRed Thermography*, 2008.

[52] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[53] Aravindhan K Krishnan. Cross-Calibration of RGB and Thermal cameras with a LIDAR. In *IROS 2015 Workshop on Alternative Sensing for Robot Perception*, 2015.

[54] Rongqian Yang, Wei Yang, Yazhu Chen, and Xiaoming Wu. Geometric Calibration of IR Camera Using Trinocular Vision. *J. Lightwave Technol.*, 29(24):3797–3803, 2011.

[55] Thomas Luhmann, Johannes Piechel, and Thorsten Roelfs. Geometric Calibration of Thermographic Cameras. *Thermal Infrared Remote Sensing*, 17(1):27–42, 2013.

[56] J Rangel, S Soldan, and A Kroll. 3D Thermal Imaging: Fusion of Thermography and Depth Cameras. *Conference on Quantitative InfraRed Thermography*, 2014.

[57] Clancy Soehren. Ultrasonic sensors push the limits of automotive applications, 2014.

[58] Gustavo Pelaez. *Monitoring the driver's activity using 3D information*. PhD thesis, Universidad Carlos III de Madrid, 2015.

[59] H. Gonzalez-Jorge, P. Rodriguez-Gonzalvez, J. Martinez-Sanchez, D. Gonzalez-Aguilera, P. Arias, M. Gesto, and L. Diaz-Vilarino. Metrological comparison between Kinect i and Kinect II sensors. *Measurement: Journal of the International Measurement Confederation*, 70:21–26, 2015.

[60] Sebastian Budzan and Jerzy Kasprzyk. Fusion of 3D laser scanner and depth images for obstacle recognition in mobile applications. *Optics and Lasers in Engineering*, 77:230–240, 2016.

[61] Georgios Mastorakis and Dimitrios Makris. Fall detection system using Kinect's infrared sensor. *Journal of Real-Time Image Processing*, 9(4):635–646, 2014.

[62] Texas Instruments. Time-of-Flight Camera – An Introduction. http://www.ti.com/lit/wp/sloa190b/sloa190b.pdf, 2014. [Online; accessed 19-September-2016].

[63] H Gonzalez-Jorge, P Rodríguez-Gonzálvez, J Martínez-Sánchez, D González-Aguilera, P Arias, M Gesto, and L Díaz-Vilariño. Metrological comparison between Kinect I and Kinect {II} sensors. *Measurement*, 70:21–26, 2015.

[64] ST. Radar Based Adas. http://www.st.com/content/st_com/en/applications/automotive-and-transportation/active-and-passive-safety/radar-based-adas.html, 2016. [Online; accessed 19-September-2016].

[65] P Viola and M Jones. Rapid object detection using a boosted cascade of simple features. *Computer Vision and Pattern Recognition (CVPR)*, 1:I—-511—-I—-518, 2001.

[66] Navneet Dalal. Finding People in Images and Videos. *I Can*, page 149, 2006.

[67] T Ojala, M Pietikainen, and D Harwood. Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In *Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision amp; Image Processing., Proceedings of the 12th IAPR International Conference on*, volume 1, pages 582–585 vol.1, 1994.

[68] Junge Zhang, Kaiqi Huang, Yinan Yu, and Tieniu Tan. Boosted local structured HOG-LBP for object localization. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1393–1400, 2011.

[69] Xin Yuan, Xiaosen Shan, and Li Su. A combined pedestrian detection method based on Haar-like features and HOG features. In *2011 3rd International Workshop on Intelligent Systems and Applications, ISA 2011 - Proceedings*, 2011.

[70] G Monteiro, P Peixoto, and U Nunes. Vision-Based Pedestrian Detection Using Haar-Like Features. *Robotica 24 (2006)*, pages 46–50, 2006.

[71] P F Felzenszwalb, R B Girshick, D McAllester, and D Ramanan. Object Detection with Discriminative Trained Part Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[72] Kelly Assis de Souza Gazolli and Evandro Ottoni Teattini Salles. Using holistic features for scene classification by combining classifiers. 2013.

[73] Hamed Masnadi-Shirazi, Vijay Mahadevan, and Nuno Vasconcelos. On the design of robust classifiers for computer vision. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 779–786, 2010.

