Development of a Light-Based Driver Assistance System

Von der Fakultät für Maschinenbau
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des akademischen Grades

Doktor-Ingenieur

genehmigte Dissertation
von

M.Sc. Hatem Shadeed

geboren am 10.10.1976 in Kalyobiya/ Ägypten

2012

1. Referent:	Prof. Dr.-Ing. Jörg Wallaschek
2. Referent:	Prof. Dr. rer. nat. Cornelius Neumann
3. Vorsitz:	Prof. Dr.-Ing. Lutz Rissing

Tag der Promotion: 14.11.2012

Kurzfassung

Die Verbesserung der Fahrsicherheit bei Dunkelheit hat in den letzten Jahren viel Aufmerksamkeit im Forschungsgebiet der Fahrerassistenzsysteme erlangt. In der vorliegenden Arbeit wird ein lichtbasiertes Fahrerassistenzsystem vorgestellt, das auf Basis von Umfeldsensorik und eines fusionierten Umfeldmodells die Fahrbedingungen bei Dunkelheit durch gezielte situationsangepaßte Beleuchtung des Vorfeldes deutlich verbessert. Die Verbesserung der Beleuchtung wird durch zwei Module erreicht. Die Markierungslicht-Funktion hebt alle relevanten Objekte und Lebewesen, die den Fahrer gefährden könnten, wie z.B. ein unvorsichtiger Fußgänger, der eine befahrene Straße überquert, mit einem zusätzlichen Lichtstrahl hervor. Die Fernlichtassistenz-Funktion, auch als "blendfreies Fernlicht" bezeichnet, blendet gezielt andere vorausfahrende oder entgegenkommende Verkehrsteilnehmer aus, so dass der Fahrer immer unter optimalen Lichtbedingungen fährt, ohne Andere zu blenden.

Objekterkennung und Situationsinterpretation sind Eckpfeiler in Fahrerassistenzsystemen. Eine deutliche Verbesserung im Bereich der aktiven Sicherheitssysteme jedoch setzt eine Leistungssteigerung der Erfassungssysteme der Fahrzeugumgebung voraus, was ein breiteres Sichtfeld, eine höhere Präzision und eine bessere Zuverlässigkeit der Umfeldsensoren beinhaltet. Dies kann über eine Verbesserung der vorhandenen Sensoren oder auch durch Fusionieren der Informationen von verschiedenen Sensoren erreicht werden. Diese Arbeit stellt eine Sensorfusions-Plattform vor, die die Informationen von verschiedenen Komponenten zu einem Umfeld- und Situationsmodell zusammenführt. Die erste Komponente ist ein Vision-Sensor, mit dem die Lichtquellen der entgegenkommenden sowie vorausfahrenden Fahrzeuge detektiert werden. Die zweite Komponente ist ein LIDAR-System, das Objekte und Hindernisse vor dem Versuchsträger erfasst. Die dritte Komponente ist ein GPS-Empfänger, der die Position, den Kurswinkel und die Absolutgeschwindigkeit des Versuchsträgers ermittelt. Die vierte Komponente ist ein Fahrzeug-zu-Fahrzeug-Kommunikations-Modul, das Daten über die relative Positionierung zwischen dem Versuchsträger und den anderen Straßenteilnehmern ermittelt. Die auf dem CAN-Bus verfügbaren internen Fahrzeugdaten über den Status und die Dynamik des Versuchsträgers bilden schließlich die fünfte Komponente.

Die hier vorgestellten Methoden und Techniken zur Sensorfusion sind allgemein und können auch bei anderen sicherheitsrelevanten Systemen angewendet werden so eine vollständige Repräsentation des Verkehrsraumes des Fahrzeugs erforderlich ist. Die Algorithmen und Verfahren wurden mit realen Sensoren-Messungen aus verschiedenen Verkehrssituationen und Umgebungen verifiziert.

Schlagworte: lichtbasiertes Fahrerassistenzsystem, Objekterkennung, Datenfusion

Abstract

Night driving safety has recently attracted much attention in the driver assistance system research field. In this thesis, a Light Based Driver Assistance System (LBDAS) is presented. This system aims at increasing the comfort and safety of the driver by enhancing the driving conditions, through actively interacting with the environment of the host vehicle. The main contribution of the proposed system goes into two directions. The first direction is the marker light function, in which all the relevant objects and entities that may endanger the driver, like a careless pedestrian crossing the driving road, will be highlighted with an extra light beam. The other direction is a high beam assistance function, which is also called “glare-free high beam”. This function enables the driver to use a permanent high beam light distribution to illuminate the road; meanwhile the headlamps controller prevents glaring the other drivers by shutting off the light in their corresponding areas.

Object recognition and situation interpretation are key roles in the driver assistance systems. In general, the currently available object detection sensors for monitoring the driving environment are the main reason behind the limitation of using the driver assistance systems. Basically because they provide a very limited amount of the information which is necessary to manage higher level driving tasks. Studies and research programs show that achieving significant progress in automobile active safety technology requires a considerable increase in the performance of the driving environment monitoring systems. This includes wider range, higher precision, and better reliability information from the sensor. This can be achieved via improving the existing sensors (LIDAR, radar, laser scanners, and vision systems) as well as by fusing the information from different sensors. This research presents an application based on a data fusion model to handle the sensors constraints as well as to fulfill real-time requirements.

The data fusion platform is structured from different types of components. The first component is a vision sensor that is used to detect the light sources of the oncoming as well as vehicles ahead. The second component is a LIDAR system that detects the entire objects and obstacles in front of the host vehicle. The third component is a GPS receiver which finds the position of the host vehicle, in terms of longitude and latitude coordinates, in addition to other information such as the heading angle and absolute velocity. The fourth component is a Car-to-Car communication module that wirelessly exchanges data between the host and other vehicles. The last component is the on-board vehicle sensors which supply the other modules with information about status and dynamics of the host vehicle such as ego-velocity, yaw-rate, and steering angle.

The presented methods and techniques are general, and can be applied to other safety systems, whenever a complete representation of the vehicle surroundings is required, including information about objects on the road and driving path estimation together with the relative position of vehicles and infrastructure. All algorithms and methodologies developed in the thesis have been tested using real sensors data from various traffic situations and environments.

Keywords: Light Based Driver Assistance System, Object recognition, Data fusion

Contents

Contents v

1 Background and Motivations **1**
1.1 Introduction 1
1.2 Safety and Lighting Technology 2

2 Research Analysis **7**
2.1 Related Work 7
2.2 Objectives of the Research 11
2.3 Thesis Organization 11

3 Technical Approach **13**
3.1 System Requirements Analysis 13
3.2 Hardware State of the Art 14
3.3 Solution Description 22
3.4 Conceptual Design of LBDAS in Mechatronics Discipline 22
3.5 System Hardware Components 24
3.6 System Design 30
3.7 Data Acquisition Subsystem 32

4 Environment Perception **33**
4.1 Camera Object Recognition 33
4.2 IDIS Object Recognition 51
4.3 Vehicle Detection Via Car-to-Car Communication 57
4.4 Lane Estimation using GPS and Digital Maps 58

5 Sensors Data Fusion **65**
5.1 Object Data Fusion 65
5.2 Lane Fusion 82

6 Situation Analysis and Threat Assessment **95**
6.1 Parameters for Estimating the Base Light Distribution 96
6.2 Parameters for Estimating the Object Illumination Strategy 97
6.3 Decision Making 99
6.4 Illumination Modeling and Headlight Control 101

7 System Evaluation and Results **103**
7.1 Recognition Range 103

7.2 Object Classification Performance 107
7.3 Performace Evaluation of Different System Configurations 110

8 Conclusion and Future Work 113
8.1 Conclusion 113
8.2 Future Work 115

Bibliography 117

Chapter 1

Background and Motivations

1.1 Introduction

Since the introduction of the transport vehicles in 15th century and especially in the recent time, safety equipment are one of the important and main parts in the vehicle's development process. These requirements stem from the fact that road accidents are considered as one of the main causes of death of young people. In Europe, the numbers reach to 1,700,000 accidents, causing over 40,000 deaths and more than 1,300,000 injuries each year [Com01]. The investigations of the European Road Safety Action Program show that this high number of car crashes can be reduced by more than 50 percent by following three approaches. Firstly, by encouraging drivers to behave in a safe manner through training, penalties and harsher policing. Secondly, by improving road infrastructure. Thirdly, by making vehicles safer through improving the active and passive safety measures and technologies. The focus of this thesis lays on some of the aspects of the third approach, and more precisely on the improvement of the driver's visibility conditions at night.

Over the last few decades, traffic safety has become an important concern for the automotive industry. Safety evaluation organizations, such as European New Car Assessment Program (Euro NCAP), provide customers with information about the safety ratings for different models. A few years ago, these rankings were only based on the vehicle's passive safety considering components such as airbags, seat belts, and crumple zone[1]. The success of the passive safety features is impressive. The number of serious injuries in Germany has been reduced to one half and the number of road deaths down to one-third of the 1970 statistics (the year with the highest number of fatalities [BB]), despite the fact that the total distance driven is more than doubled.

Recently, automotive manufacturers as well as governmental authorities and research institutes are investigating and putting much efforts in the active safety; believing that these kind of safety systems can and will decrease the number of accidents, especially fatal accidents. Active safety systems, also known as Advanced Driver Assistance Systems (ADAS), are a collection of integrated electronic components designed to help the driver in confusing and difficult traffic situations. Their main target is to reduce traffic accidents via supporting the driver to take fast and efficient decisions in complex scenarios. Another aspect of the ADAS is to increase the comfort of the driver through reducing the work-load of routine tasks. For example, automatic beam selection system can relieve the driver of the need to manually select and activate the

---

[1] The crumple zone of an automobile is a structural feature designed to compress during an accident to absorb energy from the impact.

correct beam as traffic, weather, and road conditions change. ADAS is interacting with the driver through producing warning signals (e.g. Front Collision Warning, Lane Departure Warning and Night Vision), or by providing warning and guidance information (e.g. Blind Spot Detection), or even by taking action independently through overriding the driver commands to avoid an accident or at least to minimize the consequences of the accident (e.g. Collision Mitigation and Pre-Crash).

Driving at night time can be considered as a great threat to any vehicle's driver. The studies of the Germany's Federal Statistics Bureau [Bur02] showed that more than 40 percent of all automobile accidents causing death occur at night, despite the fact that there is up to 80 percent less traffic on the road than during the day. Driving at night is not only a threat to the driver but also to the pedestrians on the road. The statistical data of the National Highway Traffic Safety Administration[2] (NHTSA) showed that more than 69 percent of pedestrian fatalities occurred at night time. Figure 1.1.1 shows pedestrian fatalities by time of day. The analysis of these traffic fatalities reverts that to the poor visibility at night, the adverse weather, and the bad road conditions as well. With these premises, it is obvious that any effort in reducing the cause of accidents and/or reducing the effects of accidents at night and in particular increasing the safety of the pedestrians is highly appreciated.

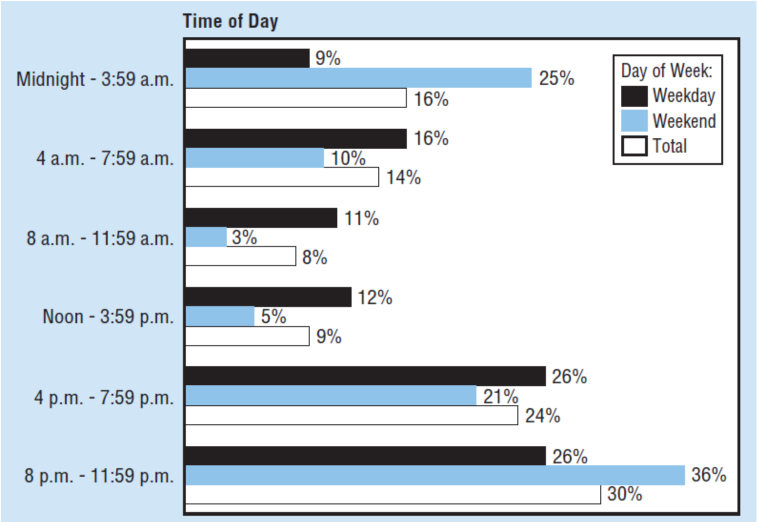

Figure 1.1.1: Pedestrian Fatalities by Time of Day and Day of Week [Tra06]

The proposed system in this thesis addresses this problem by trying to put guidelines to a headlight system, which improves the driver visibility, warns the driver of obstacles on the driving road, and warns the pedestrian of threat probability of the oncoming vehicle.

1.2 Safety and Lighting Technology

Vision plays a very essential role in human's life, especially while driving automobiles. Researchers proved that humans perceive more than 90 percent of their information by vision [BBF00]. In typical traffic situations the contrast sensitivity and visual acuity as well as the

---

[2] www.nhtsa.dot.gov

speed of perception and recognition of danger are strongly dependent on the ambient light distribution. Internal effects such as tiredness and external such as weather conditions can influence the driver's vision indicators drastically. This makes it difficult to identify objects at short distances, read traffic signs, locate lane as well as pavement markings, spot other vehicles as well as pedestrians, and in general driving at night due to the glare of other vehicles headlights. Studies [FD06, OS86] have shown that the Perception Reaction Time (PRT), under normal driving conditions for a normally alert driver is typically around 1.5 to 1.75 seconds. During nighttime driving, the perception time can be increased due to many contributing factors, resulting in a PRT of up to 2.5 seconds or more, depending on the conditions.

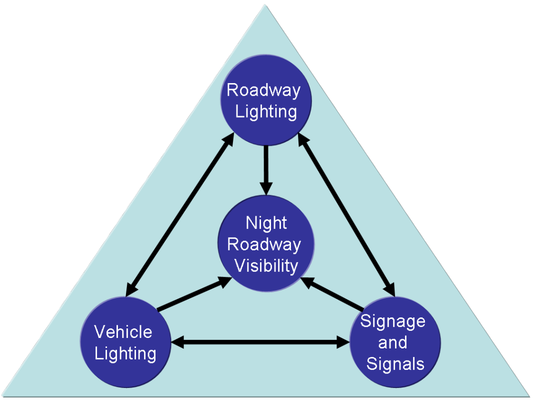

Figure 1.2.1: Main Parameters Affecting the Roadway Visibility at Night

One of the most important tasks of the automotive lighting technology is to support human perception at night and under adverse weather conditions. The easiest way to improve the visibility is to produce a lot of light along the road ahead. However, the problem with this approach is that strong light can dazzle the other road users, reducing their view of the road. So far, a compromise had been adopted to resolve this conflict, which was in the form of introducing the high beam and dipped beam light distributions. The dipped beam is a non-dazzling light distribution, which illuminates about 60 m in front of the vehicle. On the other side, the high beam is a bright light distribution which illuminates more than 300 m efficiently in front of the vehicle and may glare oncoming traffic up to 800 m. According to ECE[3] national regulations, high beam is allowed to be used only when there is no threat to glare other drivers; otherwise dipped beam should be used. Recently, the number of vehicles and their speed-limits are increased rapidly resulting in a more complicated traffic situation, which had made the classical light distributions inadequate. Thus concepts of new light distribution have been introduced. These concepts concern either by modifying the dipped beam to gain more visibility distance or by adopting new technique's domain such as invisible light area like infrared light beams, for example. In the next section an overview of the state of the art in the automobiles lighting technologies will be presented.

1.2.1 Night Vision Systems

In contrast to the normal light distributions, which illuminate directly the road in front of the vehicle, night-vision systems are based on displaying infrared images on visualization device, almost a monitor. They are acquiring information about the objects, which the driver cannot

---

[3] Economic Commission for Europe

see with a dipped beam light distribution. Night-vision system can be classified as either passive or active [BBK^+03, LKVB03, SM04]. The passive systems rely on a far-infrared (FIR) detector to sense thermal radiation from the objects in front of the car. However, active systems operate in the near infrared (NIR) and use an extra infrared source to illuminate the road ahead.

1.2.1.1 Passive Night Vision

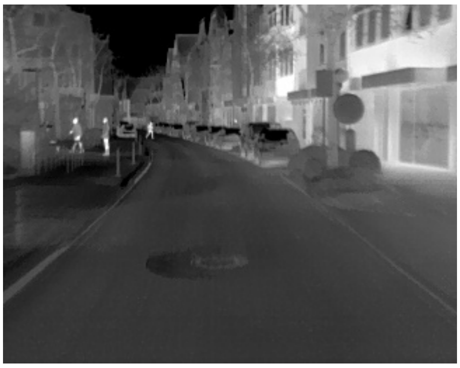

Figure 1.2.2: Passive Night Vision

The main advantage of passive systems is their ability to differentiate warm objects, such as pedestrians and animals, from a cold background providing a visibility distance up to 300 m [OM99, OJG10]. Although passive systems are particularly effective during the dark winter months, they can hardly detect pedestrians with heavy clothes, cold vehicles, and cold obstacles, e.g., big stones on the road [Gro]. However, these objects can produce a real threat to the driver.

1.2.1.2 Active Night Vision

The alternative to FIR systems is the active night-vision. Active systems use cameras (CCD or CMOS) to image a scene illuminated with an NIR radiation source in the range 780-1000 nm [WWBH07]. They can support the driver with a visibility range of up to 150 m [OGWM94]. These systems eliminate the drawbacks of passive night vision but it also produces new challenges. For example, the active night vision can be glared from oncoming IR-Radiator of other vehicles.

1.2.2 Advanced Front Lighting System (AFS)

AFS functions are considered as one of the main revolutions in the automobile lighting technologies in the last two decades. Many studies claimed that the AFS can improve the illumination of the road in front of the vehicle extremely by splitting the low-beam function into different light distributions [Voe05b, LV04, BL07a]. They are offering the driver an optimal light pattern in nearly every situation. In the next section the main AFS functions will be highlighted.

1.2.2.1 Road Dependent Light Distribution

These light distributions are targeting to provide optimal illumination in various driving conditions by automatically modifying the dipped beam pattern of the headlight system in response to vehicle speed and road situations [WWBH07]. These modifications can take one or even a combination of the following forms:

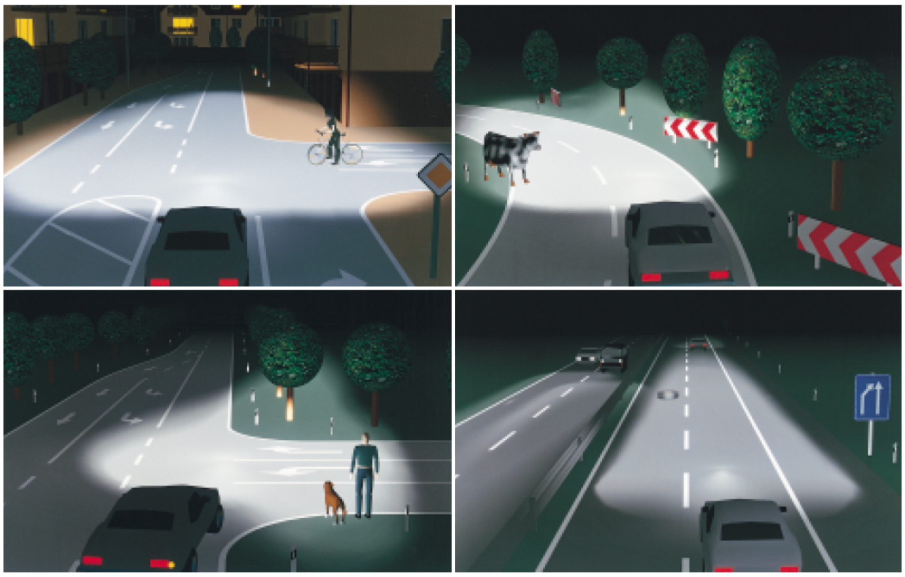

Figure 1.2.3: Examples Of Adaptive Light Distribution: Town Light (Top Left), Dynamic Curve Light (Top Right), Static Curve Light (Bottom Left), Motorway Light (Bottom Right). Photo: Hella

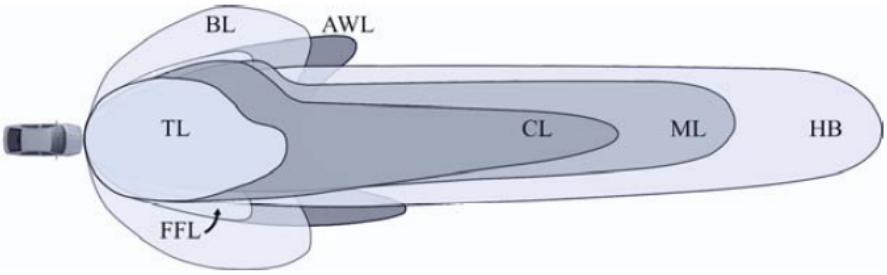

Figure 1.2.4: CL – Country Light, ML – Motorway Light, HB – High Beam, FFL – Front Fog Light, AWL – Adverse Weather Light, TL – Town Light, BL – Bending Light [WWBH07]

1. Country mode: the dipped beam is distributed in such a way that the left edge of the road is illuminated more brightly and widely. Therefore, the driver's range of vision is increased by around ten meters.

2. Town mode: is a symmetrical wide light distribution. In low speed situations the beam has the maximum spread, which leads to an optimized orientation in the outside areas. By increasing the driving speed the spread becomes smaller to enable the headlight to project more light in the driving lane.

3. Motorway mode: is a uniform cone of light, which illuminates the entire road width to a range of up to 120 meters enabling the driver to gain up to 60 percent better visibility than classical dipped beam.

1.2.2.2 Adverse Weather Light Distribution

Dense fog or heavy rains situations are the worst cases for the driver and the other road participants where they get the lowest level of information. In such cases, drivers can lose their orientation quickly because of the absence of familiar guiding features such as the road markings and the road edges. Thus, AFS provides a compromise between the ability to be seen, the visibility distance, and the effects of self and reflex glare.

1. Enhanced fog lamps: its main task is to project a broad beam in such a way that the road edges in particular are well illuminated. Therefore, the nearside of the road is illuminated more efficiently, improving the driver's visibility in this area. At the same time this wider light distribution reduces back glare in foggy conditions. This lighting function is automatically activated as soon as the rear fog lamp is switched on as long as the ego-velocity is under specific limits.
2. Rain mode: is an asymmetric beam with high visibility distance on the driving lane and reduced reflex glare for the oncoming traffic.
3. Heavy Rain mode: is light distribution similar to the town light

1.2.2.3 Static Bending Light

Static bending light is an extra fixed light source in the headlight that turned on in low speed sharp turns. It is switched on automatically in presence of high steering wheel angle, low vehicle velocity, and the confirmation of the turn signal switch.

1.2.2.4 Dynamic Bending Light

It is also known as "Active Cornering Headlight System". In contrast to the static bending light, dynamic system features beam patterns which follow the driver's steering movements and swivel in the driving direction - almost immediately - as the vehicle enters a curve. The illumination range with classical dipped beam in curve of radius 190 m is about 30 m, while it is 55 m when using dynamic bending light. It means that this function improves the illumination of the curvature lane by up to 90 percent. It is worth mentioning here that the dynamic bending light can be extended by coupling it with a lane curvature information, GPS information, and digital maps [WWBH07, EWG01] to produce a predictive dynamic bending light distribution.

1.2.2.5 Automatic Leveling Headlight Systems

They use a control module as well as various sensors for vehicle pitch and yaw rates to automatically adjust the angle of the headlights in response to changes in the road vertical curvature [Kuh06]. The headlight projector unit is swiveled up or down by an electric motor. This keeps the light beam at the same angle to the road surface during acceleration, braking, body pitch, and body roll due to changing road surfaces.

Chapter 2

Research Analysis

2.1 Related Work

2.1.1 Light-Based Driver Assistance Systems

Intelligent Lighting Systems or Light-Based Driver Assistance Systems (LBDAS) are a new trend in the automotive lighting technology [WLS06]. They go further behind the circumscription of AFS. The system is so intelligent that it can automatically adapt its light distribution not only to the road and weather conditions, but also to the corresponding traffic situation. LBDAS are based mainly on the interaction of modern remote sensors, powerful software for signal processing, and state-of-the-art headlight technology.

2.1.1.1 High-Beam Assistance System

A scientific study in the United States has shown that drivers underuse their high beams in circumstances where their use is highly recommended [SAMF03]. High-Beam Assistance System (HBAS) aims to enable the driver to use the high beam more efficiently. In September 2005, BMW had introduced the HBAS in the 5 Series, 6 Series, and 7 Series automobiles. The system switches the headlights from high beam to dipped beam as soon as it detects oncoming traffic or adequate street lighting. A camera integrated in the rear-view mirror identifies the headlights and rear lights of vehicles, as well as the ambient brightness. When the road ahead is clear, the system automatically switches to high beam and vice versa.

2.1.1.2 Adaptive Cut-Off-Line System

Adaptive Cut-Off-Line (ACOL) adjusts continuously the light distribution such that the driver has good visibility with the longest possible range [BL07b]. This is achieved through the adaptation of the headlight range to preceding or oncoming motor vehicles as shown in top view of Fig. 2.1.1. This means that the dipped beam does not stop as usual at around 60 meters on the oncoming lane anymore, but it is increased to gain several hundred meters.

The bottom view of Fig. 2.1.1 shows that the headlight cone always ends at nearest vehicle to avoid dazzling the other road users. If the vision system does not detect any vehicles on the road, the illumination range extends automatically to provide the driver with light up to the high beam level. Once the system detects other road traffic, the range of the headlights is adapted accordingly.

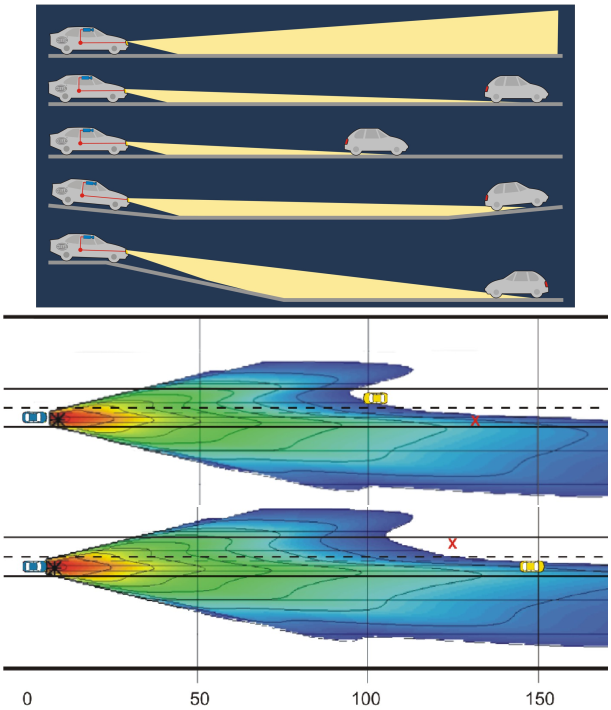

Figure 2.1.1: Light Distributions of The Adaptive Cut-Off-Line System [Hel06]

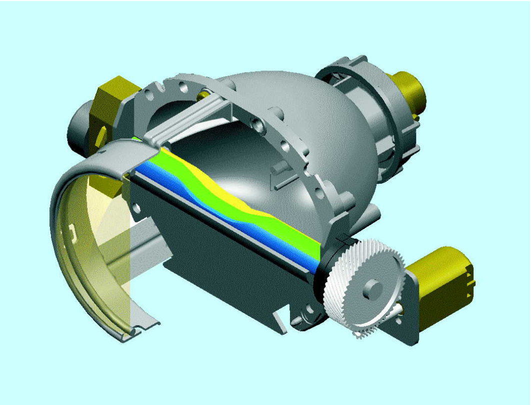

Figure 2.1.2: Construction of VarioX Headlight Module [Hel06]

There are various technical solutions to implement the above mentioned light function, for example by using Hella VarioX-Module. This module is based on a free form rotating drum which composed of different shapes. By changing the position of those shapes via rotating the drum as well as swilling the module, a high resolution adapted light distribution can be generated.

2.1.2 Active Lighting System

Active Lighting System (ALS) has been investigated intensively by Hella KGaA and Paderborn university in their joint research center L-LAB. The system detects the objects on the road and reacts by producing various adaptive light distributions. The ALS project has been under consideration for more than eight years and has resulted in a new thinking trend in producing intelligent headlight systems. As the name "Active Lighting" suggests, such a system interacts automatically with the environment through different sensors and it processes the collected data in a way well-aimed to detect all the relevant objects that may endanger the driver, threaten his life, or the life of the any other road users. Although Adaptive Cut-Off-Line (ACOL) System can be considered as the most intelligent LBDAS in the market so far, it cannot produce the optimum light distribution in all the traffic situations. The investigations in [RW04, SHS$^+$06, SW06, SW07b, WKE03, RW03] showed that controlling only the horizontal-cut-off-line (HCOL) of the projected light by modifying its position is inadequate to provide the driver with the suitable light distribution in all traffic scenarios. ACOL would increase the safety of the pedestrian at the side of the road, but it would not really benefit the pedestrian crossing the road. Thus, the project Active Lighting has adopted a new concept of modifying the HCOL. The idea behind the new concept is based on the fact that it is not necessary to modify all the light distribution along the HCOL when a relevant object is detected, but it is totally sufficient to modify only the area where the object is found. For this reason a new methodology to deal with the objects on the road has been implemented. Hence in other words, ALS is an object dependent lighting system; for example, objects like vehicles will be blinded out, and obstacles on the driving lane will be highlighted (marked), and pedestrians will be warned, simultaneously. The concept of ALS has been introduced and discussed in many publications. The first prototype illustrated in [Ros05] had proposed a new light function called "Glare-Free High Beam", which enables the driver to use a permanent high-beam to recognize a wide range of road, taking into consideration not glaring other drivers.

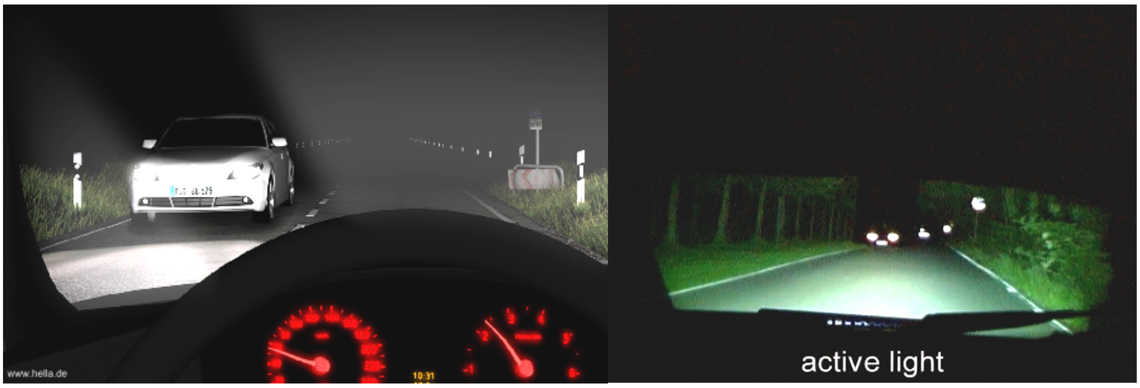

Figure 2.1.3: Left:Concept of The Glare-Free High Beam, Right: Output of The Prototype

Figure 2.1.3 shows the concept of the glare-free high beam. As can be seen in the figure, the illumination in the area of the oncoming vehicle is reduced to avoid dazzling the driver, however the left and the right areas of the vehicle is illuminated efficiently with high beam. This prototype is based on Lidar sensor (see Section 3.2.1.2) and a Digital Micro-Mirror Device headlight (DMD) (see Section 3.2.2.1). The quality and the safety gain of the new light distribution had been evaluated from test-persons and it had shown a very good results as well as a high customer acceptance. The main drawback of this prototype is a very short range of its object recognition module.

As shown in Fig. 2.1.4, the driver of oncoming vehicle percepts the ALS light distribution as a dipped beam as long as his vehicle is within the detection range of the system's sensor, otherwise

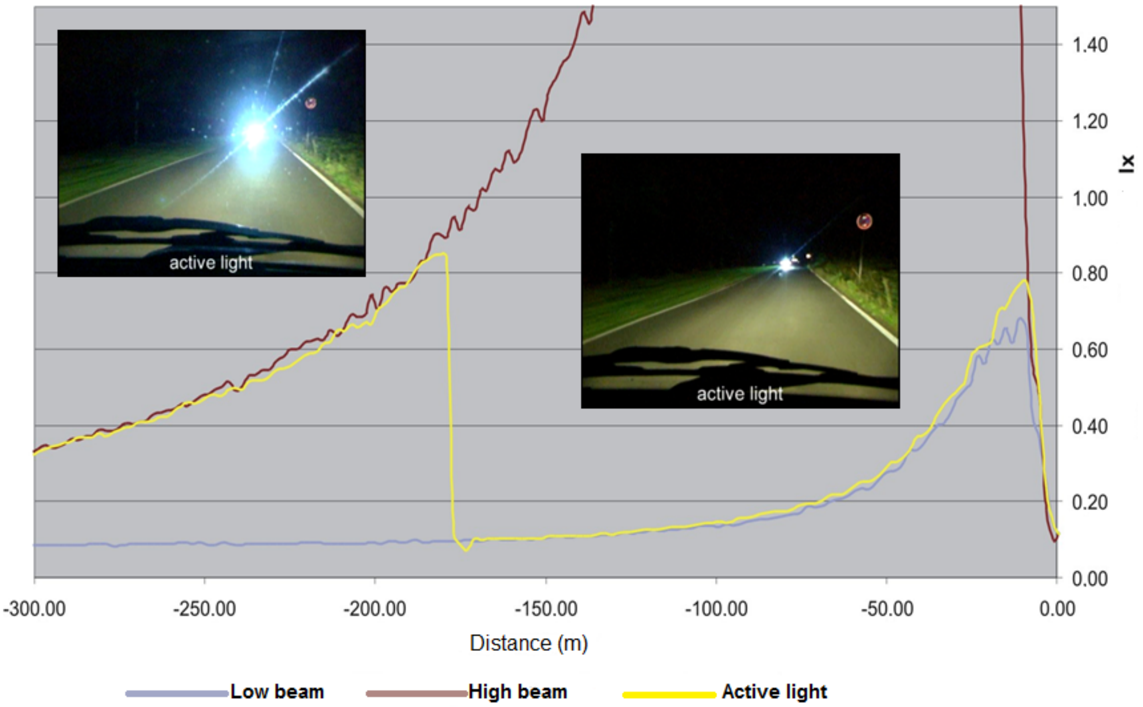

Figure 2.1.4: Light Distributions of The ALS Prototype

he percepts it as a high beam. Thus, one of the main targets of the current research is to explore new techniques to increase the range. Also a new light function “Marker Light” will be integrated.

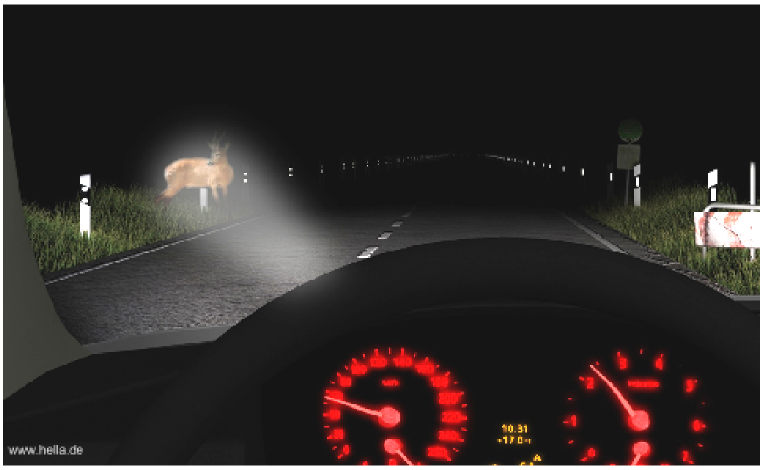

Figure 2.1.5: Concept of The Marker Light

Figure 2.1.5 demonstrates the concept of the marker light. The new light distribution aims to draw the driver’s attention to any hazard potential as early as possible by directing marker lights to objects that are of particular relevance for the visual perception of the driver. Marker light is intended to be used where the glare free high beam is not permissible to be used; for example, on the poorly illuminated city roads.

2.2 Objectives of the Research

The overall objective of this research is to develop a platform for vehicle space recognition using different types of techniques within the "Active Lighting" project. Other goals are to develop a situation analysis methodology that allow to choose the optimal light distribution that suites the current situation. In addition, this research aims to investigate if the Hella ACC system prototype (IDIS) can provide the basis for the new light function, namely "marker light".

The main tasks of the thesis can be summarized in the following points:

- Analyzing the system requirements to gain an overview about the required hardware.
- Selection of the hardware components of the system.
- Preparing a test vehicle for data acquisition.
- Development of a data acquisition platform, to acquire data for developing the algorithms and for the testing purposes.
- Design of a hardware architecture.
- Design of a software architecture.
- Development of the necessary algorithms for object recognition and for controlling of the headlamp as well, to be integrated in the system.
- Implementation of the algorithms in the Visual C++.
- Testing and evaluating the system in different traffic situations.

2.3 Thesis Organization

After introducing the main goals, the motivation, and the objectives of the thesis in the first two chapters, the technical approach is highlighted in chapter 3. The implementation of the algorithms for environment perception and sensors data fusion are presented in chapter 4 and 5, respectively. Chapter 6 illustrates the methodologies used in the situation analysis, object threat assessment, and the control strategies of the headlamp. The system is evaluated and the obtained results are discussed in chapter 7. Finally, in chapter 8, the work in the thesis is summarized and the suggestions for the future work are proposed.

Chapter 3

Technical Approach

3.1 System Requirements Analysis

3.1.1 Object Recognition

1. The system should be able to detect and classify different types of objects, which may be found on the driving road and can lead to an accident, such as vehicles, bulky obstacles, and pedestrians. Also it should be able to detect those objects which are not visible to the driver, i.e., they are not in the illumination range of the headlights.

2. The recognition distance for oncoming traffic should be more than 800 m and for leading vehicles more than 400 m [BL07b].

3. Azimuth angle should be more than ± 15° [Hel06].

4. The time between detection of a vehicle to the headlight control should be less than 1 second [BL07b].

5. The results published in [Kos03] showed that at 64 km/h driving speed the perception distance at which detection criteria is satisfied to non-expectant dark-clad pedestrian crossing a road from left-to-right is about 22 m and from right-to-left is about 36 m. Thus, to gain a benefit from the LBDAS the system should guarantee, for non-vehicle objects, at least a detection distance more than double of the maximum perception distance (i.e., more than 72 m).

6. For the marking function, the time latency from detection to the headlight control should be less than 100 ms (2.8 m at 100 km/h relative velocity).

7. To increase the reliability and performance of the system, it should be able to distinguish between visible and non-visible objects to avoid unnecessary marking/glaring to objects which are visible to the driver; for example a pedestrian wearing reflecting clothes or a vehicle standing on the road side and its warning lights are on.

8. The system should be able to assign the objects to the road lanes in order to determine precisely the relevant objects and to avoid the misuse of the light function.

3.1.2 Headlight

1. It should be capable of producing free programmable light distributions.
2. It should guarantee a fast response time, the recommended time for switching from the dipped beam to full high beam should be less than 50 ms [Ros05].
3. Basic light distributions (high and dipped beam) should be always available and should confirm the ECE and SAE standards.
4. For integration compatibility, the interface between the headlamp and the system's control unit should support one of the common vehicle's communication buses (CAN or FlexRay).

3.1.3 Processing Capacity

The system should be able to process the sensors' information and to take the required decisions in real time.

3.2 Hardware State of the Art

As stated before, the main components of any LBDAS system are the environment sensing devices which constitute the vehicle remote-sensors and the light producing equipment incorporated in the headlights. In the following section, the state of the art of those components will be highlighted in some of details.

3.2.1 Remote Sensors

A key role in any driver assistance system is obstacles detection and situation interpretation. Such systems can only support the driver intelligently if they have access to the sensors which supply them with adequate information about the vehicle itself, the road condition, and particularly objects as well as obstacles in the vehicle environment. A modern vehicle has many internal sensors which are widely used at many places and serve as a corner stone for many safety systems like ABS and ESP.

On the other side are external sensors or remote-sensors as shown in Fig. 3.2.1. They are used to detect the presence of the objects close to the vehicle and to give information about the vehicle's traffic space. Usually, a vehicle remote-sensor is a device which collects data about real-world conditions, processes the received data in real time, makes a decision, and sends commands to vehicle subsystems to help preventing an accident, mitigate its severity, or protect the vehicle's occupants. Intelligent safety systems are demanding different types of sensors with different working-principal for different applications, and in some cases, multiple sensor types for the same application. In the automotive field so far, the maximum detection range of the current long-range remote-sensors is up to 200 m, which can be achieved via 77 GHz radar or lidar. However, Laserscanners provide the maximum lateral scanning opening angle.

3.2.1.1 RADAR

Radar is an acronym for RAdio Detection And Ranging. The radar device emits a microwave signal and observes the echoes returned from the objects [Sko89]. The elapsed time between emission and return is a function of the distance of the object from the radar device. The

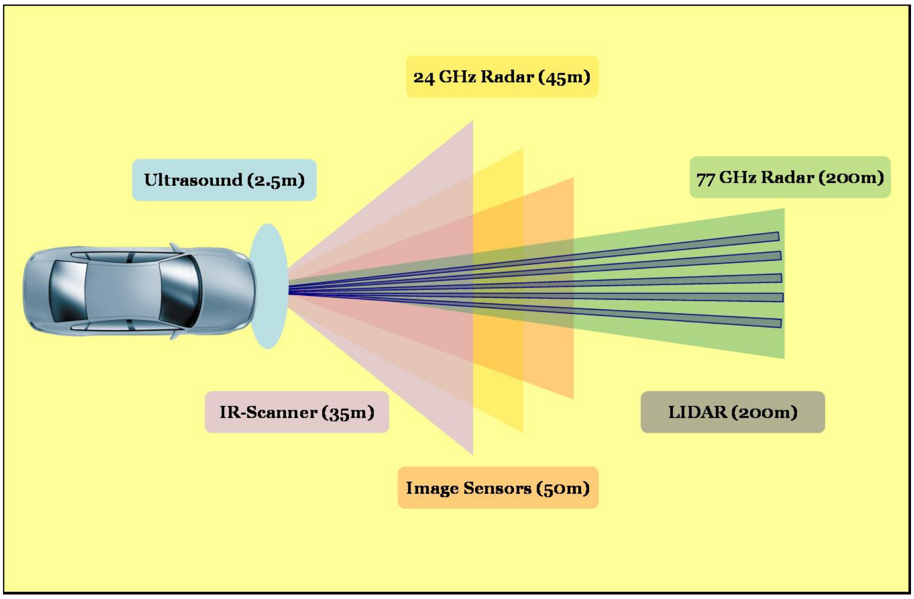

Figure 3.2.1: Automotive Remote-Sensors [Mah06]

distance can be computed using equation (3.2.1). However, the speed and direction of a moving object can be determined by analyzing the shift in the frequency of the microwave signal using Doppler effect as shown in equation (3.2.2). The use of microwave radar sensors for obstacle detection in the near vicinity around the vehicle has advantages in special situations like bad weather, poor visibility or harsh environmental impacts like ice, snow or dust coverage [LHF+08].

$$Distance = \frac{\Delta T.C}{2} \tag{3.2.1}$$

$$Velocity = \frac{\lambda.F_{dp}}{2} = \frac{C.F_{dp}}{2F_t} \tag{3.2.2}$$

where: ΔT is the elapsed time between the transmitted signal and the received echo, C is the velocity of the microwaves, λ is the wave length of the transmitted signal, F_{dp} is the Doppler frequency, and F_t is the transmission frequency.

3.2.1.2 LIDAR

LIDAR (LIght Detection And Ranging) sensor is a modern opto-electronic measurement technology based on the principle of measuring the time of flight [HLL05]. It can be considered as a low-cost alternative solution for the radar systems. Lidar has successfully been used as a range sensor mounted on vehicles for the purpose of detection of other vehicles as well as pedestrians. This is done by performing an analysis of the light echoes reflected by the object. On the basis of the known propagation rate of light in a given medium, it is possible to calculate the distance to an object through the measurement of its propagation time. A concentrated short duration infrared light pulse with a high power is generated by a laser diode and is transmitted in the direction of the object. The distance to the recognized object can be computed using the same equation given in (3.2.1). However, the object's velocity can be determined by derivation of the distance. For simplification, the velocity can be considered as the change of the position, $\triangle d$, over a specific period of time, $\triangle T$, as given by equation (3.2.3).

$$Velocity = \frac{\triangle d}{\triangle T} \tag{3.2.3}$$

To achieve high lateral resolution and a wide horizontal opening angle, the lidar sensor has been designed as a multi-channel device. Since lidar does not use the principle of Doppler effect for range measurements, it does not require the object that be measured to move with high velocity. Hence, for the detection of slow moving and stationary objects at the city roads, lidar is an ideal sensor [HLL05].

3.2.1.3 Laserscanner

Laserscanners are the 2d-counterparts to the long-used fixed-beam range finders [FDEW02, FDL03, WFD01]. This system uses the same method of the lidar to estimate the distance and the velocity of the object. It consists of one or more EDM (Electronic Distance Measurement) unit combined with a beam deflection unit (Fig. 3.2.2). The EDM unit combines an InGaAs laser diode for pulse emission with a detector diode.

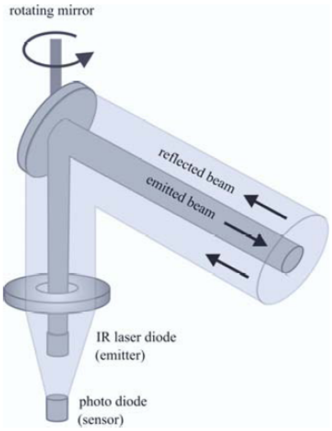

Figure 3.2.2: Construction of Laserscanner [WWBH07]

The laser diode emits pulses in the near infrared. To cover a viewing angle of up to 270° around the sensor, the laser beam is deflected by a rotating prism in the sensor head. The angular resolution is mainly limited by the maximum laser pulse frequency. Current technology allows 0.25° angular resolution at 10 Hz scan frequency or 1° angular resolution at 40 Hz scan frequency.

3.2.1.4 Camera

Charge coupled device cameras (CCD) and complementary metal oxide semiconductor (CMOS) are two main types of imaging chips considered when it comes to choose the digital camera for a vision system design [FP02]. A digitized image is a two-dimensional picture generated in a form which can be stored and processed by the computer. The images are typically produced at a 25-30 Hz frame rate and are stored as a matrix in the computer memory which can be processed, displayed, and searched for recognizable features using a model-specific filter. Maximum-response locations are used as the initial search points for the model-matching process, possibly supported by other features such as shape, symmetry or the use of a bounding box. Vision systems are exceeding the other object detection systems in its ability to provide information about the environment texture and objects class. Since vision systems are based on a passive sensor, they have not considerable limitations comparing with Lidar and Radar which must work in a specific emitting power range. Actually, the main limitation of the vision system is the capacity of its processing unit, which nowadays is not a problem anymore. A significant advantage of such systems is usage of stereo-imaging techniques which allows constructing 3D maps of the objects.

Another important property of cameras is their ability to capture the colors, which can be used to isolate the objects from the background. In the computer field, there are two well-known color systems which are used to represent the color images shown in Fig 3.2.4.

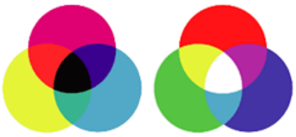

Figure 3.2.3: Color Systems: Left Subtractive, Right Additive

The first one is subtractive color system, which is used in printing technology. Subtractive color mixing means that one begins with white and ends with black; as one adds color, the result gets darker and tends to black. The CMYK color system is an example of such systems. The second system is additive color. Additive color mixing begins with black and ends with white; as more color is added, the result is lighter and tends to white. RGB is a good presenter to the additive color system. In image processing, RGB is widely used to extract the color information from the image. For digital cameras, in order to capture the real color of an object, three sensors should be used. Each sensor is dedicated to capture a specific color. However, its technically and economically difficult to produce a camera with three sensors. Therefore, most of the color cameras in market has one sensor and a so-called Bayer mosaic filter shown in Fig 3.2.4 referring to Bryce Bayer patent in 1976. It is a color filter array for arranging RGB color filters on the square grid of the photo-sensors in form of 50% green, 25% red, and 25% blue.

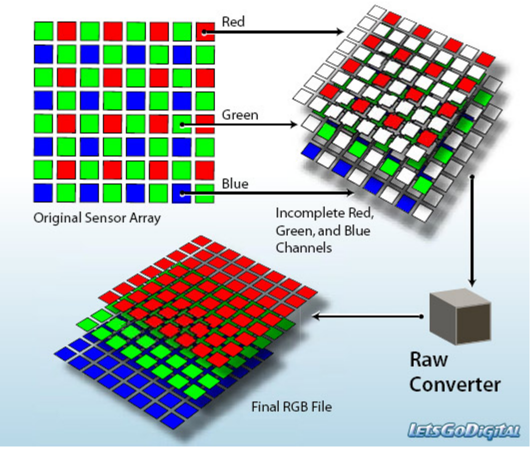

Figure 3.2.4: The Bayer Arrangement of Color Filters on the Pixel Array of an Image Sensor

The Bayer filter uses twice as many green elements as red or blue to mimic the physiology of the human eye. The retina has more rod cells than cone cells and rod cells are most sensitive to green light. Since each pixel is filtered to record only one of three colors, the data from each pixel cannot fully determine color on its own. To obtain a full-color image, various algorithms can be used to interpolate a set of complete red, green, and blue values for each point.

3.2.2 Addressable Headlights

The current headlight modules for realizing AFS allow only a fixed number of light distributions and do not fulfill the demands as well as prerequisites to generate complex assisting light functions under all circumstances. For this reason, new headlight systems have to be developed which can generate freely adaptable light distribution; for example, illuminating an object on the road such as a pedestrian and at the same time able to reduce the illumination on another object such as a vehicle [KEW04, Voe05a]. In the following section, three active headlight concepts will be presented and their functional principals will be explained.

3.2.2.1 Digital Micro-Mirror Device headlight

The core of a Digital Micro-Mirror Device (DMD) headlight is an array of micro aluminum mirrors that reflect the light of a high intensity light source. Such a micro-mirror array is composed of hundreds of thousands of mirrors with an edge length of about 13 µm mounted on small hinges atop a CMOS device [DD05, GE05].

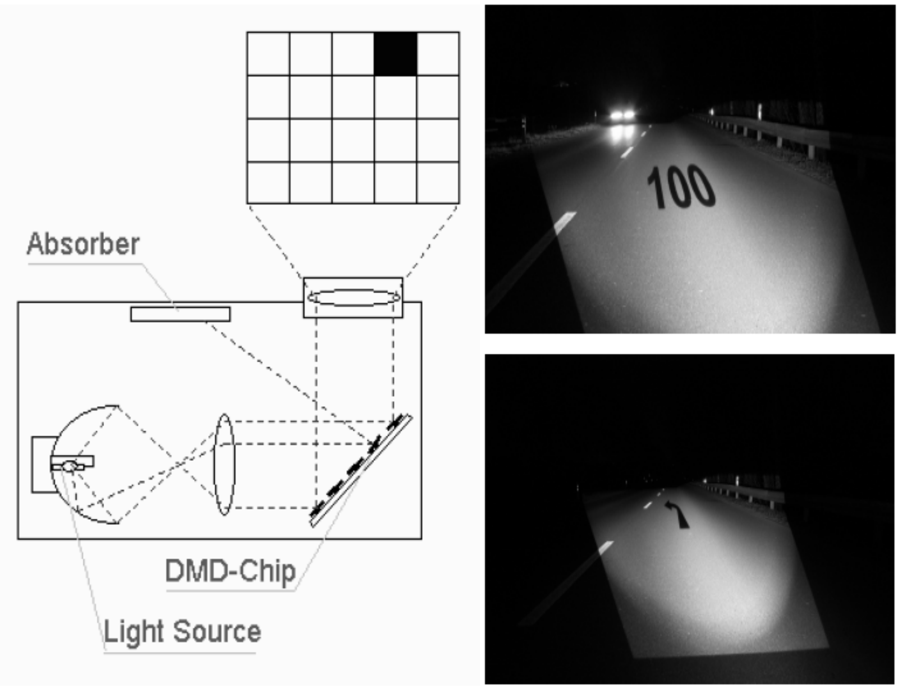

Figure 3.2.5: Left: Working principle of a DMD headlight. Right: Light distribution with display function [KEW04]

The individually addressable mirrors can be tilted between two positions. Either the light is directed through a projection lens to illuminate a target area or it is guided towards an absorbing surface. Therefore, a DMD device allows the generation of a picture consisting of light and dark pixels. Installing such a device as a headlight not only allows the generation of a large variety of light distributions, it also permits the display of characters and symbols on the street, as shown in Fig. 3.2.5.