[74] Jiaolong Xu, David Vazquez, Antonio M. Lopez, Javier Marin, and Daniel Ponsa. Learning a part-based pedestrian detector in a virtual world. *IEEE Transactions on Intelligent Transportation Systems*, 15(5):2121–2131, 2014.

[75] Javier Marin, David Vazquez, Antonio M. Lopez, Jaume Amores, and Bastian Leibe. Random forests of local experts for pedestrian detection. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[76] David Forsyth. Object detection with discriminatively trained part-based models. *Computer*, 47(2):6–7, 2014.

[77] Hironori Hattori, Vishnu Naresh Boddeti, Kris Kitani, and Takeo Kanade. Learning scene-specific pedestrian detectors without real data. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 07-12-June-2015, pages 3819–3827, 2015.

[78] G Ros, L Sellart, and J Materzynska. The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes. *Proceedings of the*, 1(600388):4321–4330, 2016.

[79] Fan Zhu and Ling Shao. Weakly-Supervised Cross-Domain Dictionary Learning for Visual Recognition. *International Journal of Computer Vision*, 109(1):42–59, 2014.

[80] Stefano Debattisti, Luca Mazzei, and Matteo Panciroli. Automated extrinsic laser and camera inter-calibration using triangular targets. In *IEEE Intelligent Vehicles Symposium, Proceedings*, pages 696–701, 2013.

[81] Davide Scaramuzza, Ahad Harati, and Roland Siegwart. Extrinsic self calibration of a camera and a 3D laser range finder from natural scenes. In *IEEE International Conference on Intelligent Robots and Systems*, pages 4164–4169, 2007.

[82] Clearpath Robotics. ROS 101: Intro to the Robot Operating System. http://robohub.org/ros-101-intro-to-the-robot-operating-system/, 2014. [Online; accessed 4-October-2016].

[83] Sick. Ld-mrs laser measurement sensor, 2011. original document from Sick.

[84] MicroStrain. 3dm-gx2 gyro enhanced orientation sensor, 2007. original document from MicroStrain.

[85] David Martin Juan Carmona, Fernando Garcia and Jose Armingol. Data Fusion for Driver Behaviour Analysis. *Sensors*, 15(10):25968–25991, 2015.

[86] Fernando Garcia Juan Carmona et al. Embedded system for driver behavior analysis based on GMM. In *2016 IEEE Intelligent Vehicles*, 2016.

[87] Fernando Garcia Juan Carmona et al. Analysis of Aggressive Driver Behaviour Using Data Fusion. In *VEHITS2016*, 2016.

[88] Miguel Angel de Miguel Paraiso. Desarrollo de herramienta para comunicación con vehículo a través de can-bus. Master's thesis, Universidad Carlos III de Madrid, 2015.

[89] C.Cortes, C.Cortes, V.Vapnik, and V.Vapnik. Support Vector Networks. *Machine Learning*, 20(3):273˜–˜297, 1995.

[90] Jesus Vicente Lopez. Diseno e implementacion de un sistema de deteccion de peatones basado en hog y svm. Master's thesis, Universidad Carlos III de Madrid, 2016.

[91] Fernando Garcia, David Martin, Arturo De la Escalera, and Jose Maria Armingol. Sensor Fusion Methodology for Vehicle Detection. In *IEEE Intelligent Transportation Systems Magazine (Volume: 9, Issue: 1, Spring 2017)*, 2017.

[92] ROS.org. Wiki ROS. http://wiki.ros.org/, 2016. [Online; accessed 19-November-2016].

[93] Martin a. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[94] Philip H. S. Torr and Andrew Zisserman. MLESAC: A New Robust Estimator with Application to Estimating Image Geometry. *Computer Vision and Image Understanding*, 78(1):138–156, 2000.

[95] You Li, Yassine Ruichek, and Cindy Cappelle. 3D triangulation based extrinsic calibration between a stereo vision system and a LIDAR. In *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, pages 797–802, 2011.

[96] Kiho Kwak, Daniel F. Huber, Hernan Badino, and Takeo Kanade. Extrinsic calibration of a single line scanning lidar and a camera. In *IEEE International Conference on Intelligent Robots and Systems*, pages 3283–3289, 2011.