3.2.2.2 LED-Array Headlight

In contrast to DMD, the LED-array headlight does not need moving elements to generate different light distributions. Instead, the light sources are addressed directly [Str05]. The light distributions are generated by creating an image of a matrix of LED-chips. The possibility of individually controlling each LED-chip of the matrix allows the generation of different shapes of light. Activating or deactivating single LED-chips of the matrix can easily realize assisting light functions [GE05]. Using a PWM (Pulse Width Modulation) to drive the LED-chips makes it possible to produce different levels of brightness, which allows adjusting the light intensity according to the road illumination. Activating single chips that contribute to the light distribution above the cut-off line could be used to realize a marking light function.

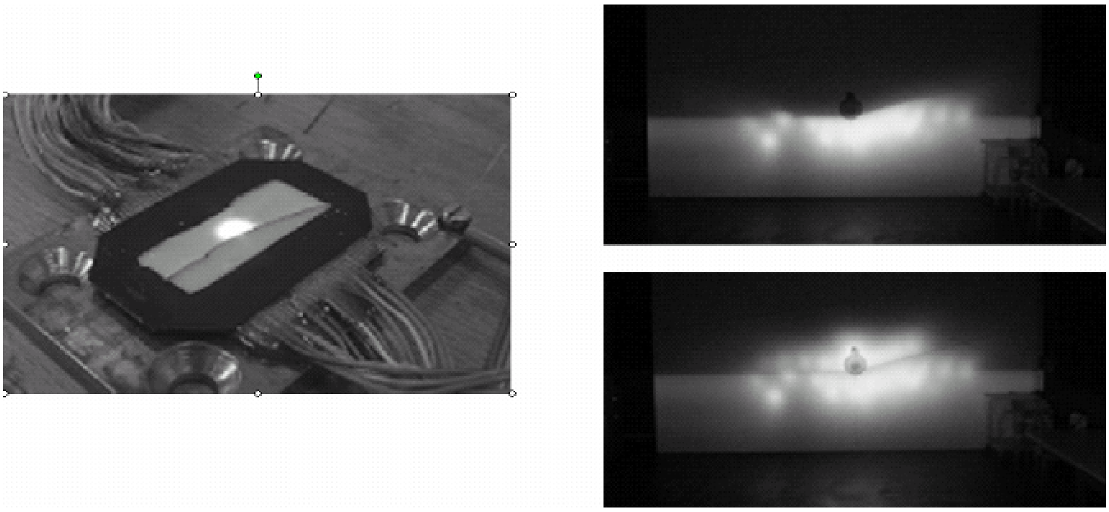

Figure 3.2.6: Left: LED-Array. Right: Exemplary Low-Beam and High-Beam Light Distribution Generated with The LED-Array

Fig. 3.2.6 depicts a LED-array prototype as well as a low-beam and high-beam light distribution generated with this array. The possibility of separately controlling the brightness of each LED-chip of the matrix allows the generation of driver-specific light distributions.

3.2.2.3 Segmented Shutter Headlight

The above mentioned headlight concepts are of interest for future work, but today they are much more expensive and less efficient than conventional headlight systems [GE05]. In order to get an affordable headlight with the characteristics mentioned previously, a system based on a conventional HID headlight was developed. It consists of a projector module and a variable shutter, which replaces the shield of the projector module [MS07].

Figure 3.2.7 shows a prototype of the high variable shutter. Every segment of the shutter can move separately up and down with a step width of about 10µm, resulting in 1000 vertical steps.

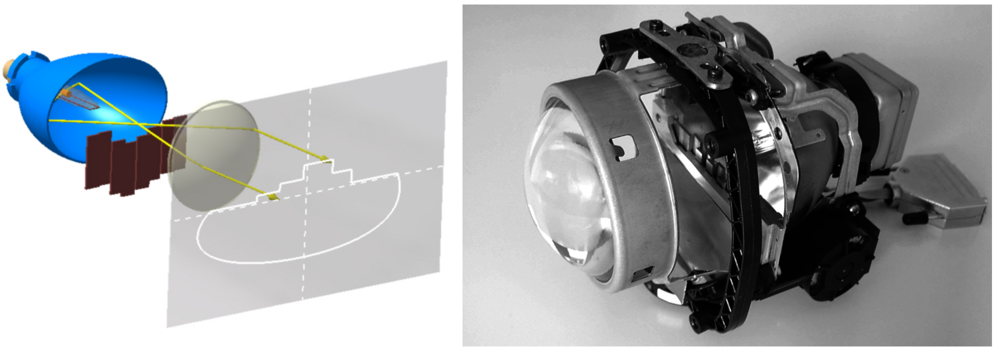

Figure 3.2.7: Segmented Shutter Headlight Construction

Thus arbitrary light distributions can be formed using the upper outline of the shutter segments. The horizontal resolution is limited by the number of shutters. In the current prototype, nine segments have been used with unequal width for the outer and inner shutters. The tight shutters are placed in the middle of the headlight, which is in accordance with the prospective projection principle considering the fact that distant oncoming traffic appears near the center of the 2D projection map of the road as shown in Fig. 3.2.8.

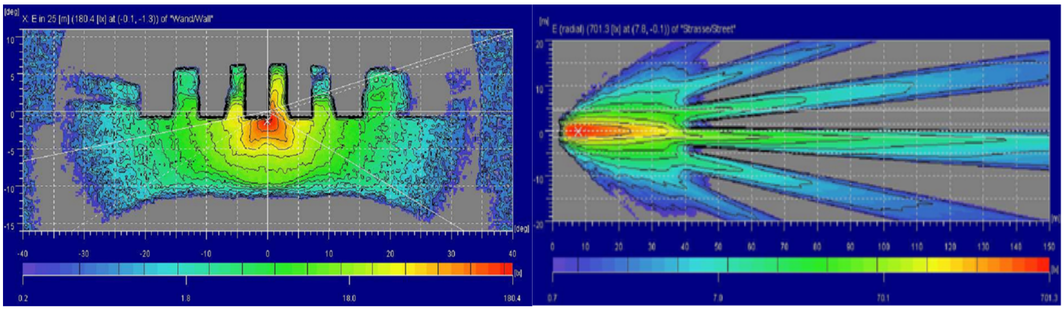

Figure 3.2.8: Illuminance of Segmented Shutter Headlight (Left : Front View, Right: Top View)

In addition to the high flexibility of this headlight, the shutters can move with a speed of up to 50 cm per second, which means switching from dipped to high beam takes about 20 ms, in contrast to HID[1] and halogen system that need about 150 ms. This system in combination with dynamic curve lighting is able to realize highly dynamic and accurate light distributions. The shutters are controlled by internal microcontroller connected to a CAN[2] bus. It receives commands for moving the shutters from any device connect on the communication bus. Such commands can be mark or shad an object. Based on these commands, the microcontroller generates the required movements of the shutters.

Figure 3.2.9 shows a car with oncoming traffic. Both cars have traditional dipped beam light distribution; the one on left is equipped with the segmented shutter headlights. In this case, one of the shutters is moved down to cut off the light in the direction of the oncoming car as shown in Fig.3.2.10 left. Now there are two alternative strategies for blinding out the vehicle. The first is to move down the neighbor shutter when the oncoming car comes closely. The other is to move

---

[1] High-Intensity Discharge (Xenon Headlamp)

[2] Controller Area Network

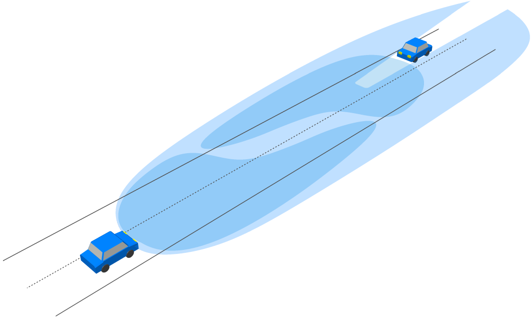

Figure 3.2.9: Light Distributions of Segmented Shutter Headlight

the curve lighting and then to shut down the neighbor shutter. The second option gives better results for distant cars, while the first one is preferable for close cars. Marking an object can be realized by sequentially moving the shutters to their end positions as shown in Fig. 3.2.10 right.

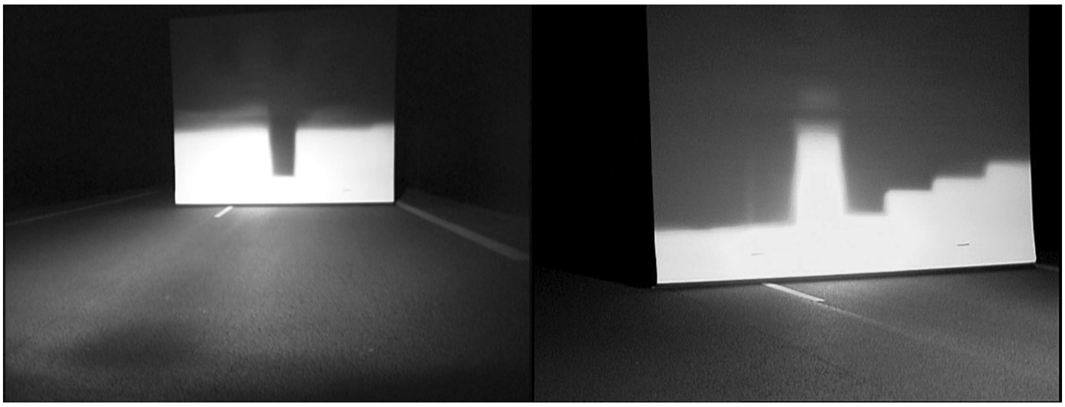

Figure 3.2.10: Projection of Several Light Distributions on 25 m Distant Wall

3.3 Solution Description

As stated in section 3.2.1, no sensor currently available in the market covers the above mentioned system requirements. For this reason, a network consisting of different types of sensors will be used to develop an object detection platform which can fulfill the system demands. The system is designed according to the flow of information as follows. Firstly, both internal and external sensors are needed to monitor the vehicle states and the environment in the vicinity of the vehicle. Then the acquired information must be processed. Among others, this involves disciplines such as object recognition to gain the necessary knowledge about the vehicle's surrounding as well as driving dynamics to take into account the vehicle's state. In the central information processing, all the data collected by the sensor systems is analyzed, sorted, and interpreted in order to gain knowledge about the environment in the vicinity of the vehicle and the vehicle states. Merging all the important data from these areas gives us the system's internal representation of the current situation. Then an appropriate strategy has to be chosen for the given situation. The strategy includes the information on the basis of which objects are blinded out or marked and how to do so. Finally, the chosen strategy is applied to the projected situation and the desired light distribution is calculated.

In this context, the implemented solution presents two operating modes. The first one represents a standalone concept, where the host vehicle has not the ability to exchange information with the other road participants. In this concept, a multi-sensor data fusion system will be designed to detect objects in front of the host vehicle. This multi-sensor data fusion system consists of a set of internal and external sensors from which information is incorporated within a single data fusion unit. Internal sensors give information about the host vehicle state, such as its velocity and steering angle. External sensors (e.g., lidar and image sensors) sense information outside the vehicle, such as oncoming/ongoing traffic, obstacles, and road information. All the sensors and the data fusion unit are connected via various physical and virtual buses. A system specification of communication messages will be built according to external sensors constraints.

In the second mode, which is called cooperative concept, vehicles are a part of cooperative communication network and each of them can exchange its own information with its surrounding as well as the other vehicles. The development of Wireless LAN using IEEE 802 standard has enabled the capabilities of the development of new services based on Car to Car (C2C) communication and Cars to road side Infrastructure (C2I) communication. In this system, the position of the vehicle will be determined via GPS and digital maps. Then this information in addition to the information about the vehicle state will be autonomously exchanged with the other road participants, which leads to the extension and improvement of the detection range of the vehicles.

It is worth mentioning that for the time being, the cooperative mode cannot be a replacement of the standalone mode but it is actually an extension to it, since building secure vehicle networks and preparing the required infrastructure to support DAS still requires further research. The aim of investigating the second mode is to measure the benefits and the performance gain if these techniques come in the market.

3.4 Conceptual Design of LBDAS in Mechatronics Discipline

Mechatronics is an interdisciplinary area of specialization, which involves the simultaneous application of mechanical and electrical engineering principles together with computer software in designing the so-called smart devices. In mechatronics systems design, the functional units have communication and decision-making capabilities, which means endowing them artificial intelligence. Sensors and actuators are two major components of every mechatronics system [Bis02]. The mechatronics concept establishes basic principles for a contemporary engineering design methodology. In this methodology, engineering products and processes have components that require manipulation and control of dynamic constructions to the required high degree of accuracy. In addition, the design process requires integrating enabling technologies such as information technology and control engineering.

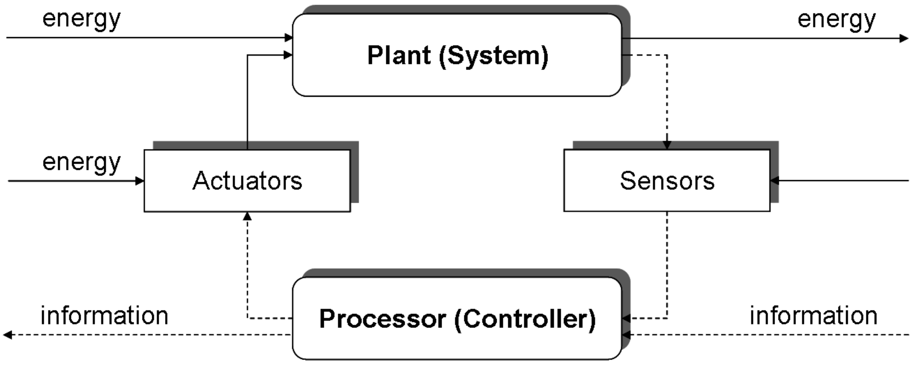

Figure 3.4.1: Layout of a Mechatronics System

As shown in Fig. 3.4.1, a typical mechatronics system consists of a plant, a sensing unit, a processor, and an actuating unit. A sensing unit can be as simple as a single sensor or can consist of additional components such as filters, amplifiers, modulators, and other signal conditioners. The controller receives the information from the sensing unit, makes decisions based on a control algorithm, and outputs commands to the actuating unit. The actuating unit, which in its turn influences the system behavior by generating the required energy to manipulate the status of the system.

The mechatronics discipline makes it possible to design reliable and versatile systems, which have encouraged the engineers to use it in designing and developing of automotive equipments such as the anti-lock braking systems. LBDAS is another example of the mechatronics systems in the automotive field as shown in Fig. 3.4.2. LBDAS is based on environment perception unit representing the sensor, an illumination device (headlight) which corresponds to the actuator, and a light control unit which represent the processor of the mechatronics system.

In [RS05, WWBH07], the authors have introduced guidelines to design intelligent systems in the framework of the mechatronics, which have inspired a new methodology to enhance the performance of the LBDAS proposed in section 2.1. For this reason, the following factors have been considered during the system design:

1. Since the system is aimed at operating in a dynamic environment, it should be able to rapidly change its behavior in response to the anticipation and changes in the environment.
2. The system should have the ability to decide when and how to change the system behavior without waiting for external instructions. For example, in case of any hardware failure (such as no image from the camera in glare free high beam mode) or non-plausible sensors data (i.e., when ego-speed is 400 km/h) the system should switch to the dipped beam and the driver is informed about not ready to assist state.

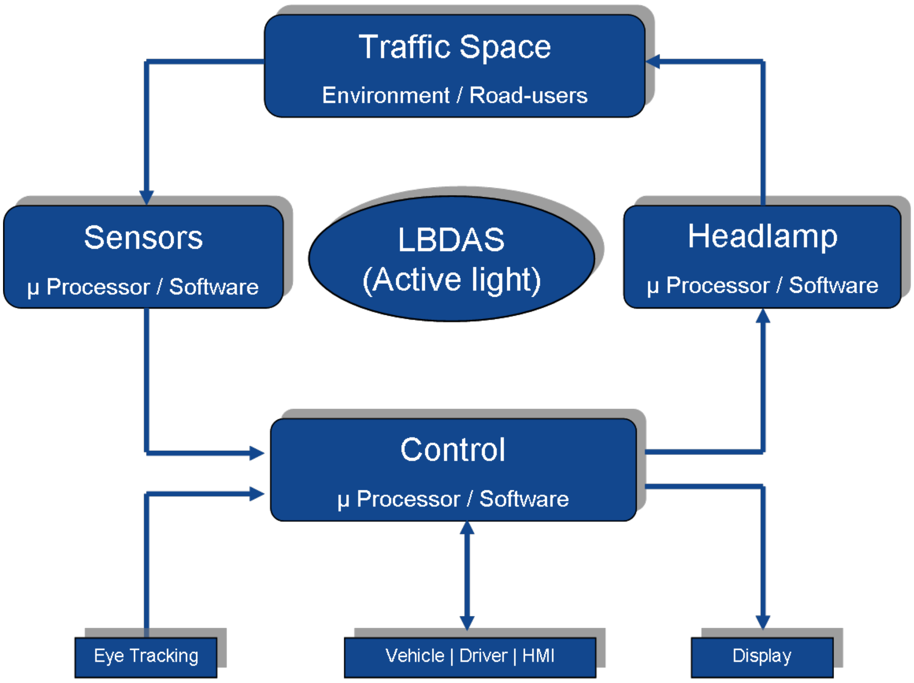

Figure 3.4.2: Layout of Basic Design of LBDAS

3. It is recommended to design the system as a network of autonomous units, which guarantee a high flexibility to the system while exchanging common information and adding or removing a node; for example, investigating various hardware and software configuration. The Metcalf's Law[SV99] confirms this concept by stating that the value of a network is equal to a square of the number of its nodes. The implication is that the increase in utility of a network, as it grows, is polynomial whilst the increase in expenditure for building extra nodes is linear.

4. Since most decisions affect more than one node in a network, it is necessary to involve all affected units in the decision-making process. The negotiation and voting could be the mechanism to support this involvement. In addition, when the decision-making is distributed to network nodes, which are close to sensors and actuators, the system is capable of reacting far more swiftly to unexpected events than centralized system with long reporting/instruction paths between information sources and executive mechanisms.

3.5 System Hardware Components

This section gives a brief introduction to the Hardware which will be used throughout the project, including types of measured data and the communication interfaces.

3.5.1 Hella Multibeam Infrared Distance Management System Prototype (IDIS)

The ACC system prototype from Hella is based on the LIDAR technology and extends the functionality of conventional cruise control systems. The IDIS infrared ranging sensor which Hella

Figure 3.5.1: Hella ACC IDIS Sensor, Photo: Hella

currently uses in its ACC is based on state-of-the-art optoelectronic measurement technology, while IDIS uses lidar technology to measure the time that light requires for a specific distance.

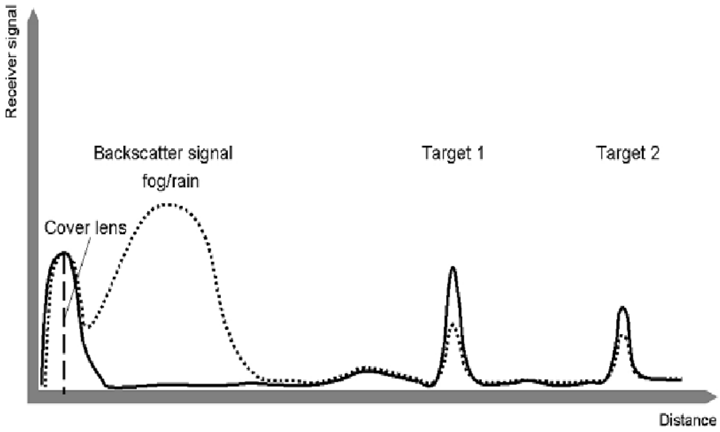

Figure 3.5.2: Working Principle of Lidar [Boe05]

A reduced visibility of the sensor under bad weather conditions (e.g., rain, fog, spray, etc.) is recognized by the backscatter signal of the sensor and target observation from the tracking as shown in Fig. 3.5.2. Based on this information, the sensor is able to warn the driver in case of reduced visibility, or even to shut down the functionality.

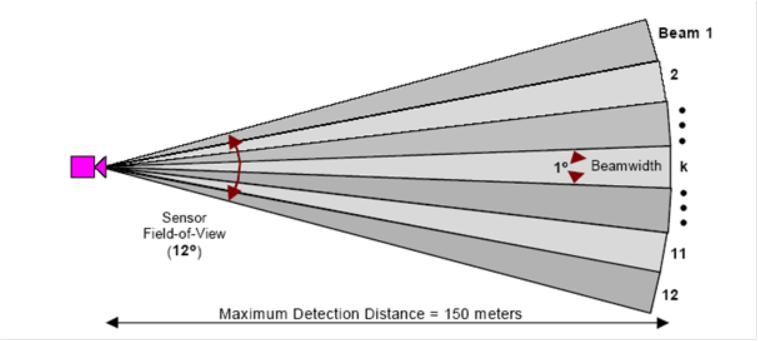

Figure 3.5.3: IDIS Detection Envelope

For each laser channel, up to three objects can be identified (Multi-Object-Detection). The data of the relevant target and the data of up to 25 internal objects are tracked by the sensor and taken into account by the ACC controller according to their distance, lateral position, and relative velocity. Lateral object separation is based on the lateral resolution of the sensor (1° different objects in neighbor). Channels can be separated, if they have a speed difference greater than 1.5 m/s.

3.5.2 Specification

1. Characteristics:
   a) Distance resolution: 0.1 m.
   b) Update rate: 60 ms.
2. Detection range:
   a) Opening angle (azimuth): ±6º.
   b) Maximum sensor range (undistorted visibility): 200 m.
   c) Distance differential resolution: 0.1 m.
   d) Absolute accuracy (3 m -150 m): ± (1% +1 m).
   e) Lateral deviation resolution: 1º.
3. Exta measured variables:
   a) Relative velocity:
      i. Range = -180 km/h ... +180 km/h.
      ii. Resolution = 0.1 km/h.
      iii. Accuracy = ± 1.8 km/h (tracking time = 1 s).
   b) Relative acceleration:
      i. Range = -10 m/s^2 ... +6 m/s^2.
      ii. Resolution = ± 0.01 m/s^2.
      iii. Accuracy= ± 0.2 m/s^2 (tracking time = 1.5 s).
4. Deployment: signal processing and ACC algorithms integrated in the sensor.
5. Communication interface: CAN Bus.
6. Multiple targets detection even in each individual channel.
7. Able to detect movable and stationary objects.
8. Able to detect non illuminated objects (not viewable).
9. The ACC algorithm can be used and adapted to suit marker light.

3.5.3 Camera

The decision whether to use a CMOS or CCD is of great importance to the overall design as there are significantly different hardware requirements, output data types, and other considerations described before. The final decision was to use a CCD because of its high light sensitivity, which is very important to detect weak and distant light sources.

Figure 3.5.4: Vision Sensor

An industrial FireWire (IEEE 1394 interface) camera based on “Sony Progressive Scan CCD sensor ” has been used as the vision sensor (mounted inside the vehicle behind the windshield along the central line) to take the images of the environment in front of the vehicle, including the road, vehicles on the road, traffic signs on the roadside, and sometimes other objects on the road. The used camera has the following specifications:

- Resolution: 640x480 pixels.
- Frame rate 30 fps[3] in full resolution.
- Platform: independent.
- Optics: Pentax high dynamic optics with anti-blooming filter.
- Color system: RGB 24 bit.
- Opening angle: ± 16° (estimated).

3.5.3.1 Detection Range Of Light Sources

The Camera angle and resolution makes detecting far away tail lights a great challenge [LHB+08]. For instance, a tail light with a size of 10×10 cm at a distance of more than 150 m is represented by less than one pixel in the image, which is not sufficient at all to initialize a light source hypothesis. Fortunately, the emitted light beam forms a cone so that a tail light at 700 m still hits areas of about 4 to 8 pixels and for head lights at distance 1000 m hits area of 6 to 10 pixels.

3.5.4 GPS Receiver

Two modules of FV-M11 GPS-receivers from San Jose Navigation were used to obtain the geographical position. The updating rate can be set in between 1 to 5 Hz by changing the GPS configuration through standard NEMA[4] commands. Changing the updating rate would be followed by changing the reading baud rate in the values specified from the specification table **??**.

---

[3] Frame Per Second

[4] NEMA: National Electrical Manufacturers Association

The interface to the GPS receiver is a serial port. GPS unit provides mainly the position of the vehicle in terms of longitude and latitude. Moreover, some other data are also available and can be extracted from GPS signal

- SoG (Speed over ground).
- CoG (course over ground) it represents the angle measured from the north pole.
- Altitude (height over ground).

3.5.5 Industrial WLAN Module

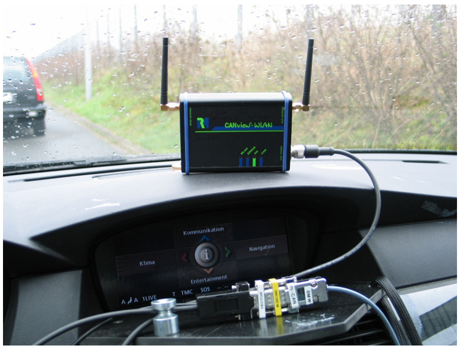

Figure 3.5.5: CAN View WLAN

Developing vehicular networks (also known as VANETs) for intelligent transportation is challenging due to the need of maintaining a reliable communication platform between high-speedi vehicles [Lue04, RSK08]. Since 2004, Cars Manufactures tried to organize their work in order to develop new standards of VANETs Communications. In Europe, car-to-car communication consortium defined a new IEEE 802.11p standard, targeting mobile applications. The standard bandwidth for VANETs communication was set to be in between 5.885 to 5.905 GHz. Cars OEMs[5] tend to use IPV6 as a basic network protocol to provide higher flexibility in nodes addressing compared to IPV4. IPV6[6] has been developed to meet the basic constrains of large data transfer between high speed vehicles [JTM$^+$06]. Although IPV6 is not a real time network protocol, it has been adapted by so many car-to-car modules manufactures. Additionally, tendencies were made to use mobile IPV6, which provides flexible and arbitrary access changes in IPV6 networks.

The WLAN module used in this project is a “CAN View WLAN Module”. This module allows wireless data transfer from one CAN-Network to another remotely located CAN-Network (which is in this case the other vehicle). The module aims to transfer absolute geographical positions, obtained from the GPS receiver, to provide accurate local relative positioning. The concept behind adapting such a new intercommunication concept is to enable the vehicles to communicate with each other to ensure safety and comfort for the drivers.

3.5.6 Addressable Headlight

The segmented shutter, which was introduced in 3.2.2.3 has been used.

---

[5] OEM: Original Equipment Manufacturer

[6] IPV6 (Internet Protocol version 6) is a network protocol to handle exchange of data packets and flexibility of addressing (2^{128} addresses)

3.5.7 On-Board Sensors

The test vehicle is equipped with internal sensors, which monitor the vehicle status. The main sensors are the speed, steering angle, and yaw-rate sensors.

3.5.8 Ego-Vehicle Gateway

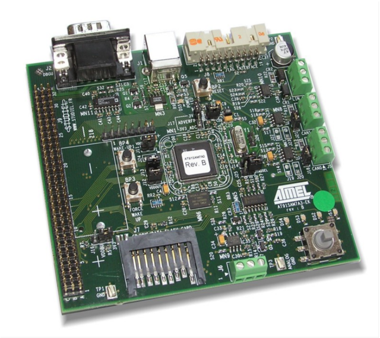

Figure 3.5.6: Communication Gateway

The evaluation board based on ATML AT91SAM7A3 Microcontroller was used to realize the communication gateway between the IDIS sensor and vehicle CAN bus. The main function of that gateway is to convert the vehicle dynamics data to IDIS specific format. In addition, it checks the plausibility of the on-board sensors data.

3.5.9 Processing Unit

An Industrial computer with two mainboards from Intel® equipped with Pentium® 4 (2.8 GHz) processor and 2 Giga byte of RAM have been constructed. Also, additional I/O ports for FireWire and CAN interface have been added. The communication between the two processors was achieved via a Giga bit Ethernet bus.

3.6 System Design

System design describes the system from two aspects. The first aspect identifies the configuration of the system hardware and how they are connected together, which is called hardware architecture. The second aspect illustrates the software configuration its functionality, and dependability, which is called software architecture.

3.6.1 Hardware Architecture

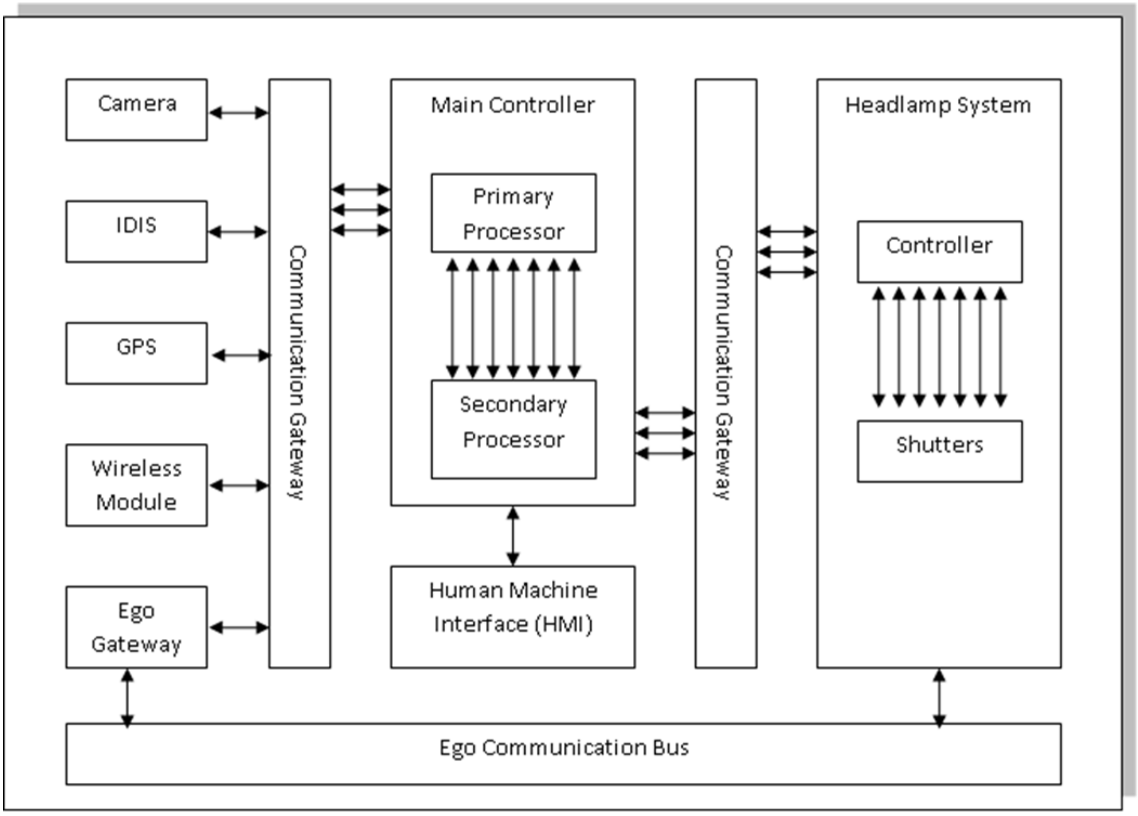

Figure 3.6.1: Hardware Architecture

As shown in Fig. 3.6.1, the external sensors are connected to the main controller unit via a communication gateway consisting of various communication buses (such as CAN, Firewire, Serial, USB, ect.). The processing software modules are distributed on the two processors. The sensor specific information processing modules are running on the primary processor, however, the post processing, decision making, and control signal generation modules are running on the secondary processor. A human machine interface is used to inform the driver about the system internal status and to receive driver's commands as well. The headlamp controller receives the control strategy setting from the secondary processor via a private CAN bus.

3.6.2 Software Architecture

Figure 3.6.2 describes the main scheme of the software architecture. The proposed architecture has the advantage of modularity. Each module is standalone and communicates with the other

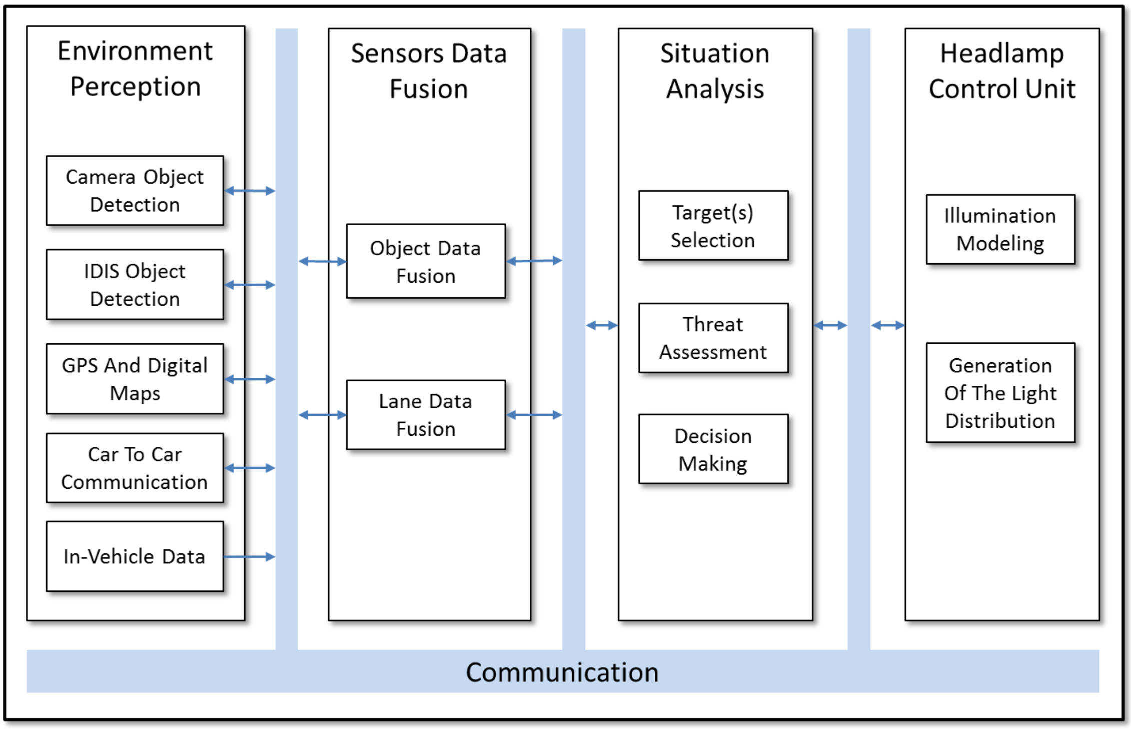

Figure 3.6.2: Software Architecture

system components via a pre-defined communication interface. This property enables to investigate different hardware configurations without changing the topology of the architecture. The environment perception function supplies the system with information about the objects on the vehicle space and about the road infrastructure. Data fusion is considered as the main management unit in the system that is responsible for the data synchronization and interconnection between the different software modules. The output of the data fusion is a list of objects with a mixture of features from the different sensors and information about the driving path. In the situation analysis, the relations between the objects and the infrastructure are estimated. By analyzing the relations, system decisions has been taken. Headlamp control module is at the end of the data flow, which is responsible to interpret the issued decisions into physical light distribution.

Since the decision making is based on the object type as well as its status, in the proposed architecture, object classification is not a local function of a specific module but its a system global concern. Firstly, each sensor's specific-module classifies the object based on its local information, then in the data fusion, the information of the different sensors are compared and completed in order to extend the features of the object. Based on the new gained feature the object class is confirmed. Thereafter, in the Situation analysis, the global relationships between the objects and the environment are used to increase the classification confidence of the system.

In the next sections, the software architecture and the implementation of its sub-functions will be illustrated in details.

3.7 Data Acquisition Subsystem

The data acquisition is a simple form of the hardware architecture described in section 3.6.1 without any information processing functionality. The software architecture consists of one main data capturing unit connected directly to the data sources. The system is used to collect data from the different sensors in the vehicle and save it in a form that it can be replayed afterwords offline for developing and testing the algorithms. The major challenge of any data acquisition system is to capture synchronized data in real-time and be able to play it synchronized again. In our system the camera image is used as medium to save all the system information. This is due to the reason that, the camera data is too much to be integrated in any other sensor information and image can describe the situation quickly which is helpful in filtering the acquired data for the labeling process. In addition, encoding all data in one file is reliable and portable.

The last five rows of the image, which are approximately 9 kb, are reserved for the external sensors data. In view of the fact that each pixel is represented in three bytes (red, green, blue), the first two bytes are utilized to save the data, however the last byte is used to save the calculated gray value of this pixel, which may be useful for lane detection, therefore reconstruction of the image is performed in the lane detection sub module before extracting the lane segments.

As previously highlighted, the hardware works with different update rates, thus the first byte in each row is used as an indicator of the status of the relevant sensor data, if it is updated in this cycle or not. Reading the synchronized data again from the file is then a trivial task, since it corresponds to a unique image frame number.

In the next chapter the environment perception module and its corresponding methods and algorithms will be illustrated.

Chapter 4

Environment Perception

4.1 Camera Object Recognition

Since many decades, camera or vision-based systems are under investigation for applications in driver assistance systems. So far, most of the studies use the camera to detect objects and obstacles in the vehicle's environment in short range, up to 50 m, under the day light conditions. As already established in many image processing publications, detecting objects at night-time using image processing is a challenging task because most of the current algorithms depend on extracting the contour of the object and comparing its features with a predefined criteria and thresholds in order to classify it. But at night-time, the camera cannot get the contours of far objects . Even with infrared cameras, objects cannot be detected at a distance more than 150 m, which is also not suitable for DAS such as High-Beam assistant system. However, the camera is still the most reliable candidate sensor to detect vehicles at night. The physical nature of the camera sensor enables it to detect the bright object on a dark background at a long distances. This property is used to detect the vehicles from their light beams. The dilemma is that all the detected bright objects on the road should not always be a light source of a vehicle. It could be a traffic light, or a reflection of traffic sign/road reflector, or a streetlamp. Therefore, many methodologies and algorithms were introduced to extract the right bright blobs which are corresponding to the vehicles on the road. In [CH03], the lane boundaries were modeled as a curve in the form of a second-order polynomial by using image objects corresponding to reflectors on the ground plane. Then all the bright spots between the road boundaries are considered as vehicles. However, the reflectors sometimes are broken in some sections of a highway and are occluded by the front vehicles in the image. So, it is difficult to recover the lane boundaries from the limited number of reflectors in the images. Also in [CH03] the authors assumed the rear-light to be in the form of a white bright region surrounded by red pixels and have used a fixed threshold to segment the image. Although, color signature of tail lamps is an important feature for detecting leading vehicles, however the red and white pixels will be falsely detected under different illumination with a fixed threshold. In addition, the shapes of the rear-light are different in braking and moving situations. In [TDH94], the authors assume that the headlights are the bright blobs relative to the dark background. Next, they perform binarization to extract the headlights of the oncoming vehicles on the highway at night. However, the drawback of this approach is that the detected headlights are easily affected by the other bright blobs. In [WHF05], the lane boundaries were extracted by detecting the painted lane marks assuming that the lane marks are fairly visible as daytime situations. However, at night, the painted lane marks are not obvious in the image enough to generate sufficient segments to reproduce the lane boundaries. Besides, studies in [WHF05] have not taken into account the detection of the oncoming traffic. In our approach, a novel method has been implemented. It is based on multi-band adaptive thresholding of the input color image to extract the relevant blobs. Then for each blob a set of spatial features are extracted from the image contents as well as the global relations among the blobs. After that, the feature vector is classified and the blob is tracked as will be shown in the next section.

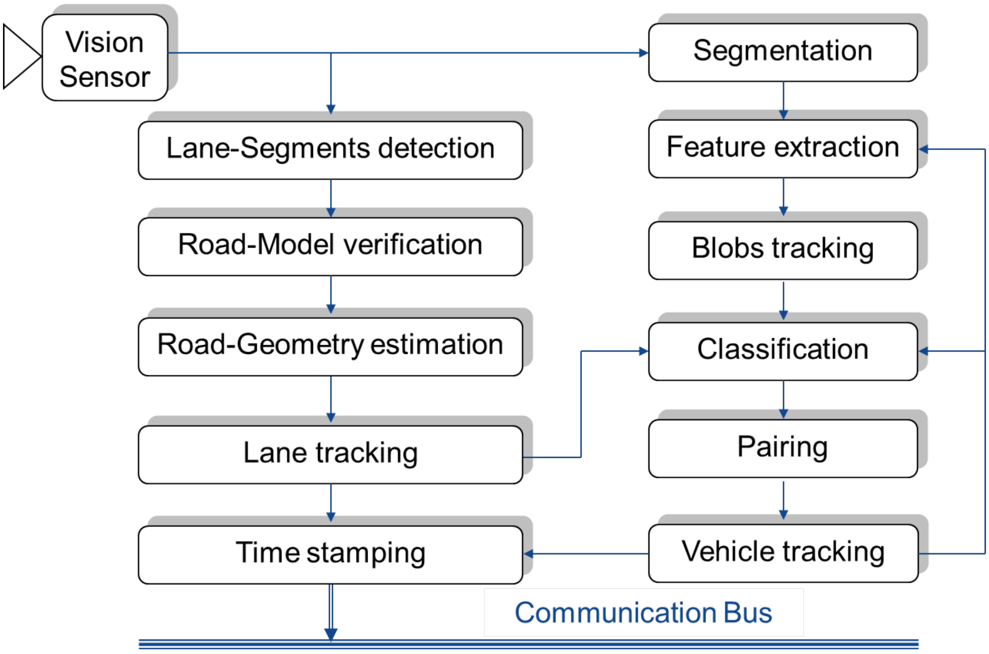

Figure 4.1.1: Functional Flow of Camera Object Recognition Submodule

4.1.1 Road Lane Detection

Lane detection is an essential function in many intelligent vehicle applications, such as Lane Departure Warning, Intelligent Adaptive Cruise Control, and for future full autonomous vehicles [TsBM99]. The function provides information about the road construction (e.g., number of available lanes) as well as the position and orientation of the vehicle within its driving lane. The main feature used to detect the road lane is the white markers. Moreover, other contributions have explored the possibility of using the road infrastructure such as guiding reflectors and road guarding rails to estimate the road boundaries.

Detecting lane at night is a great challenge, since the lane markers can only be seen at a distance of a few meters in front of the host vehicle (about 30 m). Even using the second concept of using the guiding reflectors is affected dramatically with the light sources of the oncoming traffic. Therefore, the determination of the driving path in our system is not based only on the detected lane from the camera, but it also makes use of the data from IDIS as well as digital maps to estimate an accurate driving path for a long distance.

In our system, a simple lane detection module was implemented based on the detection of lane markers. It assumes that the road 30 m in front of the vehicle is either straight otherwise for the curvature, it can be approximated with connected straight segments. In addition, we also assume for each lane left and right markers are available.

4.1.1.1 Lane Segments Detection

Hough transformation is a well-known technique to detect straight lines in digital image processing [BdMRJ04, DH72, MFS, MFS01]. In the current system, the Hough transformation is applied on the lower part of the image, where it is expected to find the lane marker. The detected segments are sorted according to their inclination angle with respect to the vehicle in form of right lane or left lane. Since we are dealing with solid and dashed lane markers, multiple peaks in the Hough space have been considered as the relevant segments. As the lane marker is not a thin line, thus multiple segments candidates appear on the same marker, therefore the concurrent segments are fused together to remove the redundant segments.

4.1.1.2 Road-Model Verification and Road-Geometry Estimation

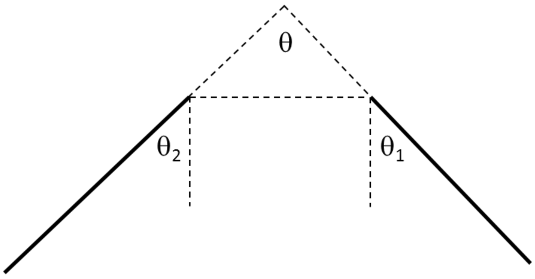

Figure 4.1.2: Lane Geometrical Model $\theta = \theta_1 + \theta_2 \leqq 90^{\underline{o}}$

In image space, from the vehicle point of view, the angle between left and right lane should be smaller or equal to 90 degree according to the geometrical projection principle of parallel lines on the image plan [FP02]. Thus, the algorithm starting to detect the ego lane by investigating the inner segments to find a pair fulfills the angle criterion. Then, for each lane side the remaining segments are checked if they can be an extension to the estimated lane segment. After detecting the ego lane successfully, the process is repeated again to detect available neighbor lanes. If the segments coupled to the ego lane are not detected, the segments, which have angle $<$ 90 degree, are used to initialize the lane model for the verification in the next cycles. After successful detection of lane markers, the camera parameters and the flat-world assumption are used to estimate the lane width in meters. The computed width is verified against the allowed lane width.

4.1.1.3 Lane Tracking

The result of the previous function is a list of segments describing the road lane geometry. Each segment is represented by a start and an end point. These points are used as a tracking feature. An (α, β) filter analogous to the one illustrated in section 4.1.4.1 is used again to find the lane in the next frame and to estimate the detection confidence.

4.1.2 Light Sources Hypotheses Generation

The objective of this step is to generate reasonable hypotheses, which can be candidates of possible light sources. This is often referred to as the process of clustering the relevant image pixels into a set of non-overlapping regions, which often have similar color values. The aim is to define the boundaries among the light sources candidates and the background, which is mainly dependent on the determination of the thresholds.

4.1.2.1 Isolating Relevant Pixels using Thresholding

Thresholding is a shape extraction technique, in which the images are processed in a way to separate the target objects from the background [NA02]. If for example, the shape to be extracted is defined by its brightness, then by thresholding the image at a particular brightness level this shape can be retrieved. Thresholding is clearly sensitive to the change in illumination. A change in image illumination will also change the perceived brightness of the target shape. Therefore, in such cases, unless the threshold level is adapted according to the change in brightness level, the thresholding technique can fail. The main advantage of the thresholding technique is that it does not require much computational resources.

Basic (or fixed) thresholding can be used in applications where the illumination can be carefully controlled. However, if the overall illumination level cannot be controlled, such as in LBDAS, then the threshold should be adaptively determined. Adaptive thresholding changes its level based on the actual illumination status. Image-Histogram-Analysis is one of the popular methods to estimate a global adaptive threshold. Assuming that the background pixels are the most dominant in the image, then the optimal threshold is the value which best separates background pixels from the rest. The major problem of global thresholding is that changes in illumination across the scene may cause some regions to be brighter and some regions darker. Thus, in the bright region non-relevant object's pixel can be extracted, while in darker region relevant pixels can be missed. In order to solve this problem, local thresholds have been introduced [NA02]. Instead of having a single global threshold, multiple thresholds can be estimated across the image. In our implementation, a row based local threshold technique has been used. Since we are searching for the bright objects in the image, the maximum pixel value in each row is theoretically the ideal threshold. Although, 80% of the maximum value has been considered as a recommended practical threshold. Then this value is checked against the minimum allowed threshold in order to decrease the influence of the ambient and sensor noise.

$$g_c(x,y) = \begin{cases} f_c(x,y) & \forall f_c(x,y) \geq \tau_c \\ 0 & \text{Otherwise} \end{cases} \tag{4.1.1}$$

where τ_c is the color thresholds. The subscript c denotes the color, i.e., $c \in \{R, G, B\}$.

If $g_c(x,y)$ is the thresholded variant of $f_c(x,y)$, the value of $g_c(x,y) = f_c(x,y)$ when $f_c(x,y) \geq \tau_c$ otherwise $g_c(x,y) = 0$. The thresholding procedure is repeated for each row in the image. As we are dealing with a color image, the whole process is repeated for each color channel.

Horizontal thresholding (i.e., row-based thresholding) has the disadvantage that some blobs which are horizontally mirrored, such as the reflection of the headlights on the road, cannot be separated from each other and results in a large blob. In addition, weak light sources like distant tail lights will not be detected when in the same row a bright headlight exists, which makes the local threshold of this row too high as shown in Fig. 4.1.3. Therefore, the thresholding process is carried out again but this time in column-based manner.

For each thresholding direction, the thresholded images of the three color channels are compared together and an intermediate image is built containing only the relevant target pixels. The same is carried out to create an intermediate vertical thresholded image, which is is used to compensate

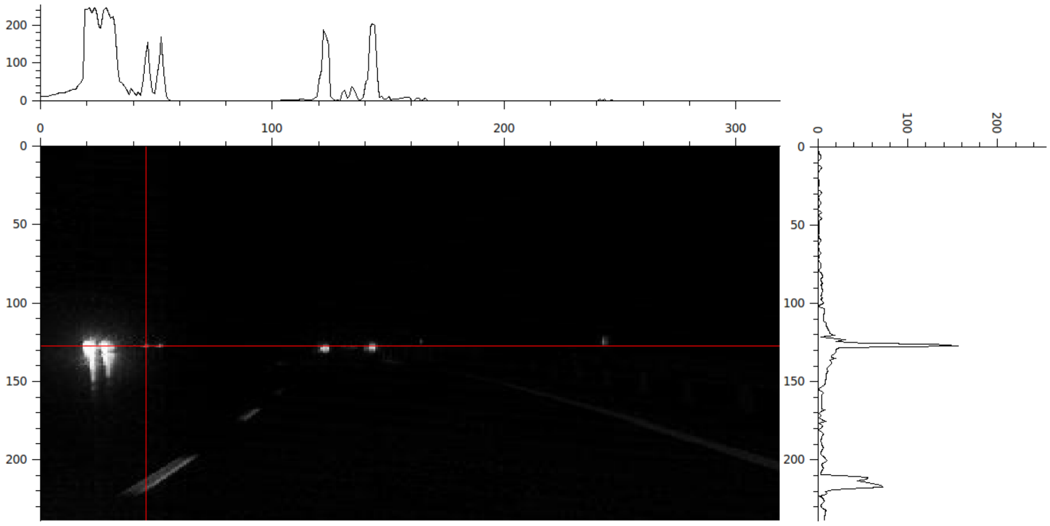

Figure 4.1.3: Thresholds Estimation

the missed relevant pixels from the horizontal thresholding and to detect if an overflow has occurred between the blob of the light source and its reflection. If an overflow is detected, the blobs are separated and a final thresholded image is created for the next processing steps.

4.1.2.2 Blobs Boundary Estimation

After isolating the relevant pixels, the resulted image is analyzed using connected components labeling technique [DST92] to generate the blobs (or the hypotheses) which represent the light sources. The goal of this function is to group pixels of the same brightness and color into regions corresponding to similarity membership criteria. If the pixel under investigation satisfies the membership criterion (considering its eight neighboring pixels), it will be merged into the region of the neighboring pixels and acquires the same label.

In order to be able to describe a blob, its boundary should be determined. Boundaries occur at points where the rate of change of the image brightness is a local maximum, which can be interpreted mathematically as the peaks of the first derivative or, also equivalent to the zero crossings of the second derivative of the Image. By looping over the region pixels searching for the local maximum, the region's contours can be extracted. Since we are searching for a solid blob (corresponding to the light sources), the inner and open contours are neglected. The result of this step is a vector of pixels, which surrounds the region candidates. Each region is considered as a light source hypothesis. Considering the color information during the region grouping as well as the boundary estimation allows to separate efficiently the blobs of the tail and headlamps even in partial occlusion situations.

To simplify the blob identification process, a so-called blob-bounding box is determined. The bounding box is an approximation to the shape of the blob in form of a rectangle. The rectangle should contain all the blob pixels, but sometimes the rectangle is larger than the blob itself because of the approximation process. Therefore, a similitude factor is calculated, which is the ratio of the blob area to the bounding box area. This factor is considered in the classification process. If the ratio is $< 2/3$, it indicates that the blob lays diagonal in the bounding box, which is a typical feature of the lane markers.

4.1.2.3 Blobs Filtration

In order to increase the system stability and to reduce the processing time, the blobs resulted from the previous process are filtered against the following criteria.

- Small blobs under 3 pixel are not considered as relevant blobs because of the great effect of the noise on these blobs and can lead to increase the false alarm rates.
- Large blobs possessing more than $^{2}/_{3}$ of the image width are neglected, since it consumes too much processing time, which may break down the system performance. Moreover, light source with this large size, should be detected from the IDIS sensor and can be retrieved afterwards via the data fusion unit. For more details refer to section 5.1.

4.1.3 Features Extraction

Features are distinct object properties which can be regarded as the signature of this object. If these features are combined together in a form of probability (classification), it should enable to identify that object. The features can be in the form of geometrical measurements like shape, size, object texture, or even object dynamic behavior. For each hypothesis resulted from the previous procedure, a feature vector is extracted as will be shown latter in this section.

4.1.3.1 Geometrical Features

This type of feature describes the physical geometrical constraints of the blob itself and its relationship to the other neighbor blobs in the image.

Relative Position to Image Center

In a typical driving situation without a large pitching activity, following hypotheses can be assumed. Firstly, small blobs which represent far away light sources should appear near the image center. Secondly, large blobs (near light sources) should appear at the right and left boundary of the image. In order to obtain plausible values, the host vehicle yaw angle should be considered in estimating the relative position.

Blob Area and Width

According to the pinhole camera projection model [FP02], the objects close to the camera possess a large area in the image given by $B_a \alpha \frac{1}{d}$ and vice versa. Practically, inverse blob area has delivered more reasonable values than using the area itself. The Blob Area can be simply calculated by counting its number of pixels. By knowing the scaling factors of the camera sensor, the blob area and width can be also estimated. In addition, combining these features with the relative position in the image, large blobs such as traffic signs can be recognized and filtered.

Width to Height Ratio

This feature estimates the squeezing of the blob. An ideal projection of a light source on the image should be a solid disk. However, due to the ambient and sensor noise, it appears almost as rectangle which tends to be a square. From that, if the width $\ll$height, it indicates that the blob can be a guiding reflector. On the other hand, if width $\gg$height, the blob can be a squeezed traffic sign.

Pairing Probability

The pairing feature provides clues about the presence of a twin blob. This is due to the fact that all four wheeled vehicles must have at least left and right head and tail lights, thus observing the corresponding two bright spots sharing similar characteristics in the image leads to increase the probability that both of blobs are light sources and belong to one vehicle. Of course, this feature is not applicable for two wheeled vehicles, therefore it can be considered as a secondary feature to confirm the classification results. Since neither the physical width of the observed target nor its width in image plan can be estimated, a look-up table was generated containing for each row in the image the expected minimum and maximum vehicle width in pixels. Then, for each blob, a Region Of Interest (ROI) with the maximum width is placed on the right and left of the blob. Each blob exists inside the ROI is investigated and a similarity factor (SF) is determined. If a twin blob is found, the distance between the two blobs is checked against the minimum allowed width.

Reflection Existence Probability

The reflection detection of the headlights on the road, especially in rainy weather is very helpful to filter and reduce the number of irrelevant blobs. Also, this feature increases the probability of the presence of the light source. To detect the reflection, a vertical correlation between the blobs is estimated. If the correlation is larger than a threshold, the lower blob is considered as a reflection and the higher blob is assigned as a light source with reflection. The threshold is depending on the position of the blob in the image and the maximum allowed roll angle of the host vehicle. From the observations, it was obvious that the reflection area is almost smaller than the blob itself and should be found in the lower part of the image (i.e., under the horizon). Therefore, these properties were used also to confirm the reflection existence.

4.1.3.2 Texture and Color Features

Difference of Color Histograms

The main idea of this feature is to find a finite number of color histograms, which can represent the various types of head and tail lights. Each histogram describes a category of light sources, which are thought to exhibit internal cohesion and/or external isolation.

For generating the master histograms, more than 500 samples were collected. They include many types of vehicles head/tail lights in various traffic situations, such as highways, secondary roads, and city streets with scenarios of dense traffic and also with few vehicles. For each light source, three 15 bins histograms were extracted (a histogram for each color channel). Practically, the important information is found in the high bins of each histogram, therefore only the last high 5 pins pro channel were considered. Afterwords, the histograms are normalized to make them size-invariant.

Computing the distance between a new entity and the 500 histograms - resulted from the above mentioned step - is very time consuming. Thus, the histograms were clustered into subsets using multi-dimensional k-mean cluster methodology [Mac67]. The k-means procedure can be viewed as an algorithm for partitioning n samples into k clusters so as to minimize the sum of the squared distances to the clusters center. The main advantage of the k-mean is the ability to determine the number of target clusters. By iteration, it was found that 4 clusters have optimally represented the headlights; however 9 clusters for tail lights were required. Then each cluster centroid has been saved as a master histogram. Thereafter, a modified histogram intersection algorithm was used to estimate the similarity factor between the target histogram (blob) and the clustered master histograms. The standard algorithm of histogram intersection is represented in [BTB05].

$$SF_i(Master, Target) = \frac{\sum_{j=1}^{n} \min(H_i^M[j], H^T[j])}{\max\left(\sum_{j=1}^{n} H_i^M[j], \sum_{j=1}^{n} H^T[j]\right)} \tag{4.1.2}$$

$$SF = \max_{i=1 \to k} \{SF_i(Master, Target)\} \tag{4.1.3}$$

where $SF(Master, Target)$ is the similarity factor (if the blob histogram is identical to one of the masters then $SF \to 1$ otherwise $SF \to 0$), n denotes the number of histogram's pins, j is a histogram bin, k represents the number of master histograms, and $H^M$and H^T are the master histogram and the target blob histogram, respectively.

Blob Brightness and Variance

A point light source is one of the forms of a point energy source that obeys the inverse square law given by the equation (4.1.4). It propagates in the space in the shape of a sphere or part of a sphere. Since the area of the sphere is related to the square of the radius of the sphere, which is the distance from the point source, the illuminance decreases by the square of the distance from the point source.

$$I = \frac{S}{4\pi r^2} \tag{4.1.4}$$

$$B = \frac{\sum_{u=x_1}^{u=x_2}\sum_{v=v_1}^{v=y_2} G_{u,v}}{|x_2 - x_1|\,|y_2 - y_1|} \tag{4.1.5}$$

where S is the luminosity, I represents the illuminance at distance r from the point source, B denotes the brightness, $G_{u,v}$ is the gray value of the pixel, and $\{(x_1, y_1), (x_2, y_2)\}$ represents the diagonal corners coordinates of the blob.

The average of the gray values is used as indicator of the intensity of the light source acquired by the camera. Since the headlights not always have the same strength and the dynamic range of cameras is limited, it is not possible to use the light intensity directly to calculate the distance between the host vehicle and the expectable targets. However, this feature can be used to indicate roughly how far the light source can be. It might be worth mentioning that, the used camera has delivered reasonable values for light sources which are more than 50 m away. All light sources which are closer than this distance lay in the camera saturation level.

Based on the computed mean gray value, the variance (σ^2) among the blob pixels was calculated. The variance is used as a measure of spread of the pixels from each other.

Estimating the mean gray value for all blobs is a heavy time consuming process, thus an intermediate representation, known as the Integral Image algorithm [PM01], was used to provide a fast alternative. The integral image, denoted by $ii(x, y)$, at location (x, y) contains the sum of the pixel values above and to the left of (x, y).

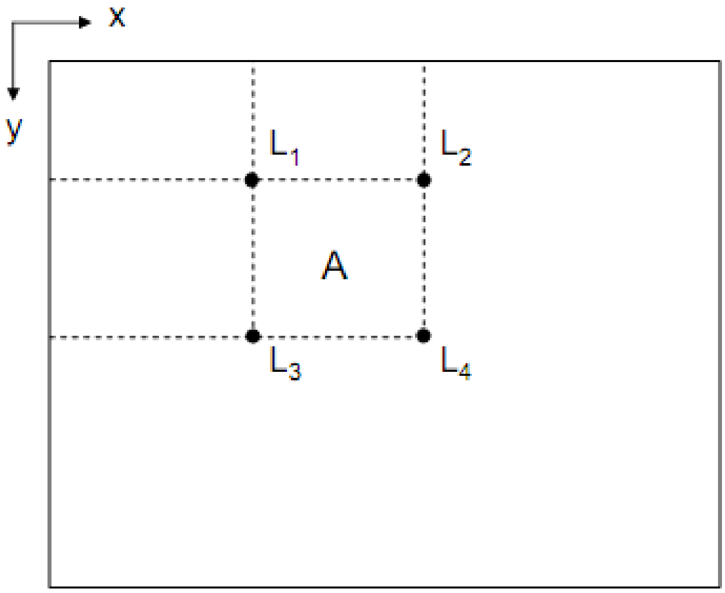

Figure 4.1.4: The Sum of Gray Values in Region A is Calculated as Follows:$G_A = L_4 + L_1 - [L_3 + L_2]$

$$ii(x, y) = \sum_{x' \geq 0}^{x} \sum_{y' \geq 0}^{y} i(x', y') \tag{4.1.6}$$

The intermediate image can be computed in one-pass over the original image using the following recurrence relation:

$$s(x, y) = s(x, y - 1) + i(x, y) \tag{4.1.7}$$

$$ii(x, y) = ii(x\text{-}1, y) + s(x, y) \tag{4.1.8}$$

where $s(x, y)$ denotes the cumulative row sum and $s(x, -1) = ii(-1, y) \equiv 0$. Given the integral image, the sum of pixel values within a rectangular region of the image aligned with the coordinate axes can be computed in only one step with four mathematical operations. For example, to compute the sum of region A in Fig. 4.1.4, the values of the four corners have to be substituted in equation (4.1.9).

$$G_A = L_4(x_4, y_4) + L_1(x_1, y_1) - [L_3(x_3, y_3) + L_2(x_2, y_2)] \tag{4.1.9}$$

The produced integral image will also be used latter in the situation analysis and threat assessment section 6 to estimate the perceptibility probability.

Blob Homogeneity

As mentioned previously, the headlight can be considered as a point energy source. Consequently, the center of the blob is always brighter than the area near its boundaries. However, traffic signs appear brighter on one side than the other due to illumination direction or the mounting position on the road side. The homogeneity of the blob is calculated as a vector of normalized differences between the centroid of the blob and the pixels on the boundaries for each side as given by:

$$H^n = \frac{\sum_{u=X_1^n}^{X_2^n} \sum_{v=Y_1^n}^{Y_2^n} G_{centroid} - G_{u,v}}{|X_2^n - X_1^n|\,|Y_2^n - Y_1^n|} \tag{4.1.10}$$

where H is the homogeneity, n denotes the number of region of interests (e.g., left, right, top, and bottom), (X_1, Y_1) and (X_2, Y_2) are the coordinates of the diagonal corners of the region of interest, and $G_{centroid}$ is the gray value of the blob-center-pixel.

For robustness and to reduce the effect of the applicable noise, 5% of the pixels around the center were used to calculate the average value of the centroid pixel. Nevertheless, 10% of the boundary pixels were used to calculate the homogeneity factor for each side.

Color Ratios

Color ratios in blob area are simple features, which are used to give a quick information about the classification tendency of the blob. The color ratios have been computed as follows:

$$Ratio = \frac{Number\,of\,colored\,pixels}{Blob\,area} \tag{4.1.11}$$

- Red ratio: Red pixels indicate the probability of the presence of a tail light. A pixel is considered red if the red component is dominant i.e., $\frac{R}{R+G+B} > \tau_r$.
- White ratio: White pixels represent headlights or guiding reflectors as well as white traffic signs. A pixel is white when the color components $R + G + B > \tau_{w1}$ and $R, G, B > \tau_{w2}$.
- Black ratio: In other hand black pixels give information about any texture in the blob. This feature has been used to filter white traffic signs, since the information on the white signs are almost written in black color. A pixel is black if the color components $R + G + B < \tau_b$ and $R, G, B > \tau_{b2}$.

The thresholds τ_r, τ_{w1}, τ_{w2},τ_{b1},τ_{b2} can be experimentally determined.

4.1.3.3 Dynamical Features

This type of features depends on finding correspondences between two or more sequential frames to indicate the moving direction of the objects in the image. Thus it is useful to estimate this feature after the tracking, because no extra matching is required anymore. Therefore, this feature has been used to confirm the classification of moving objects and to increase its classification probability.

Lite Optical Flow

Optical flow [FW06] is a well-known image processing technique to detect moving objects in video stream. The main disadvantage of optical flow is that it needs a huge processing capacity. In this work, a simple optical flow was implemented. Instead of finding a correspondence for each pixel in the blob, only the change in the blob centroid position between two frames was computed in order to determine roughly the moving direction. This feature is aimed at detecting overtaking vehicles with significant high relative velocity, which are found on the side boundary of the image.

4.1.4 Blob Tracking

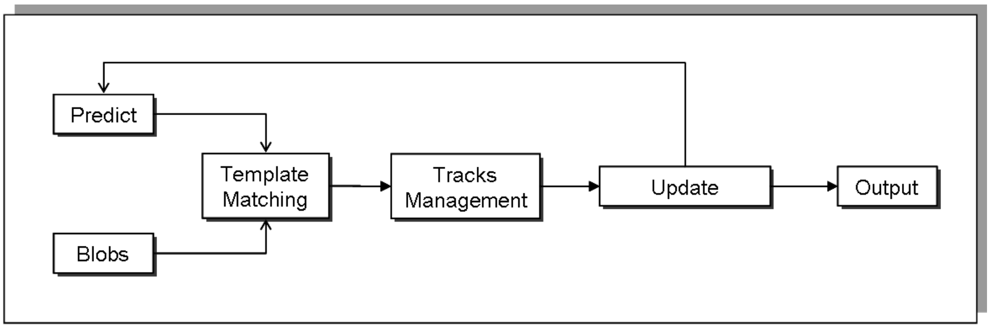

Figure 4.1.5: Tracking Work Flow

This function is dedicated to track the blobs independent of their classification. This gives the opportunity to build a history for the blob, which will be used in the classification module to confirm the classification results. Since in this stage we are still in the image space and dealing with colored objects, which have a specific color signature, a color template matching tracker was used to track the objects. Template matching is a popular technique in computer vision, which has been widely used in visual object recognition and tracking. The main drawbacks of the traditional template matching methods are that they consume usually too much processing time. This is due to the fact that, the template needs to be matched to every location in the image to find a correspondence region[TT07], which makes it not suitable for real time applications. In this thesis, a novel approach to accelerate the template matching is used. The main computation saving is achieved by using an (α, β, γ) filter to predict the blob position in the next cycle, which limits effectively the searching window and improves significantly the processing efficiency. Moreover, predicting the position increases the tracking robustness by avoiding matching the blob to a wrong correspondence due to change in the blob appearance features or when there are many other similar-looking blobs in the image.

4.1.4.1 The (α, β, γ) Filter

An (α, β, γ) filter is a simplified form of an observer for estimating and smoothing the states of the system. It is closely related to Kalman filter and to linear state observers used in the control theory; however it does not require a detailed system model. The procedure is performed in two steps as follows.

Predict: Constant cycle interval is a major parameter for predicting accurately the future states of a dynamic system. However, in image processing, the cycle interval varies depending on the complexity of the situation. Therefore, a new methodology has been used to overcome this problem. Instead of predicting the future states, the current states are predicted from the previous ones as shown in Fig. 4.1.5. This means the cycle time will be estimated in each cycle and used to predict the states in the current time. The direct measurements, which can be determined in each cycle, are the x and y coordinates of the blob. While, velocity and acceleration components v_x, v_y, a_x, and a_y can be observed. Then the state of the system can be represented as the vector $[x, y, v_x, v_y, a_x, a_y]^T$. The prediction of states can be calculated as follows:

$$\hat{x_p}(k+1) = \hat{x}(k) + \triangle T \cdot \hat{v}(k) + \frac{\triangle T^2}{2}\hat{a}(k) \tag{4.1.12}$$

$$\hat{v_p}(k+1) = \hat{v}(k) + \triangle T \cdot \hat{a}(k) \tag{4.1.13}$$

Update: In this step, the predicted values are used to estimate a filtered status for the system given by:

$$\hat{x}(k) = \hat{x_p}(k) + \alpha \cdot [\hat{x_m}(k) - \hat{x_p}(k)] \tag{4.1.14}$$

$$\hat{v}(k) = \hat{v_p}(k) + \frac{\beta}{\triangle T}[\hat{x_m}(k) - \hat{x_p}(k)] \tag{4.1.15}$$

$$\hat{a}(k) = \hat{a_m}(k) + \frac{\gamma}{\triangle T^2}[\hat{x_m}(k) - \hat{x_p}(k)] \tag{4.1.16}$$

where $\hat{x}$, $\hat{v}$, and $\hat{a}$ are smoothed x and y components of position, velocity, and acceleration, respectively. The parameters $\hat{x_m}$, $\hat{v_m}$, and $\hat{a_m}$ represent the measurements, while $\hat{x_p}$, $\hat{v_p}$, and $\hat{a_p}$ are the predicted states from the last cycle and $\triangle T$ is the cycle time.

The values of $\{\alpha, \beta, \gamma\}$ are estimated experimentally depending on the system dynamics and constraints, where $0 \leq \{\alpha, \beta, \gamma\} \leq 1$. In general, large values tend to produce faster response for tracking transient changes, while smaller values reduce the noise in the state estimates.

4.1.4.2 Template Matching

For each blob, a template in form of attributes vector is defined. Firstly, the bounding box of the blob is divided horizontally and vertically into four parts and a 30 pin color histogram for each part is extracted. After normalizing the resulting histograms, the pins values are added sequentially in the template vector. In addition, the blob area, width, and height are inserted.

The target of the template matching procedure is to minimize the following objective function

$$O = w_1 \frac{\sum_{i=1}^{n} \min(F^C[i], F^T[i])}{\max(\sum_{i=1}^{n} F^C[i], \sum_{i=1}^{n} F^T[i])} + w_2 \frac{|A^T - A^C|}{A^T} + w_3 \frac{|W^T - W^C|}{W^T} + w_4 \frac{|H^T - H^C|}{H^T} \tag{4.1.17}$$

where $F^T$and F^C are the color attributes of tracked blob and correspondence candidates, respectively. Moreover, n is the number of template vector color attributes while, A, W and H are the Area, Width, and Height, respectively. Weights w_1, w_2, w_3, and w_4 are used to adjust the importance of the computed errors.

The dynamics of the host vehicle are modeled as noise added to the predicted position in form of standard deviation σ. The σ value is adapted to the position of the blob in the image. When a correspondence blob is found, the template of the new measurement is copied to the track template in order to get an adapted template according to the updates in the blob status.

4.1.4.3 Tracks Management

This unit is responsible to mange the tracks attributes, such as assigning the identification number (ID), filtering the track list from dead hypotheses, and to check the tracking plausibilisation.

4.1.5 Classification

The major target of this function is to identify the generated hypotheses and to group them systematically in categories. Since we are dealing with an open environment, there is not an applicable technique that can be used to classify all the different types of objects in the vehicle's space. Therefore, the proposed classification function deals with the identification of three categories of objects, namely tail lamps, headlamps, and reflectors, which are the most probable in the traffic situations and may be considered as the relevant targets to our system. All other objects are classified as unknown.

4.1.5.1 Classifier Selection

Selecting a classifier is depending mainly on the objects to be classified and the ability to mathematically model the relationships between the object features to distinguish it from its surroundings. It is not always trivial (sometimes impossible) to derive the model governing the features relationship, especially while dealing with different types of objects in a complex environment.

Artificial Neural Networks (ANN) is one of the methodologies that have been used effectively to model such complicated systems. The ANN estimates the relationships and the contribution of each feature while recognizing the object via training samples. It tries to weight the input features in a manner that gives the required output. In our classification problem, a hardware ANN classifier implemented on FPGA chip called “CogniMem” from general vision company[1] has been successfully integrated in the recognition system. The CogniMem chip is a fully parallel silicon neural network. It consists of a chain of identical elements that represent the ANN neurons addressed in parallel and have their own “genetic” material to learn and recall patterns without needing any synchronization to any supervising unit. A neuron is a reactive memory which can autonomously evaluate the distance between an incoming vector and a reference vector stored in its memory. If this distance falls within its current influence field, it returns a positive classification.

The CogniMem neural network features two non-linear models for the recognition: K-Nearest Neighbor (KNN) and Radial Basis Function (RBF). In KNN mode, the notion of influence field is discarded and the network always returns a response, which is the nearest category even if the

---

[1] http://www.general-vision.com

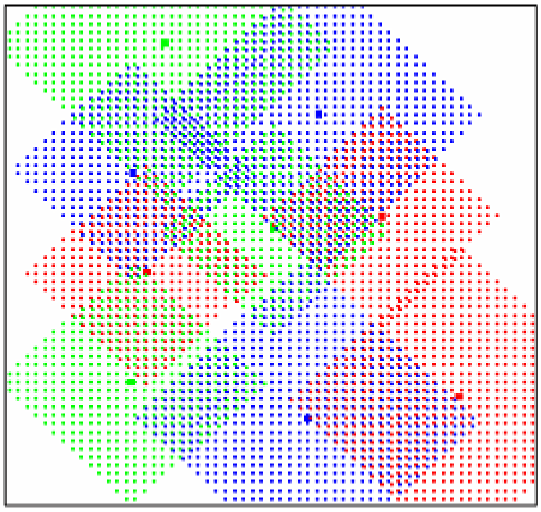

Figure 4.1.6: RBF: The Space is mapped partially with certain zones being unclassified (Black Color). The zones with multiple colors are zones of uncertainty [Vis08].

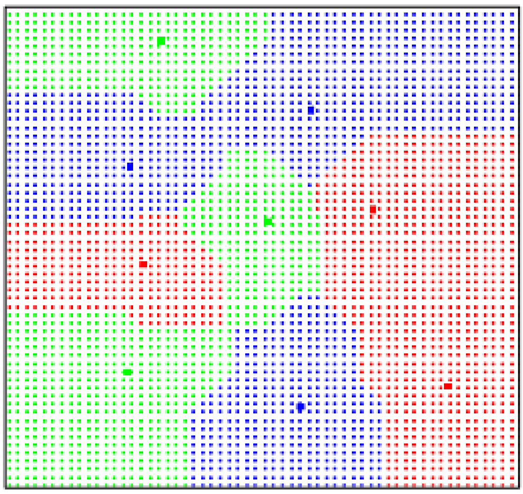

Figure 4.1.7: KNN: The entire space is mapped with a single possible category per position [Vis08].

shortest distance value can still be high since the zones of uncertainty are in-existent as shown in Fig. 4.1.7. However, in RBF mode, which is presented in Fig. 4.1.6, the response can be more refined. The classifier delivers one of three possible classification statuses: Identified, Uncertain, or Unknown. The RBF classifier is especially suited for anomaly or novelty detection where the classification uncertainty is also important.

Another important feature of the CogniMem is the Automatic Adaptive Model Generator. This feature enables the chip to build adaptively an ANN model without demanding the description of the internal setup of the network, such as the number of hidden layers or the number of neurons to be used to solve the classification problem.

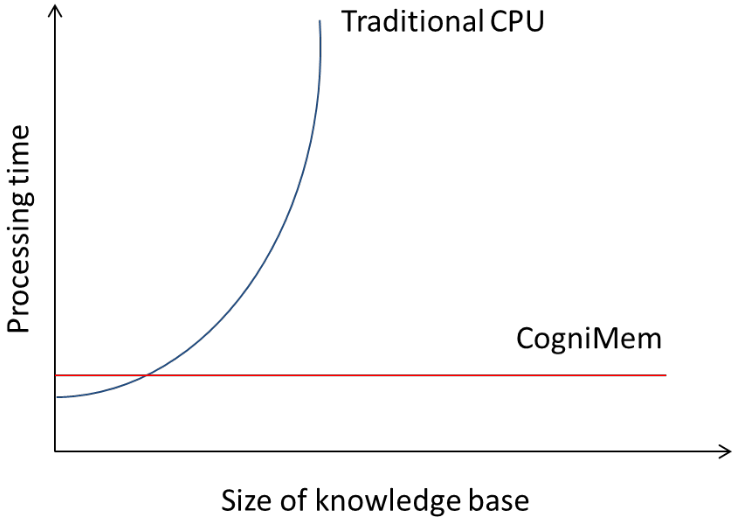

Figure 4.1.8: Processing Time Comparison Between CogniMem and Traditional CPU

The significant advantage of neuron arrangement into a parallel network is its capability to learn and recognize a vector in a constant amount of time, independent from the number of neurons. A second achievement is the ability to expand the size of the network by cascading chips. The chip used in this project runs RBF model and has 1024 neurons performing 100,000 recognitions of a 256-bytes vector/second [Vis08].

CogniMem is delivered as an embedded chip in an Image Recognition Board (IRB). The main drawbacks of the IRB are that in the current implementation, the internal module can only investigate one region of interest per frame; moreover, in case of using the CogniMem video signature extraction, a sub-sampling of the pixels in gray scale, only, inside the region of interest fits in the 256-byte feature vector. To overcome these problems, the all pre-classification process, such as hypotheses generation and feature extraction, are done in a software module running on the system's processor and CogniMem is used as an external classifier. In addition to the previously mentioned advantage, CogniMem has the ability to interact with different types of vision sensors via extended digital I/O, which means if the pre-classification process is implemented on the chip; it would enable to produce a low cost camera object recognition system on the chip.

4.1.5.2 Preparing Features Vector

We commence by preparing the feature vector for classification. The extracted features are normalized to get size invariant features to guarantee the classification generalization. By analyzing the features, it was observed that a combination of features (especially the geometrical features) obtained by the multiplication of their values deliver a new feature which in some cases can significantly help to identify the object. Therefore, the feature vector is extended with these correlation values.

Since the CogniMem can only process a vector of bytes, consequently the normalized features values are re-mapped between 0 and 255. In some cases, to represent the feature in a reasonable accuracy, two bytes were used to describe the feature.

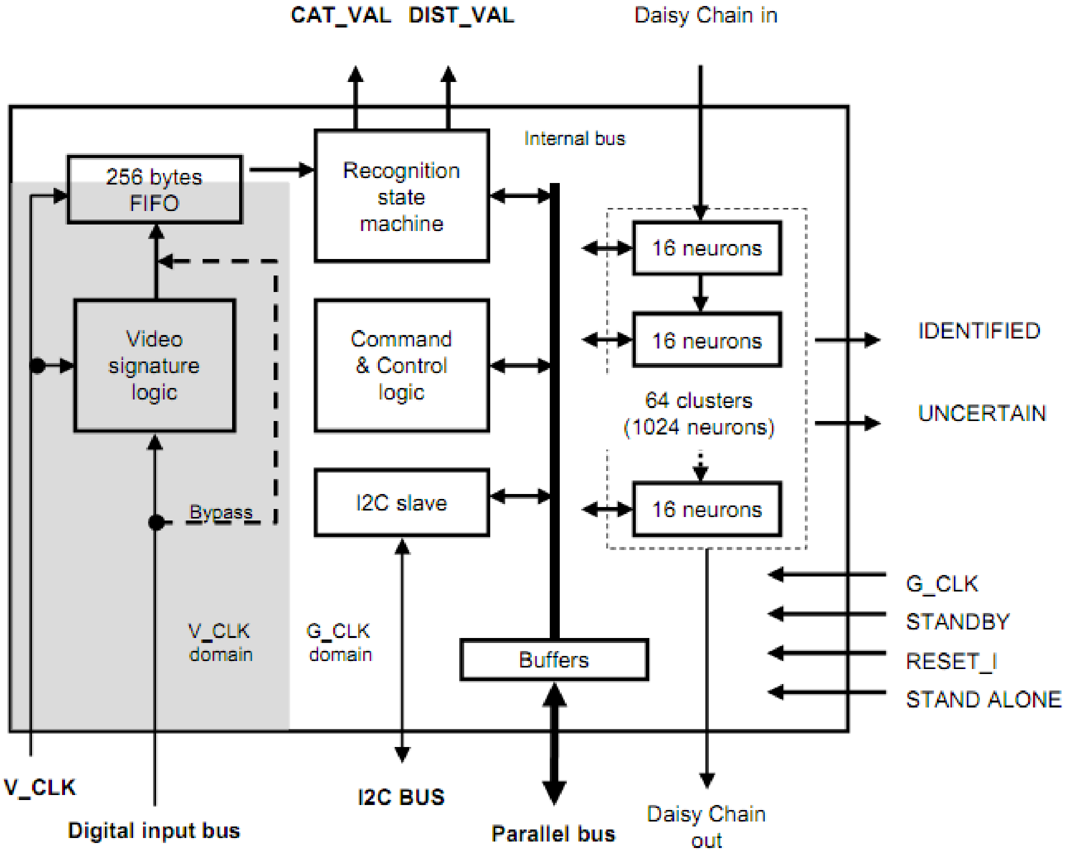

Figure 4.1.9: CogniMem Architecture [Vis08]

4.1.5.3 Classifier Training

For the training procedure, a set of 3000 feature vectors for the positive and negative samples have been generated. The set is divided randomly into two categories: a training set which includes 2/3 of the total samples while the other set containing the remaining 1/3 of the samples for the testing propose. In order to accelerate the training process, the CogniMem Emulation Kit (CMET) has been used. The emulation kit imitates all the functionality making it possible to train the classifier offline, and then the resulted knowledge base can be uploaded to the chip. The property of save and restore the contents of the neurons has been used to fine tune the classification reliability by undertaking further training procedure at any time. The internal training of the CogniMem can be illustrated in the following example.

If a sample A is taught as headlight, the CogniMem neural network first identifies if it recognizes the sample. If only one category is recognized and it is a headlight, the network discards the sample and does not take any action. If several categories are recognized, such as a tail light or a reflector, including the category headlight, the network does not commit a new neuron to store the example, but instructs the neurons responding with a category other than headlight to shrink their influence fields. Otherwise, a new neuron is committed to recognize this sample.

4.1.5.4 Primary Classification

The classification of a feature vector consists of evaluating if it lies within the influence field of one or more neurons modeling the decision space. When a vector is broadcasted to the neural network, all the neurons calculate their distance between the input vector and the prototype stored in their memory. If the distance of a neuron is greater than its influence field, the neuron excludes itself from the list of neurons recognizing the vector. Otherwise, it fires to indicate that it recognizes the vector. The similarity range is expressed with the distance value. Its dimension is a function of the type of data stored in the vector and the norm in use to calculate the distance. Several neurons can recognize the input vector. The one with the smallest distance value has a prototype in memory which is the closest to the input vector. Also, more than one neuron can fire with the same smallest distance. If they have identical categories, it reinforces the confidence level of the recognition. If they do not have the same category, they point a level of uncertainty in the recognition and potential ambiguities between certain categories. This uncertainty can be considered by reading the categories recognized by the next firing neurons, that is with the next smallest distance value. The higher the distance, the less similarity between the prototype and the input vector. If a neuron has a distance equal to 0, it means that the input vector matches exactly to its prototype. As soon as the last component of the vector has been broadcasted to the neurons, the ID (identified) and UNC (uncertain) flags are updated and written in the Network Status register. The response of all the firing neurons, if any, can be read as illustrated in Fig. 4.1.10.

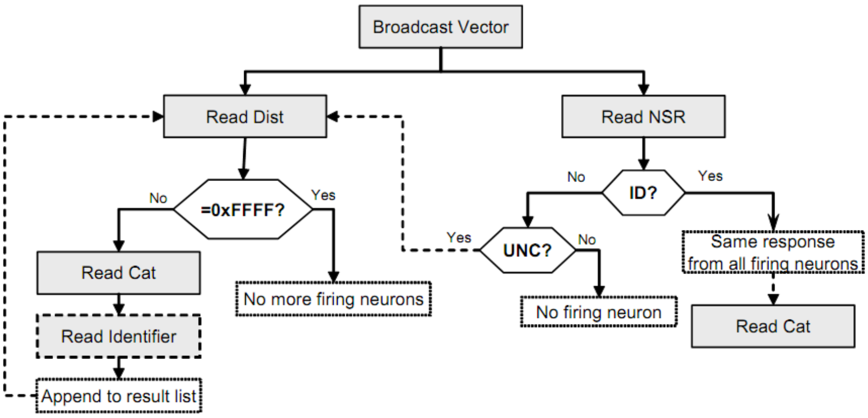

Figure 4.1.10: CogniMem Vector Recognition Procedure [Vis08]

The classification result and the uncertainty are saved in separate parameters set per category, thus each hypothesis has four classification parameters sets indicating its classification tendency. Depending on these parameters, the temporal classification category is estimated in each cycle.

4.1.5.5 Secondary Classification

This classification step depends on a heuristic classifier to use a secondary set of features, such as the optical flow and relative position to the detected lane, to confirm the categorization results of the primary classification. Based on the result of this step and the result of the primary classification, an overall classification probability is computed. In addition, for all blobs which have not been confirmed from the tracker in the current cycle, a lost count is increased while the classification quality would not to be updated.

4.1.6 Vehicle Recognition

In this stage, only objects which have been confirmed in two successive frames as a head/tail light will be processed. Based on the classification results and the pairing probability feature computed in 4.1.3.1, the object will get a new classification category; either a four-wheels or a two-wheels vehicle.

Firstly, the pairing probability between each blob and its twin, which was assigned from the previous cycles, is estimated and compared with the probabilities between the blob and the others blobs in order to find its most suitable twin. Then a rough estimation of the distance between the ego and the detected four-wheels vehicles is estimated using equation (4.1.18).

$$distance = \frac{W_{3D} \cdot f \cdot s_x}{W_{2D}} \tag{4.1.18}$$

where W_{2D} is the width of object in pixels, f is the focal length of the camera in m, s_x is camera scaling factor in pixel/m, and W_{3D} is the practical estimated value of the allowable distance between the vehicle head/tail lights which is approximately 1.2 m.

The optics of the camera is assumed to be ideally planar without any distortion. Moreover, the pixels of the camera are considered to be uniformly distributed on its opening angle. Thus, by using equation (4.1.19) the object angle θ can be computed.

$$\theta = \frac{\varphi}{2} - \frac{x \cdot \varphi}{W_{image}} \tag{4.1.19}$$

where φ is the camera opening angle, x is the x-position of the middle point of the vehicle in pixel, and W_{image} is the width of the image in pixels.

For smoothing the estimated values, an instance of (α, β, γ) filter illustrated in section 4.1.4 has been used. The estimated position is used to predict the position of the vehicle in the next frame. Of course, the distance estimation is not applicable for the two-wheels vehicles, thus for those objects the prediction is based only on the angle assuming that the change in angle between two successive frames is not too large, so that it has been modeled as an added noise. The matching between the twins from the previous cycle and current one is based mainly on the blob IDs which were given by the blob tracking function and the prediction of the filter, if available.

The main tasks of this stage can be summarized as:

- Building the object list which will be used later in the data fusion module.
- Managing the object status such as filtration object list and occlusion handling.
- Determination of quality measurements such as existence probability.
- Estimating the pairing confidence.

4.2 IDIS Object Recognition

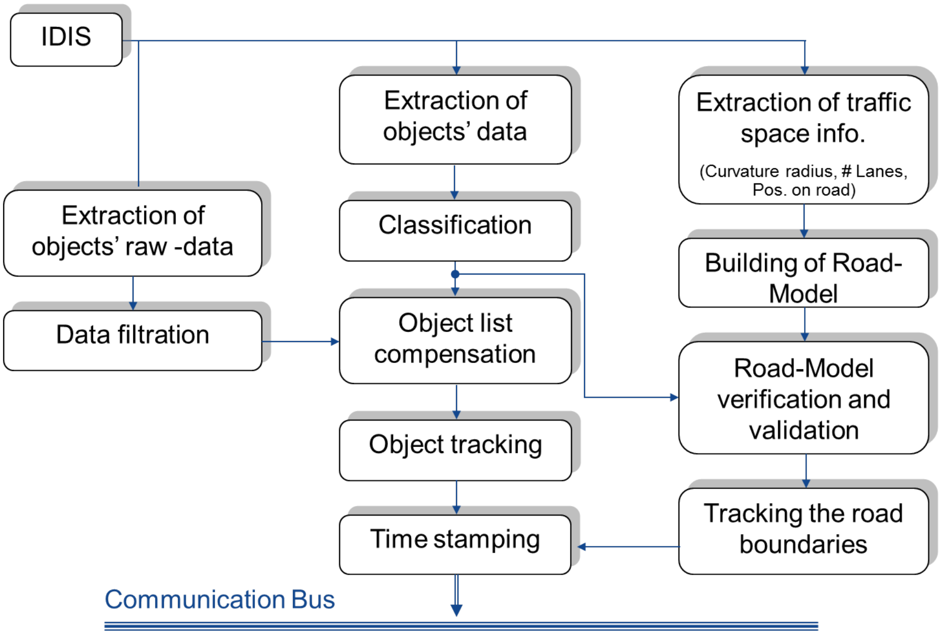
IDIS
Extraction of objects' data
Extraction of traffic space info.
(Curvature radius, # Lanes, Pos. on road)
Extraction of objects' raw -data
Classification
Building of Road-Model
Data filtration
Object list compensation
Road-Model verification and validation
Object tracking
Tracking the road boundaries
Time stamping
Communication Bus

Figure 4.2.1: Functional Flow of IDIS Object Recognition Submodule

4.2.1 Building Object List

4.2.1.1 Extracting Object Information From CAN Messages

IDIS recognition and tracking unit delivers after every 60 ms a list of the detected objects and information about their infrastructure in form of 23 CAN bus messages. Due to the limited bandwidth of the CAN bus, the list contains information about the most important 10 ACC-Objects per cycle [HLL05]. The supplied information includes the following data:

- Object tracking ID: This ID is unique and constant for each object as long as it is detected from IDIS. If the object is lost, its information will be predicted for 6 successive cycles. If it does not appear again, it will be deleted from the list and its ID will be freed. Then If it appears again, it will get a new ID.
- Object position in car coordinate system, where x direction is the distance between the object and the host vehicle, y the lateral position, while z is omitted since IDIS is a two dimensional sensor and it cannot recognize the object altitude.
- Object width.
- Object longitudinal and lateral velocity.
- Object longitudinal acceleration.

- Object lane: Based on predefined road geometry such as standard lane width, tracked object trajectory, and legitimated driving behavior, IDIS tracking unit can reliably assign a lane to each object. Therefore, the object can be assigned to own lane, right lane, left lane, or roadside.
- Object life time.
- Road curvature.
- Number of estimated road lanes.

For each object, the above mentioned data are distributed on two CAN messages. Thus, all objects CAN messages are collected in a buffer, and then a cycle validation check is performed. If this check is failed, all objects data in the buffer for the current cycle is ignored, since it contains a mixture of CAN messages from different cycles.

4.2.1.2 Extending Object List using Raw Object Data

As mentioned previously in the system hardware description, IDIS is designed originally for Adaptive cruise control assistance system, so that leading cars as well as the objects in its own lane are the most important objects for the system. Which means in some situations, despite there are vehicles or objects on the opposite direction, they will not be considered by the IDIS recognition unit. In order to solve this problem, the raw data of the sensor are extracted and filtered to compensate the delivered object list. For each channel, IDIS can detect up to 3 targets, which means about 36 possible targets. Meanwhile, almost all objects on the vehicle surroundings reflect the IDIS infrared beams, therefore the raw measurement list should be filtered before further processing in order to reduce the processing time as well as to suppress the possible noise. The filtration procedure is based on the strength of the reflected signal and the detected position of the object.

Since IDIS already provides the data of 10 tracked objects, thus regarding their raw measurements again is useless. A matching module to associate the raw data to their tracked objects has been implemented as follows.

For each tracked object, a matching gate around the object is initialized. This gate depends on the x, y position, relative velocity, and width of the object.

$$Gate = f(x, y, v, w) \tag{4.2.1}$$

If a raw measurement is found in an object gate (by using the nearest-neighbor technique), it is considered that it belongs to this object and is deleted from the measurement list. After the matching process, all raw measurements, which have not been associated to any tracked object, are regarded as possible objects candidates.

4.2.2 Object Classification

In contrary to the vision sensor, the information delivered by IDIS about the shape of the object is not sufficient to classify it precisely. Therefore, the classification will depend mainly on the dynamic properties of the object supported by its width and position [FDEW02].

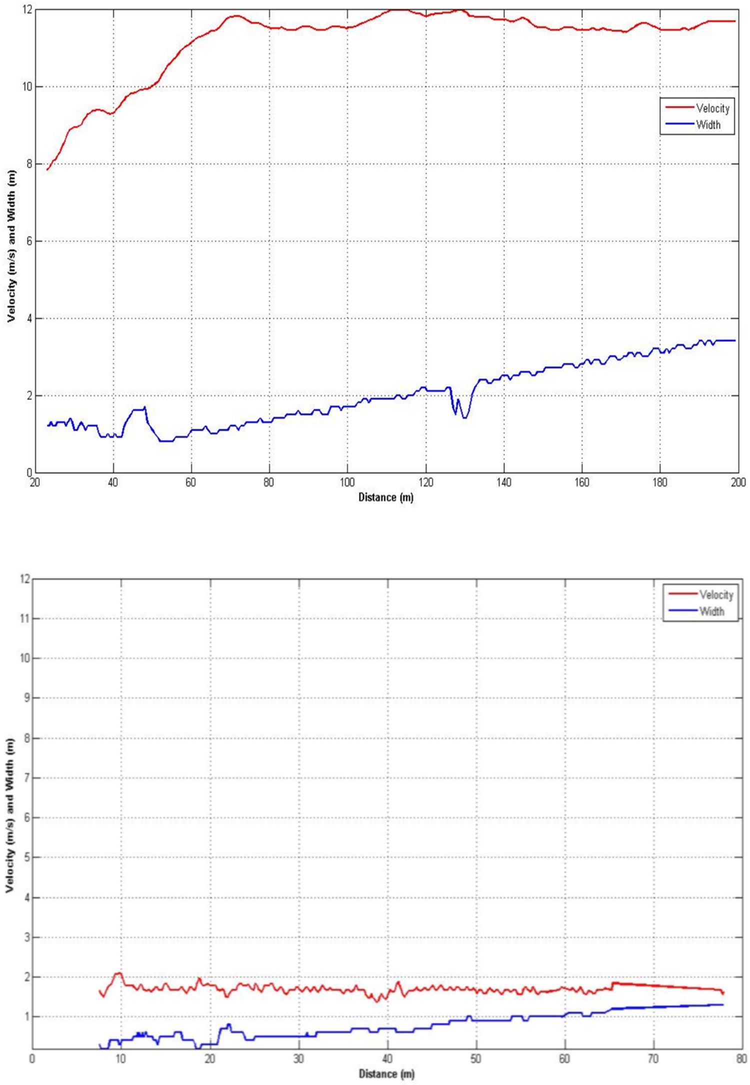

Figure 4.2.2: Velocity and Width Features of a Car (top) and a Pedestrian (bottom)

The history of the object plays also a vital role in the classification process via observing the behavior of the object for a long period of time, which leads to a more precise and robust result. According to the expected object dynamics, the following classes have been defined.

- Leading vehicle: For large width and high velocity objects.
- Oncoming vehicle: For large width and high velocity objects in the opposite direction.
- Motorcycle (Tracked): For small width and high velocity objects.
- Bicycle: For small width and moderate velocity objects.
- Pedestrian: For small width and low velocity objects.

- Unknown: For all stationary objects.

A neural network classifier like the one illustrated in section 4.1.5 was used to solve this classification problem. The following set of object's features was used to train the classifier:

1. Absolute and Relative longitudinal velocity: Due to the ego velocity error, estimating an accurate absolute velocity of the object is not possible, especially when the host vehicle drives with high velocity. Thus, the relative velocity is considered as a measure of the allowed error to judge if an object is stationary or not.
2. Absolute and relative lateral velocity: Small object, for which the lateral velocity $>>$ longitudinal velocity, tends to be a pedestrian crossing the road.
3. Absolute acceleration.
4. Width.
5. Position (x,y).
6. Maximum and minimum absolute velocity.
7. Average of velocity from the object history.

Another set of features, which can be regarded as a secondary features, are used to confirm or to append the classification results of the neural network. These features are:

1. Object lane: Position on the lane can be used to confirm the classification results, for example objects classified as oncoming vehicle should be found on the left lane.
2. Moved and stopped flag: It is a flag to indicate that the object has been already classified as a movable object but it has stopped now, therefore the object keeps its last classification category.
3. Classification confidence based on the object age and the fluctuation rate of classification category: If the classification confidence is larger than a certain threshold, the object is considered as robust classified and it should not be classified via the neural network in the next cycles.

4.2.3 Road Classification

The driving road is classified as a straight or a curved road depending on the estimated radius of the curvature [FL02, FDW02]. If the radius of curvature determined by IDIS is approximately 10000 m the road is considered to be straight, otherwise the host vehicle is driving in a curve. The curvature estimation depends on the yaw rate and steering angle of the host vehicle. Based on the road curvature and the number of lanes, a model of the road is initialized. Then by analyzing the classified object trajectory and its distribution on the road, the driving path curvature and lane association are verified.

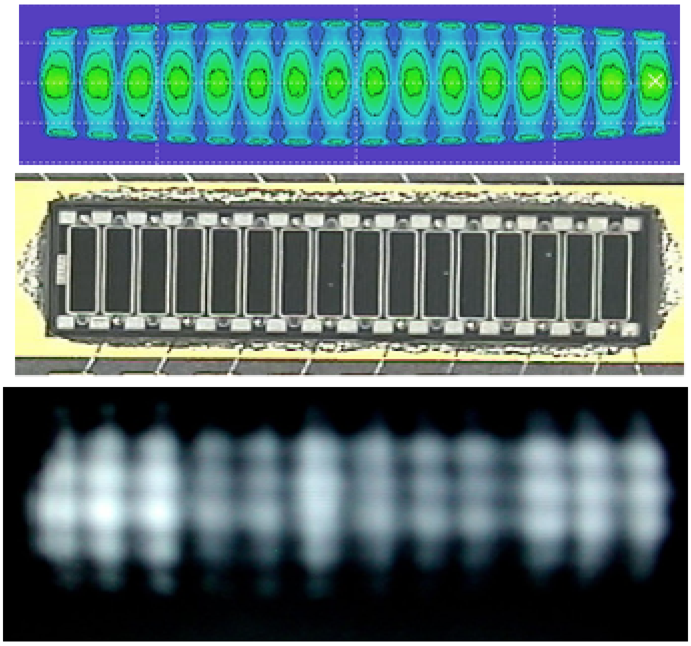

Figure 4.2.3: Infrared Beams of IDIS Sensor, Photo: Hella

4.2.4 Tracking

As mentioned before, IDIS uses 12 fixed infrared emitters to illuminate the traffic space in front of the vehicle. For each emitter, there is a dedicated receiver which is turned on simultaneously with the emitter. The gap between two successive emitters or receivers, as shown in Fig. 4.2.3, is one of the drawbacks of such object detection systems. Due to such a gap, IDIS cannot continuously detect small objects like a pedestrian crossing the road, which means crossing over the infrared channels. Therefore, an extra local object tracking dedicated to solve that problem has been developed. The position and velocity of the missed objects which exist in the gaps between the beams have been predicted using kalman filter (for technical details refer to section 5.2). Since IDIS objects are already tracked, thus only relevant objects such as pedestrians and small objects have been regarded as relevant to the local tracker. In each cycle, the small objects in the driving lane or the ones which move with a high lateral velocity in relation to their longitudinal velocity, are extracted from the IDIS object list and saved in another intermediate list. The intermediate list (IL) is matched with the local tracker object list based on the tracked object ID. The state can be expressed as follows.

$$X_{(k+1)} = AX_k \tag{4.2.2}$$

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ vx_{k+1} \\ vy_{k+1} \\ ax_{k+1} \\ ay_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta T & 0 & {}^{\Delta T^2}\!/_2 & 0 \\ 0 & 1 & 0 & \Delta T & 0 & {}^{\Delta T^2}\!/_2 \\ 0 & 0 & 1 & 0 & \Delta T & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta T \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ vx_k \\ vy_k \\ ax_k \\ ay_k \end{bmatrix} \tag{4.2.3}$$

where ΔT is the cycle time (60 ms) and k refers to the cycle index. The states vector $[x, y, vx, vy, ax, ay]$ represents the position, the velocity and the acceleration components, respectively.

The number of cycles n, in which the object should be tracked in absence of new physical measurements, depends on its position and relative lateral velocity as shown in equation below.

$$n \simeq \frac{x \cdot tan(\theta)}{vy \cdot \triangle T} \tag{4.2.4}$$

where θ is the gap angle ($\theta \simeq 0.5°$).

4.3 Vehicle Detection Via Car-to-Car Communication

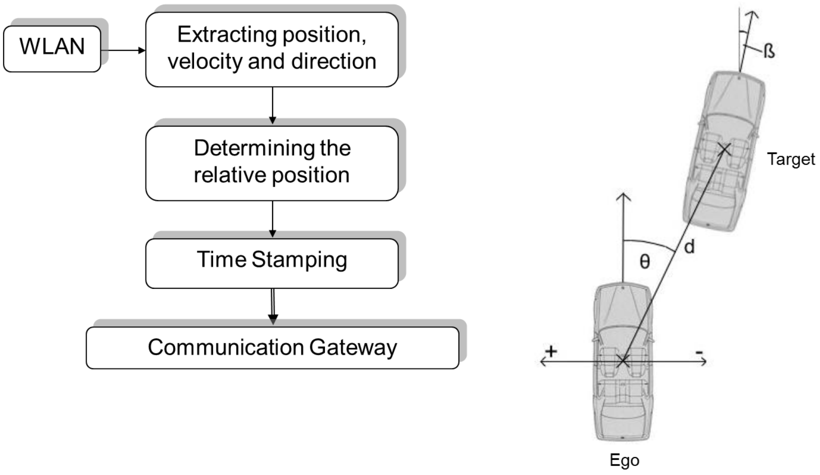
WLAN
Extracting position, velocity and direction
Determining the relative position
Time Stamping
Communication Gateway
ß
Target
θ
d
+
-
Ego

Figure 4.3.1: Functional Flow of Car-to-Car Communication Submodule

The Car-to-Car Communication function is used to decode the messages received via the wireless module to extract information about the target vehicle. The function starts by estimating the relative position d of the target and the orientation angle θ to the host vehicle. If the target is not in the illumination range of the host vehicle headlamp, its data will be neglected, for example when the target is behind the host vehicle or outside the azimuth angle of the headlamp. Then, the relative driving direction β is estimated to determine if the target will be affected by the ego headlamp. The mathematical equations to estimate the above mentioned parameters will be presented in the next section.

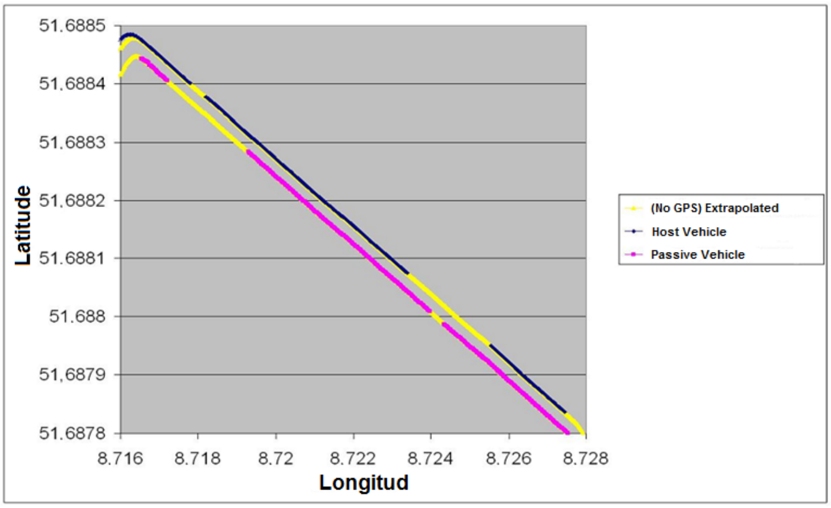

Figure 4.3.2: Coordinates Exchange using Car-to-Car Communication

4.4 Lane Estimation using GPS and Digital Maps

GPS accuracy depends on the quality of the pseudorange and carrier phase measurements as well as the broadcast navigation data. Where the pseudorange time is equivalent to the difference between the receiver clock reading when the signal (i.e., a particular code phase) was received and the satellite clock reading when the signal was sent [ZJ95]. Meanwhile, carrier phase measurement is another technique used by the GPS receiver to track the position of the satellites by measuring the wave length, which is the period of the carrier frequency times the speed of light [Hon07]. There are a number of sources of error that corrupt these measurements, such as:

- **Atmospheric interference:** The ionosphere and the troposphere layers, which extend more than 1000 km over earth surface, may cause deflection in the GPS signal and thus a time delay in receiving or transmitting the GPS signals.
- **Trilateration errors:** GPS receivers estimate the position using the trilateration concept, which measures the distance between different satellites and then intersect the spheres to get the point of intersection that lies on the earth. Errors may arise when reading from more than two satellites are not available or when the available satellites lie near each other, which may produce a surface of intersection.
- **Reflections errors:** The GPS signals may reflect from the buildings and other objects producing a time delay in receiving the signals.
- **Satellite clock errors:** The satellites contain atomic clocks that control all onboard timing operations, including broadcast signal generation. Meanwhile, the GPS uses commercial quartz clock for synchronization. However, these types of clocks are not exact and possess a drift with temperature variation. This temperature affects the correct estimation of the distance, and thus the position will be assigned wrongly using trilateration technique.

In order to show the deviation in the street coordinates, two sets of tests were recorded from a moving vehicle extracted from the Katastar maps and the second recorded from the GPS receiver. The tests were repeated along the streets for different vehicle's speeds and a great deviation is observed from the real street path as shown in Fig. 4.4.1.

Figure 4.4.2 shows the estimated deviation in meters of 4 different sensors. As can be noticed, the GPS deviation varies in an unpredictable matter. In the next section, an approach of digital map data matching will be presented to enhance the GPS positioning to be used in the threat and situation assessment.

4.4.1 Creating of the Digital Map

Mapping GPS coordinates on a digital street map will enhance the position accuracy of the GPS receiver. It is assumed that the host vehicle lies on the street, thus the lateral bias in the GPS signal will be minimized. In the context of this work, a digital map has been manually constructed to cover all the routes in which the tests should be performed. The maps are constructed by using a very precise Differential GPS (DGPS) with a maximum 10 cm deviation. With the help of these maps, GPS measurements can be directly assigned to the center lines assigned by the digital maps. The system searches for parallel routes, if they exist, a perpendicular line will be projected to the nearest lane lines center, as shown in Fig. 4.4.3. However, there are far more

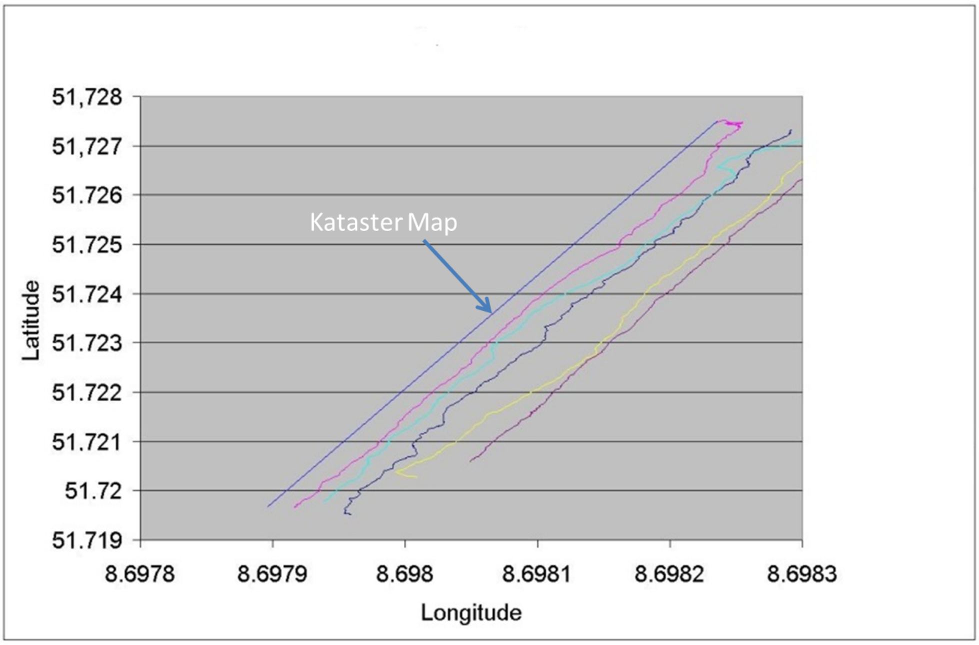

Figure 4.4.1: GPS Recorded Data From a Moving Vehicle

possibilities for this method. For example, if the road has two lanes, then the question arises that to which lane line should the GPS position be assigned.

Figure 4.4.3 shows two lane center lines with a possibility that the real position lies on one of them. The easiest way to solve this dilemma is to correct the GPS position with the nearest lane line, neglecting the possibility that the vehicle can lie on the other lane line. The correction of this assumption can be tested in the data fusion algorithms by checking and comparing the constructed lane lines and the detected ones from the vision system, as has been introduced in section 4.1.1. Nevertheless, lane lines matching algorithms can provide an optimum way to assign the vehicle rightly to its corresponding lane line.

4.4.2 Driver Path Extraction using Digital Map

In the last section, we provided a brief illustration of the idea behind using the digital map in GPS position enhancement. Moreover, digital maps can also be used to extract the information of the leading driving path. First, a data base has to be built to include information of the leading path, as depicted in Fig. 4.4.4.

In order to reconstruct the driver path from the digital map, a search region is defined to extract the GPS points in the locality of the reference point. The extracted region is used to minimize the processing time performed in correlating the GPS data. As stated earlier, the enhancement algorithm searches for parallel lane lines, which means that the digital points should be first transformed into the vehicle's coordinates to detect which lane lines are parallel to the vehicle. To derive the distance between the reference point and a digital map point, the following procedure is followed.

The latitude angle is converted into kilometers. The angle should be small so so that the function can guarantee a good approximation. Figure 4.4.5 describes "x" to calculate the ellipse arc. The

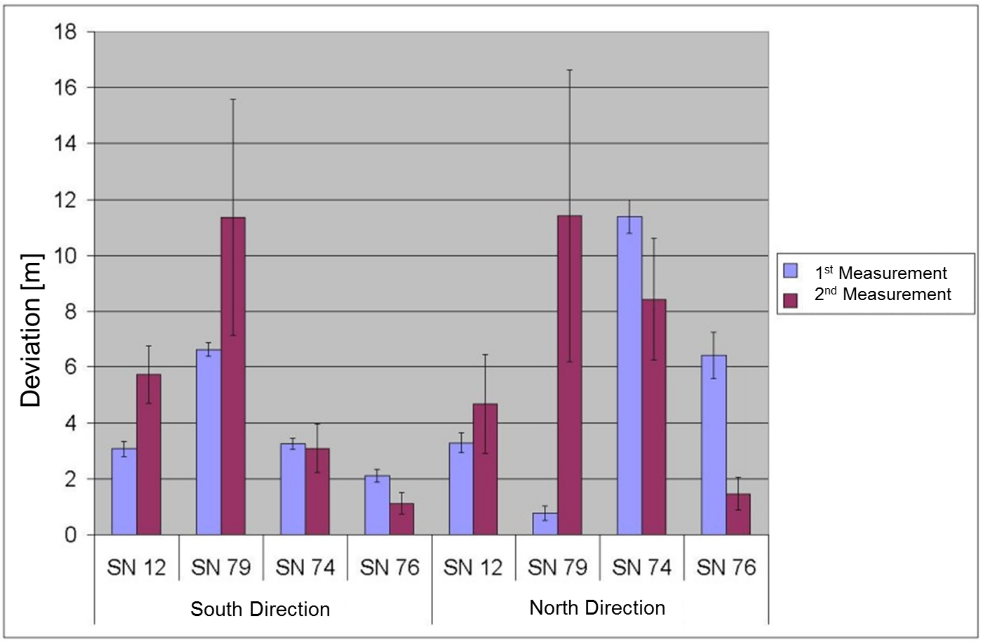

Figure 4.4.2: Deviation in GPS Tests in Meters

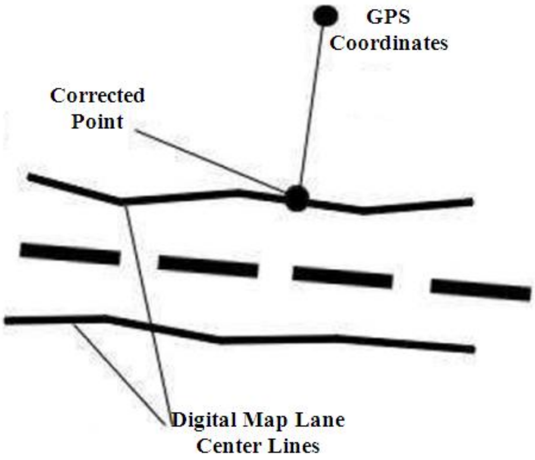

Figure 4.4.3: Digital Map Matching

replacement of the ellipse by a circle (with the radial distance of the ellipse to the appropriate place), provides a sufficiently precise approximate result. This applies, however, only for very small angle. First, the radial distance r of the ellipse is calculated, as a function in the latitude angle (α), using the following equation:

$$r\left(\alpha\right)=\sqrt{\frac{\left(a^2\cos\alpha\right)^2+\left(b^2\sin\alpha\right)^2}{\left(a\cos\alpha\right)+\left(b^2\sin\alpha\right)}} \tag{4.4.1}$$

where a and b are the vertical and the horizontal earths radius, respectively. The ellipse shown in Fig. 4.4.5, represents the distance between the GPS reference point (host vehicle) and one of

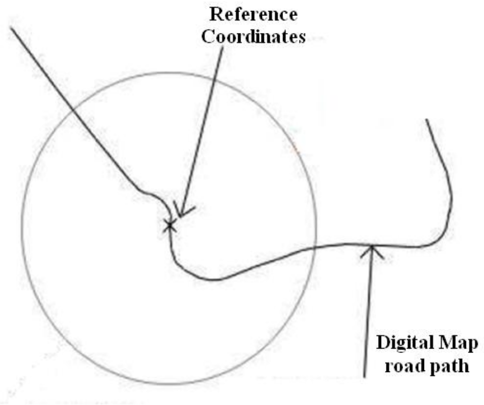

Figure 4.4.4: Buffering Digital Map Point

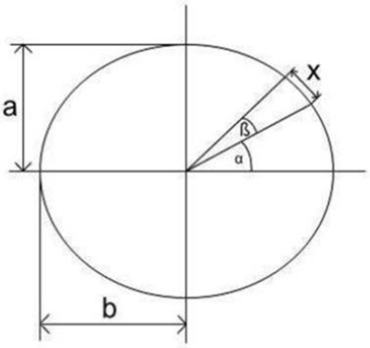

Figure 4.4.5: Distance Between Two Points

the digital map points. So, the longitudinal difference can be calculated using the equation:

$$X = \frac{(Lon - Lon_{ref})\ 2\pi\, R\ \cos\left(2\pi Lat_{ref}/360\right)}{360} \tag{4.4.2}$$

Analog to the above equation, the difference in Latitude can be converted into km using the equation:

$$Y = \frac{(Lat - Lat_{ref})\ 2\pi\, R\ \cos\left(2\pi\, Lat_{ref}/360\right)}{360} \tag{4.4.3}$$

In equation (4.4.2), X is determined by the subtraction of two longitudinal values. The same happens for Y in (4.4.3). However, the section of Y is determined by the subtraction of the two latitudes results. The distance can therefore be extracted using the statement of Pythagoras as follows.

$$Dist = \sqrt{X^2 + Y^2} \tag{4.4.4}$$

The X and Y mentioned above are the coordinates of a system's origin determined by the vehicle coordinates. These coordinates still need to be referenced relative to the direction of host vehicle. The new reference system has to be rotated by an angle γ, which is the angle resulted from subtracting the angle between the two GPS points and the COG angle of the host vehicle. Considering the angle is obtained using the subtraction procedure, the following conditions should be met:

- If the angle γ is less than -180 degrees, 360 degrees will be added to it, i.e.,$\gamma = \gamma + 360$.
- If γ is greater than 180 degrees, then 360 degree should be subtracted from it i.e., $\gamma = \gamma - 360$.

These conditions result in a negative angle (from 0 to -180 degrees) for the right side and a positive angle (from 0 to 180 degrees) for the left side of travel. The X and Y can be redefined due to the new system's origin as:

$$Y = r \cos \gamma \tag{4.4.5}$$

$$X = -r \sin \gamma \tag{4.4.6}$$

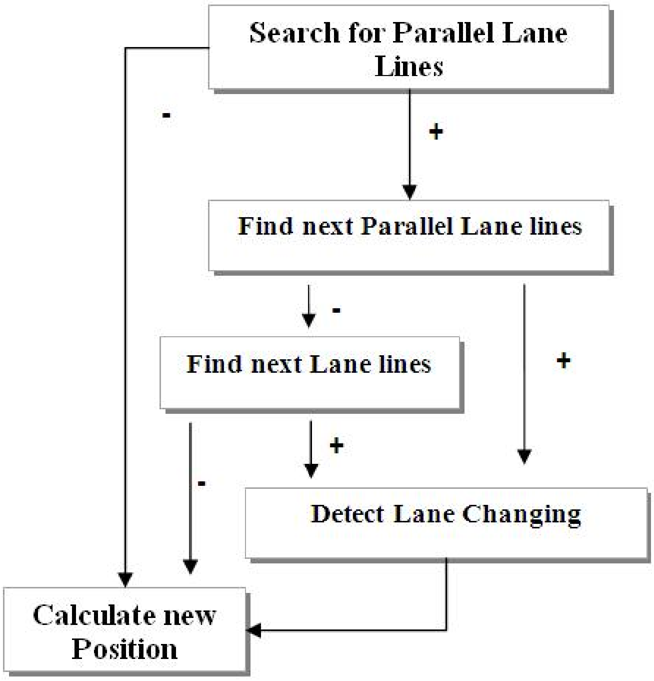

Figure 4.4.6: GPS Correction Algorithm using Digital Map

Figure 4.4.6 describes the process of correcting the GPS measurements. The process can be explained as follows.

1. First, the proposed algorithm tries to identify a parallel track to the reference point obtained from the GPS measurements. As a rule, there are two points that fulfill such a property. A close observation of GPS measurement reveals that the extracted coordinates lie within two control points from the digital map inside a distance of 1.1 m.
2. If none of these points are found, the process begins again to search for the nearest coordinates, but this time the distance greater than the previously stored value is assigned. If two points have been found, then a line connecting them is examined if it is parallel to the driver travel direction.

3. If no parallel path has been found, then the current GPS coordinates are discarded and a new coordinate from the last one, direction, and speed is calculated based on the estimated distance from the last recorded GPS measurements as follows.

$$Dist = velocity \times 1\, sec. \tag{4.4.7}$$

$$X = \frac{(Dist \times \sin CoG)\ 2\pi\, R\, \cos\left(2\pi\, Lat_{ref}/360\right)}{360} \tag{4.4.8}$$

$$Y = \frac{(Dist \times \cos CoG)\ 2\pi\, R\, \cos\left(2\pi\, Lat_{ref}/360\right)}{360} \tag{4.4.9}$$

$$Lon_{new} = Lon_{ref} + X \tag{4.4.10}$$

$$Lat_{new} = Lat_{ref} + Y \tag{4.4.11}$$

As the distance can be estimated from the average speed of the vehicle multiplied by the time of the last GPS measurements, which is 1 sec (the update rate of the GPS receiver). X and Y transforms the distance into longitudinal and latitudinal, which should be added to the last GPS measurement to extract the new position.

4. If the first lane line has successfully been found, then a search will be performed to find the next lane lines. Assumption is first made that the second lane line is parallel to the driving direction. If not, then we may consider the case of lane changing.

4.4.3 Lane Departure Detection

As a crucial condition for Lane changing detection, the position of the vehicle has to change relative to the parallel lane. This behavior can be examined by recording the change in the CoG angle, as can be noticed from Fig. 4.4.7.

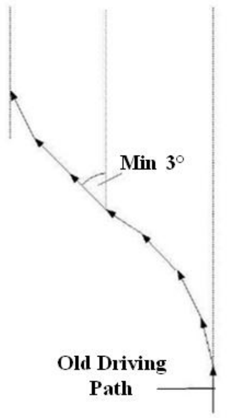

Figure 4.4.7: Lane Departure Detection Criteria

The lane changing can be detected, if the CoG angle is changed more than 3 degrees. This action will fail the digital map matching processes, which were described in Fig. 4.4.6. Thus the GPS position will be updated only by the distance covered by using equation (4.4.7). Since, the accuracy of the GPS measurement is low, the lateral bias may be great, so the initial assumption of the locating lane would be wrong. However, another consideration is made to actualize the vehicle's position in a way that the assumed position will not contradict with the road map.

Figure 4.4.8 shows a successful detection of a lane departure maneuver.

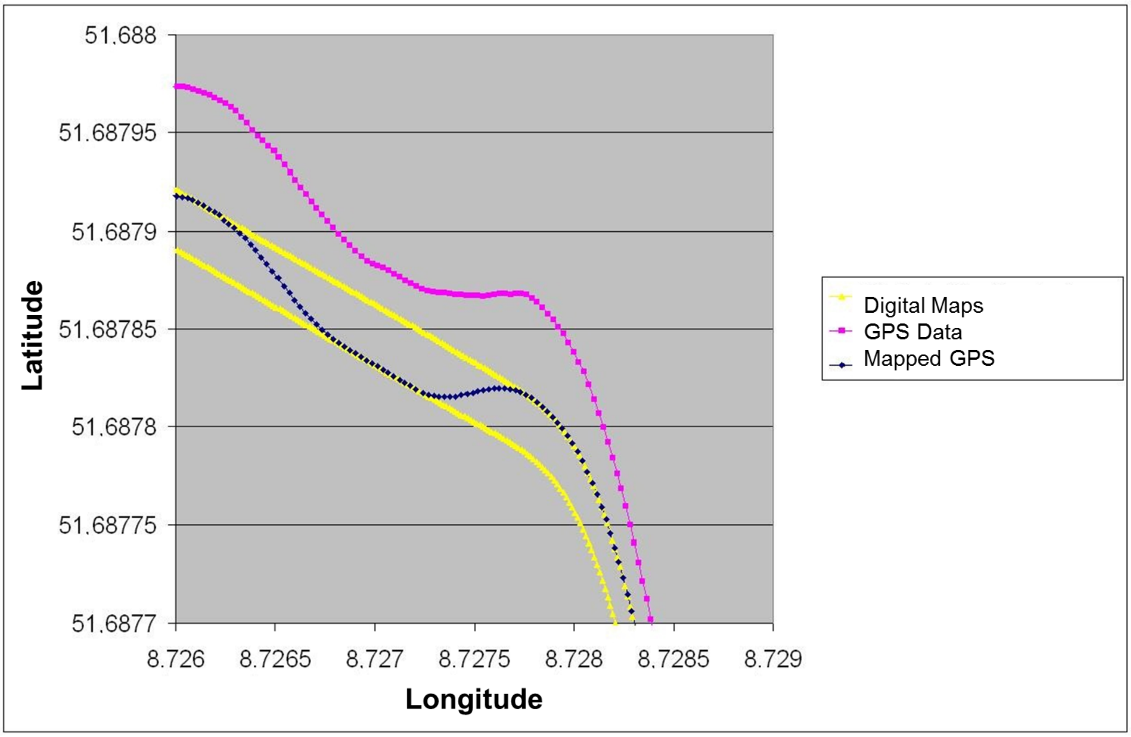

Figure 4.4.8: Detection of a Lane Departure Maneuver

Chapter 5

Sensors Data Fusion

5.1 Object Data Fusion

Generally, Data fusion is defined as a multilevel multifaceted process dealing with the automatic detection, association, correlation, estimation, and combination of data as well as information from single and multiple sources. From another view, data fusion represents the techniques to gather and combine information from different allocated sensors to provide a generalized knowledge about the sensed environment. The resulting information is more satisfactory than the one extracted from the raw data of each sensor [HM04].

Other models such as Bowman model [BSW99] express data fusion between the sensors as a process of estimation and prediction of an entity based on the measurements made from the detection algorithms of each sensor.

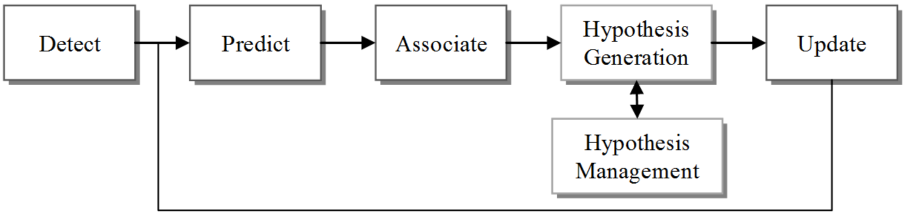

Figure 5.1.1: Bowman Model

Figure 5.1.1 shows the proposed illustration of data fusion according to Bowman model. The model divides the data fusion into a series of sub-processes.

- **Detection Phase:** In this sub-process, measurements from the sensors are provided to the data fusion.
- **Update and Predict Phases:** These two sub-processes constitute the main parts of the tracking algorithms. In the tracking process, the objects can be considered to have two distinctive state vectors, namely kinematics (e.g: position, velocity, etc.) and classification (which represents the identity of the detected object, based on the processing algorithms of each sensor). In addition to the issues of the measurements and the number of the detected objects, each sensor has a degree of uncertainty associated with it. So, instead

of representing the measurements with deterministic values, a probabilistic term will be included to describe these uncertainties.

- **Association Phase**: It is defined as the process of correlating sensors data to each other. From the previous phase, each Object has been assigned to a tracker, which mainly predicts the most probable kinematic states in the next cycle (i.e., the prediction phase). This prediction is essential to correlate the new measurements from each sensor to the previous ones.

- **Hypothesis Generation and Management Phases**: When multiple objects from different sensors co-exist within a certain region, a data association sub-process cannot correlate these objects correctly, unless hypotheses are generated as key elements of final integration process. As mentioned previously, the Bowman model is only a simplification of the data fusion process. In the next sections, detailed illustrations of the network topology, communication models, and data association algorithms will be introduced.

5.1.1 Architecture

The main issue in designing a data fusion system is to select an appropriate architecture [Ess08]. The choice of this architecture is not arbitrary, as it depends mainly on many factors, such as

- The type of sensors used.
- The nature of the observed states.
- Finally, the type of the application in which data fusion will be applied.

The design choice affects the quality of the fused system as well as it determines the algorithms or techniques that may be implemented if an architecture is adapted for use.

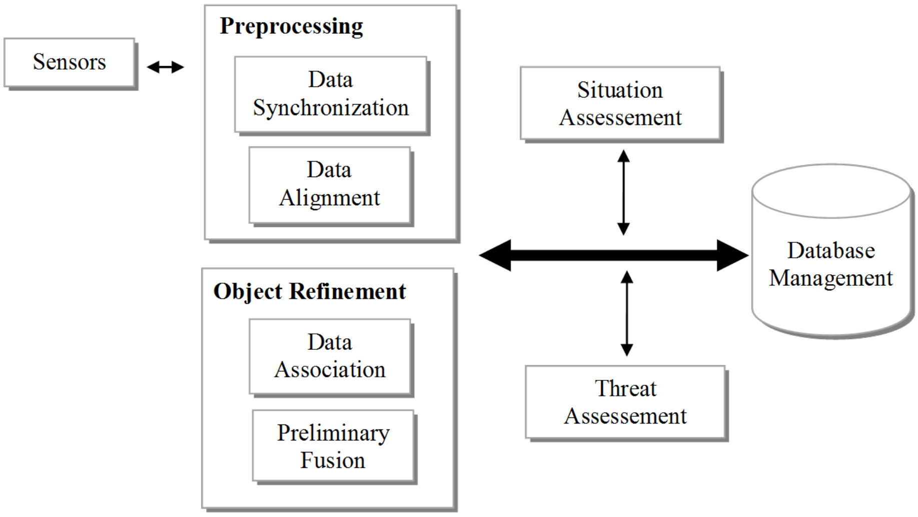

Figure 5.1.2: Joint Directors of Laboratories Model

Figure 5.1.2 shows the basic functional architecture, known as the Joint Directors of laboratories (JDL) model, which is presented by Hall [HM04]. The diagram illustrates the functions of the main sublevels of the data fusion algorithms. As preprocessed sensor data is provided to the Data Fusion Unit (DFU), it will be processed within the following levels:

- **Level 0** (*Preprocessing Phase*): Deals with the synchronization and alignment of the sensors data, which means transforming sensors data into a common spatial reference frame taking into consideration the time difference between the measurements made by the corresponding sensors.
- **Level 1** (*Object Refinement*): It tackles the problem of correlating observations from multiple sensors to the targeted objects. Data association can simply be described as merging two observations into one, as if they describe the same object in the real world. Preliminary fusion is a subfusion process, such as the one used in estimating the geographical vehicle position by integrating the GPS data and the onboard vehicle dynamics sensors as will be illustrated in section 5.2, or in estimating lane and road geometry to be used later in Level 2 and 3 in the data fusion algorithms.
- **Level 2** (*Situation Assessment*): It is the high level of data fusion. This level aims to analyze and understand the entities relationships with each other and with the road geometry. This process involves recognition of patterns and context-based reasoning based on pre-described knowledge basis.
- **Level 3** (*Threat Assessment*): It involves interpreting a situation from the consequences point of view. This level assesses the potential threats of the situation. Alternative hypotheses are generated and projected into the future to determine the likely courses of action for engagements as well as the consequences of these courses of action.

Although, the JDL Model is a generalized functional level illustration of the data fusion process, there exists another categorization of the data fusion unit based on the hardware and software architecture. The basic architectures, presented by Hall, describe the data flow and the communication model between system nodes. According to Hall, three basic hierarchical representations describe the interrelation between system's nodes. In the following, these architecture are briefly explained.

- **Centralized Architectures**
  Based on this architecture, a central processor unit is fully responsible for collecting measurements from sensor nodes and processing as well as interpreting the obtained information. Since, the central processor is fully aware of the information from each sensor, tracking and association algorithms can be performed once as shown in Fig. 5.1.3. Consequently, it reduces the errors in the classification, since the results can be validated from other sensor's raw data. Centralized architecture may also include local fusion center to decrease the amount of tasks performed by the central processor. Inefficiencies of this method can occur due to the large amount of data that needs to be transferred in real time between the sensors and the central processor. In addition, all the sensors data should have the same identity or measure the same physical quantity which is suitable to combine the data of sensors like lidar, radar, and Laserscanner.

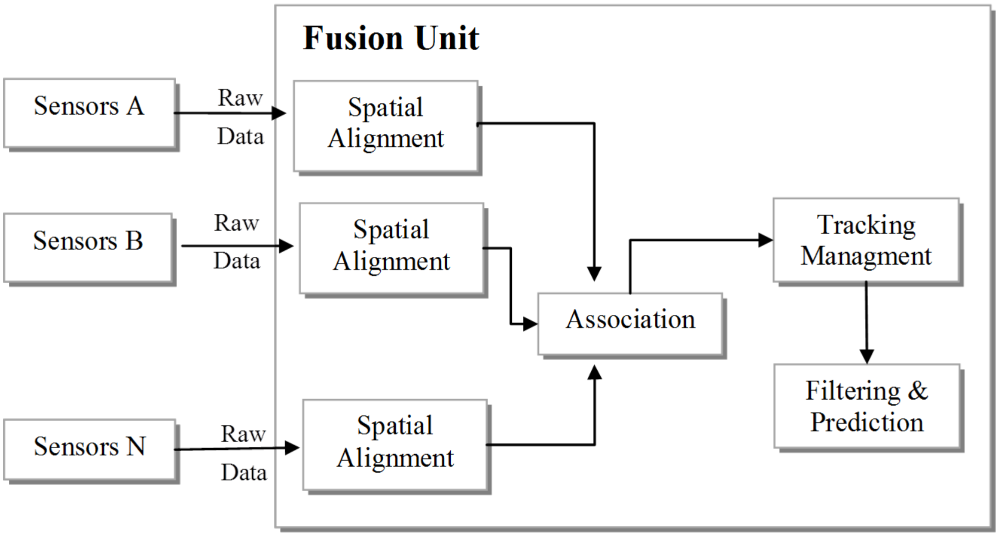

Figure 5.1.3: Centralized Architecture

- **Decentralized Architectures**
  In contrary to the centralized architectures, decentralized architectures offer better ways to overcome some of the problems and limitations associated with centralized hierarchy [HM04]. In this approach, the preprocessed sensors data serves as the inputs to the DFU and the sensors may be connected to each other to provide communication between several sensor nodes. In addition, the performance will not be dependent on any particular central processor, which means that system communication is robust against sensor failure and modular.

- **Hybrid Architectures**
  It is a combination of centralized fusion and decentralized fusion. The hybrid architecture keeps all the advantages of the centralized architecture and additionally allows the fusion of tracks coming from individual sensors in a sensor level fusion process. This type of architecture is adopted to design the data fusion of the current system due to several factors that will be discussed in detail in the next sections.

Figure 5.1.4 shows the information flow and the sequence procedures, which the data fusion algorithms go through. According to the proposed architecture, each sensor process its data before reporting it to the DFU. As in camera and IDIS modules classification and tracking algorithms are performed to the measured raw data and then a tracked objects list is reported to the DFU. On the other hand, GPS, C2C, and vehicle sensors units send their raw data to the DFU instantly. All the reported data are saved in the shared buffer within the DFU before applying any algorithms.

Considering the fact that the sensors work in an asynchronous manner; the adapted communication model seems to be the optimal solution in handling large amounts of sensor's data. Furthermore, sensors nodes do not communicate with each other, rather they communicate only

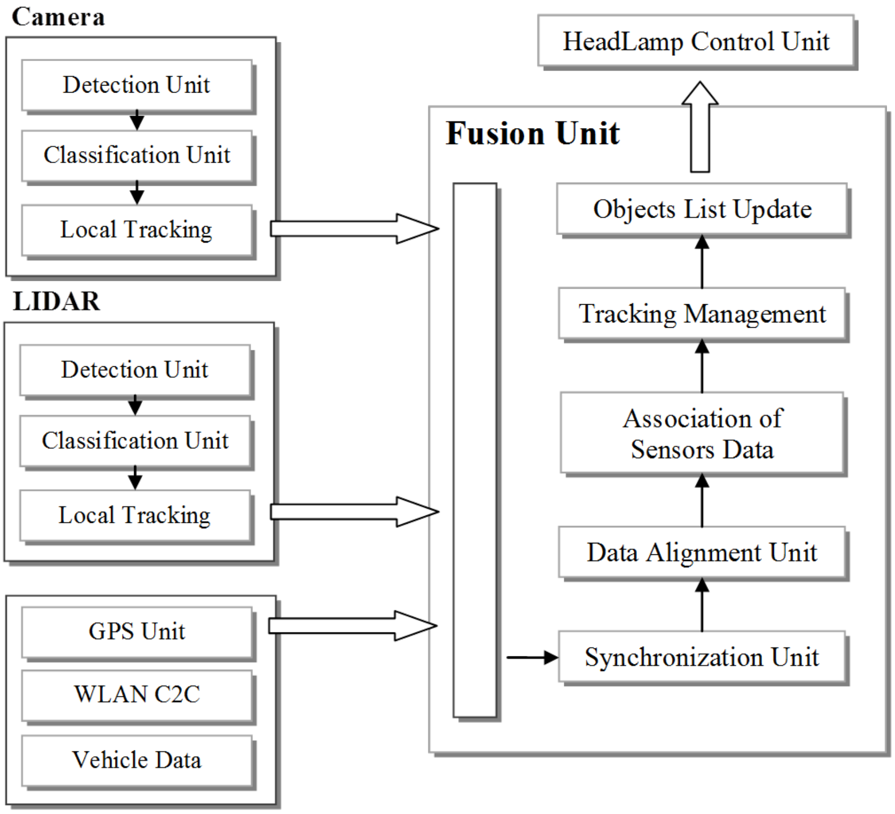

Figure 5.1.4: Data Fusion Process Diagram

with the DFU. This communication model makes the system robust against sensor's failure such as (node's communication failure). In this case, the DFU tries to resynchronize with the faulted node; meanwhile it optimizes the matter in which it handles the available data from the corresponding available sensors.

The functionality of the DFU is considered to be responsible mainly for the following tasks:

- Synchronizing the sensors data: In this task the communication protocols stamp the sensors data and resynchronize the faulted nodes as will be shown in the next section..
- Correlating the data to a unified spatial reference: This process is also referred to as data alignment, which will be described in section 5.1.3.
- Update and track objects: The tracking algorithm is essential while associating the sensors data. It will be explained in section 5.1.4.

5.1.2 Communication Model

The communication between DFU and the sensors submodules is performed using a traditional Client-Server relationship based on the User Datagram Protocol (UDP). The model synchronizes the data packets and performs different configuration tasks based on the sent messages between the system nodes. Depending on these messages identifiers, the corresponding node will develop a certain action, as will be described later in this section. Synchronization deals with timekeeping matter between different components of the system. It can be thought as the need of organizing the systems different components. We consider the problem that the DFU receives a data packet from a sensor node. Before processing the received data, certain questions should first be answered, e.g., what was the time when the raw data is recorded?, which time reference has the recording sensor?, and is the data synchronized with the other received data packets from other sensors?.

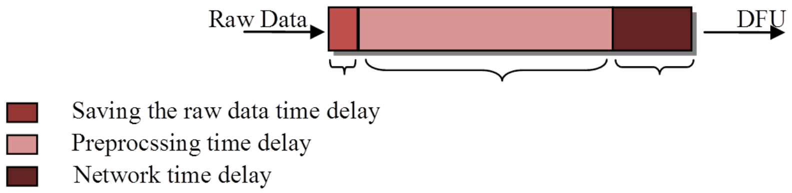

Figure 5.1.5: Time Delay Between Sensors Node and DFU

Figure 5.1.5 shows the time delay of a data packet from the time when the raw data was received by a sensor node to the time when the packet is received by the DFU. As shown, raw Data is recorded by the sensor node, preprocessing algorithms are performed, and then the data is received by the DFU with a network-imposed time delay. This time scale shows the main delays of the data till it has been received by the DFU, but the question remains; which time reference the sensor has? In the work described by Lamport [LMS85], different synchronization schemes are mentioned that have been used in different sensor networks. The schemes are briefly summarized as follows.

1. **Peer-to-peer schemes**: They constitute traditional synchronization schemes, which assume the existence of a global timescale usually realized by a master clock.
2. **Explicit schemes**: Most of those time synchronization schemes aim to keep the clock synchronized at all times. Applications assume that they can query the clock at any time that is synchronized, which means that applications can perform an explicit conversion to another timescale.
3. **Virtual time schemes**: This algorithm avoids the need of an explicitly synchronized clock. It uses a virtual time, as each sender will include in the sent message the time stamp of the sender, thus the receiver node can reset its local clock accordingly by using a value greater than the value in the received message due to the propagation delays.

In order to solve the synchronization problem, a hybrid scheme of virtual time and explicit time is adapted. Camera and IDIS modules are explicitly synchronized with the DFU, which means that they reset their clock to match that of the DFU timescale. However, GPS and C2C module cannot be re-synchronized, as they are hardware configured to send their own time only. So, it is the task of the DFU to adjust its clock according to the time reported by GPS and C2C. The time Synchronization process is performed by the DFU during the system's initialization in the following sequence:

- The server initiates connection to different sensors by sending a message which contains the main server local time.
- Clients receive that message and set their time to match that of the server and send their time back to the server.
- Server then receives every node's local time and calculates the delay over the network.
- Server orders the sensor nodes to begin sending their data packets.
- If an out of synchronization error is reported due to any unexpected time difference between the time of raw data arrival and the receiving time at the DFU, the server re-initiates the synchronization process.

Figure 5.1.6 illustrates the synchronization process. Firstly, each sensor stamps the raw data with its local time. DFU receives the data asynchronously from the sensors nodes (here only camera and IDIS are shown) and pushes them in their corresponding data buffer. Each received cycle is stamped by the receiving time at the DFU.

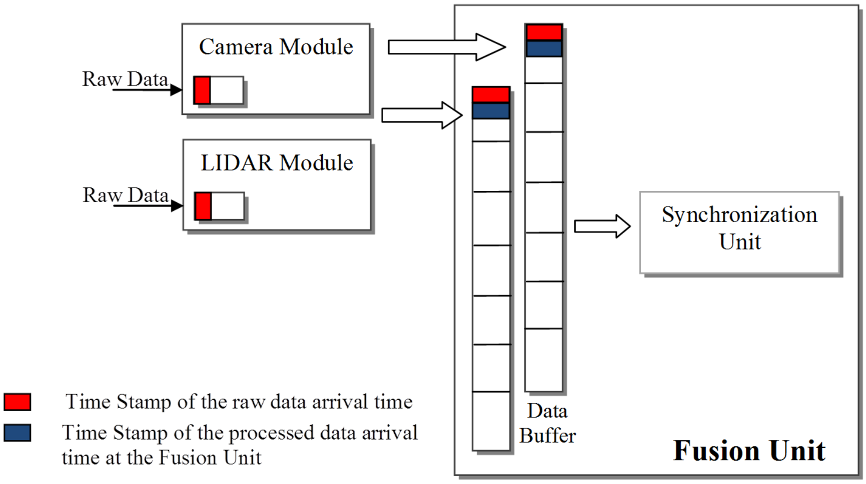
Camera Module
Raw Data
LIDAR Module
Raw Data
Synchronization Unit
Data Buffer
Fusion Unit
🟥 Time Stamp of the raw data arrival time
🟦 Time Stamp of the processed data arrival time at the Fusion Unit

Figure 5.1.6: Time Stamping Process

The task of the synchronization unit is to check the time difference between the received time of the raw data and the arrival time at the DFU. This time difference is an indication to the processing time, which the sensor module consumes in processing its raw data, and the time delay in the network. In case that the time difference exceeds the thresholds, an out of synchronization error will be reported to the DFU to re-initiate the synchronization process. As mentioned earlier, the interaction between server and client nodes is achieved through the packets of data containing a message number in its header. Table 5.1 shows the messages identifiers which are used in the interaction process.

Message Identifier	Data Packet	Description
2030	Synchronization reply	The data packet contains the actual time of the Client side to calculate the delay time in the server-client connection
2031	Receiving Mode	The clients send its data to the DFU.
2032	Synchronization request	Reads CPU time of the server and sets the local CPU time of the client with that of the server.
2033	Sending Mode	Indication that the server is ready to receive the client data.

Table 5.1: Messages of The Client and Server Listening Threads

5.1.3 Spatial Alignment

Spatial alignment is a method of referencing different data from different sensors into a unified scheme. IDIS measures the position in X, Y, Z where $Z = 0$. The vertical measurement angle of 3° is not provided by the IDIS measurements and it cannot be neglected as it would produce errors in the alignment and also in the association process. So, this angle can be estimated using lane visionary recognition to get the tilted angle of the ground plane, or also from the Altitude values of the leading road which are provided by the digital map.

Camera gets projected coordinates of the object (Projection of X, Y, Z of real world into x, y image plane). Many Researchers adapted the approach of extracting the X, Y, Z dimensions from the image by using either set of stereo cameras or more than two cameras [Zha00]. Each Camera captures the scene's image from a different angle and then the images are correlated to each other to get a 3D geometry of the observed objects. It should be noted that all the cameras should be calibrated and fixed at known distances. Maehlisch et al.[MDLR07a, MDLR07b] described a method to find the intrinsic parameters of the IDIS by capturing the reflection positions of the IDIS beams on a distanced wall in the image domain. Afterwards, several iterations should be performed to fit IDIS beams with the image plane.

As stated earlier, IDIS measures the distance of the detected objects relative to the host vehicle. Since, the main aim is to correlate the image and IDIS data, IDIS data can easily be projected to the image plane by using the projection matrix of the camera plane [SRF$^+$08]. To form such a matrix the pinhole camera model [FP02] is used. The camera projection matrix is derived from the intrinsic and extrinsic parameters of the camera. It is often represented by a series of transformations, e.g., a matrix of camera intrinsic parameters, a 3x3 rotation matrix, and a translation vector. Figure 5.1.7 shows the relation of the real world and the camera coordinates, where external parameters represent the rotational as well as the transformational matrices between the two coordinates.

The conversion from the world coordinates system to the camera coordinates system can be performed with the following equation.

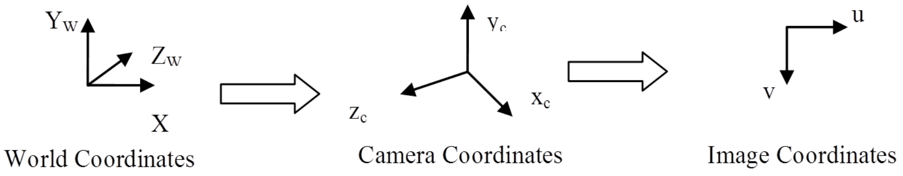

Figure 5.1.7: Coordinates Systems

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = M \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{5.1.1}$$

$$M = SPR_zR_yR_xT \tag{5.1.2}$$

where $\{x_c, y_c, z_c\}$ are the camera coordinates and $\{X_w, Y_w, Z_w\}$ denote the world coordinates. In equation (5.1.2) , M is the product of scaling (S), perspective (P), rotation $\{R_z, R_y, R_x\}$, and transformation (T) matrices.

The camera coordinates should not be confused with the image coordinates, as the camera intrinsic parameters describe the transformation from the camera coordinates into the image plane. Image coordinates can be obtained from camera coordinates by using the next equation.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \tag{5.1.3}$$

$$A = \begin{bmatrix} fk_u & 0 & u_o \\ 0 & -fk_v & v_o \\ 0 & 0 & 1 \end{bmatrix} \tag{5.1.4}$$

where $\{u, v\}$ are the image coordinates, A is the intrinsic parameters matrix, the (u_o, v_o) coordinates is the center point of the image, $\{k_u, k_v\}$ represent the scale factors of the camera, and f is the focal length.

The Matrix M in equation (5.1.2) should be modified to match the installation setting between the IDIS and camera as follows.

$$R_z = \begin{bmatrix} \cos\phi_z & -\sin\phi_z & 0 & 0 \\ \sin\phi_z & \cos\phi_z & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.1.5}$$

where ϕ_z is the rotation angle around the z-axis and R_z is the correspondence rotation matrix.

Similarly, the rotation around y-axis and x-axis are given by the equations:

$$R_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi_x & -\sin\phi_x & 0 \\ 0 & \sin\phi_x & \cos\phi_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.1.6}$$

$$R_y = \begin{bmatrix} \cos\phi_y & 0 & \sin\phi_y & 0 \\ 0 & 1 & 0 & 1 \\ -\sin\phi_y & 0 & \cos\phi_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.1.7}$$

where ϕ_x is the rotation around x-axis with $180°+$ tilted angle of the ground plane and ϕ_y is the rotation around y-axis with 180°.

In addition, the translation matrix should be included. This is due to the reason that the IDIS is mounted on the head of the vehicle, whereas the Camera is beyond it with 1 m.

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.1.8}$$

The last column in the above transition matrix (T) represents the transition of the camera view point in x, y and z arranged from the first row. After constructing the M matrix, IDIS data can be projected in image plan using the equation:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = AM \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{5.1.9}$$

5.1.4 Data Association

In a data fusion problem, we commence by defining a state of nature which we are interested in. Such a state may be a description of the spatial location of an object, its identity in terms of attributes, a complex dynamic state, or simply a single numeric quantity. The objective is to infer the true state based on, often incomplete and sometimes conflicting, information obtained from a variety of sources and to optimize the way in which such information will be handled. Data association addresses the problem of assigning many observation pairs (if any) to each other, indicating that the observations describe the same entity.

In general, the sensor unit consists of two main parts, namely the sensor element which measures the physical entity and the signal processing unit which amplifies as well as records the measured entities. Additionally an embedded Interface System (IS) is needed to provide the reading to the outside world as shown in the Fig. 5.1.8.

Sensor can be regarded as a system whose output can be described as a function of the following terms:

Entity-Name (E): This includes the name of the physical property which was measured by the sensor and the units in which it is measured. Often, the units are defined implicitly in the way the system processes the measurement value.

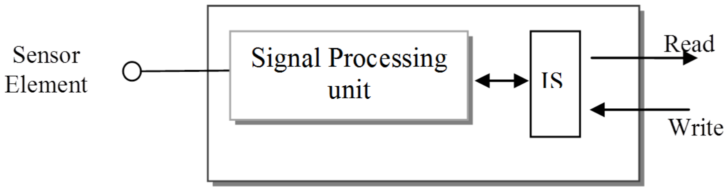

Figure 5.1.8: Overview of Sensors Elements

Time Instant (t): This is the time when the physical property was measured. In real-time systems, the time of a measurement is often as important as the value itself.

Measurement (y): This is the value of the physical property measured by the sensor element. The physical property may have more than one dimension; therefore it will represented as a vector $\mathbf{y}$.

Uncertainty (Δy): This is a generic term and includes many different types of errors in $\mathbf{y}$, including the measurement errors, calibration errors, loading errors, and environmental errors. Some of these errors are defined a priori in the sensors data sheet and others may be calculated internally (if the sensor is capable of validating its own measurements).

Variable of Interest (H): This variable should be distinguished from the measurement $\mathbf{y}$, as it is not directly observed rather must be inferred from the measurements. For example, this variable can represent identity.

Symbolically, the task of inferring H is interpreted as estimating the posteriori probability, $P(H = \theta|y, E)$, where θ represents the true value of the variable of interest H and $y = (y_{t_1}, y_{t_2}, \ldots, y_{t_N})^T$ denotes the vector of $\mathbf{N}$ sensor measurements. The posteriori probability density function (PDF) $P(H = \theta|y, E)$ can also be extended to include the uncertainty Δy as:

$$P(H|y, E) = \frac{P(y|H, E)P(H|\theta, E)}{P(y|E)} \tag{5.1.10}$$

where:

A-Priori pdf $P(H|E)$: This is a continuous probability density function which describes our a-priori beliefs about H. This pdf is related to the a-priori information through the tracking management by imposing certain weight on the hypothesis, indicating that the old object from the old cycle still exists within certain range.

Likelihood function $P(y|H, E)$: This is a continuous function which describes how the raw sensor measurements $\mathbf{y}$ depend on the true value H.

Evidence $P(y|E)$: Which represents the sensor reliability. This is a discrete probability distribution which specifies the a priori reliability of the sensor. In the simplest model, there are two states: $\{\lambda_0, \lambda_1\}$. Here, λ_0 denotes fault-free operation and λ_1 denotes faulty operation, where ordinarily $P(y|E = \lambda_0) = 1$. The above equation describes the probabilistic belief of the measurement for a given sensor. Now we consider the problem of correlating multiple measurements from multiple sensors.

In the work described by Hofmann [HRD01] on Radar-Vision Data fusion, the hypothesis region is constructed for each detected object. The region indicates the area where the object should exist arbitrary at any position in the next measurement cycle. If these hypotheses are not updated for a number of sequential cycles, hypotheses should be removed from the system and regarded as false measurements. In general, constructing hypothesis regions are dependent on the range of the sensor itself (as far away objects may be confused with the noise of the sensor). Hence a confidence factor for each measurement should be included in the assigning process. By assuming a set of n sensors measurements $[y_{11}, y_{12}, y_{13} \dots y_{1n}]$ from sensor A and $[y_{21}, y_{22}, y_{23} \dots y_{2n}]$ from sensor B, the state model can be used to relate the measurements to state vector $\mathbf{X}$, whose values are recorded in discrete times as follows.

$$y = M \times X \tag{5.1.11}$$

where M is the observation matrix and X is the states vector $X = [x_1, x_2, ...x_n]$.

Here, each x_b (b = 1, 2, ..., n) signifies an attribute or distinct object type. Each attribute or object x_b is characterized by a set of observable parameters and so the observation model consists of a parameter set $M = \{m_1, m_2 \dots m_m\}$ which relates each observation y_b to the state elements of $\mathbf{X}$. According to Bayesian theory, the chain rule of conditional probability $P(H_i|E)$ can be described as:

$$P(H_i|E) = \frac{P(E|H_i)P(H_i)}{\sum_i P(E|H_i)P(H_i)} \tag{5.1.12}$$

where $P(Hi|E)$ is the posteriori probability of hypothesis H_i being true given the evidence E, $P(H_i)$ is the priori probability of hypothesis H_i being true, and $P(E|H_i)$ is the probability of observing evidence E given that H_i being true.

The above mentioned equation describes the probability of a certain hypothesis H_i in case that a certain event E is already occurred. Also it describes the methodology behind integrating different sensors data by combining the probabilities of different sensors. Thus, the measurements from the camera and IDIS sensors can be described in a normal probability function, providing a mathematical relation between the positions of the observed data related to the spatial plane.

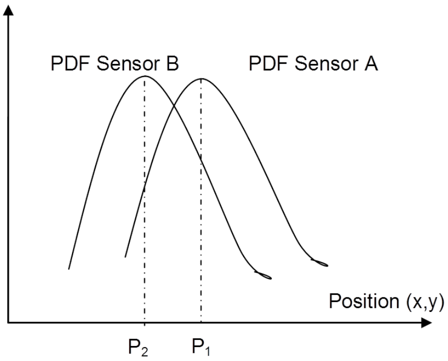

Figure 5.1.9: Spatial Probability of Two Sensors

Figure 5.1.9 shows two PDF functions of one entity from the two sensors. The graph describes the normal probability distribution of the position of the object in the spatial plane. Sensor

A observed an object at position P_1 ($\mathrm{PDF_A}$) and assigns it a normal Gaussian distribution expressing how confident the detection is. Similarly, for sensor B the detected object is at position P_2 ($\mathrm{PDF_B}$). These PDFs provide a way to generate hypotheses for each sensor in spatial plan to answer the question, at what real position is the detected Object? To combine the probability of the two sensors, the following equation is used.

$$P(H|y_1, y_2, E) = P(y_2|H, E)P(H|y_1, E) \tag{5.1.13}$$

The equation updates the hypotheses based on combining the two probabilities and describes the possibility that the observed object can be at position (x, y) by searching the position (x, y), which is the maximum value of the calculated PDF. The main problem of this equation is that the position (x, y), which has the maximum combined probability, can attain any value between (x_1, y_1) and (x_2, y_2) where the real object has been detected from sensor A and sensor B, respectively. The above equation can be reformulated in terms of association metrics. The association metrics aim to quantify the similarities between observation-pairs, based on optimization or likelihood criteria. Such a metric provides means to quantify whether observations are similar or dissimilar based on constructing suitable likelihood criteria. These likelihood criteria can be derived by comparing each observation position from one sensor to the others and correlating the minimum distanced observations to each other. This approach is known as NN (Nearest-Neighbor) and has been used widely in many applications. The NN addresses the problem of searching two observations, which minimize the following equation.

$$Diff = \sqrt{(y_i - y_j)^2} \tag{5.1.14}$$

where y_i is the observation from sensor i and y_j is observation from sensor j.

The NN is trivial to be implemented and the error of such primitive assigning may be reduced by getting the minimum distanced observation with the same classification (if applicable). In [FHL05], the authors presented an approach to minimize the error in the fused probability based on a weighting factor given by

$$W_i = \frac{Conf_b}{Dist_i} \tag{5.1.15}$$

where$Dist_i$ is the distance between two spatial measurements and $Conf_b$ is the measurement reliability

Another association metrics is to use the identity of the detected object, e.g., from camera or IDIS. In [HM04] the authors described that the winner identity will have maximum likelihood probability obtained by:

$$Max(P(y_j \cap y_2 \cap \ldots \cap y_n|H_i)) \tag{5.1.16}$$

where y_j indicates measurement of sensor j, which gives score to the identity H_i.

The equation indicates that the winning identity will have the maximum combined probability. This simple approach combines the identity in an optimal manner, but as indicated in equation (5.1.10), the probability assigned to any hypothesis is dependent on priori information, which means that in case of poor input data, the output data will give high probability to any repeated false classification. So, in order to avoid this problem, association algorithms are performed in two stages listed below.

1. **Current Cycle Association** (CCA): It integrates the sensors data from the current cycle, neglecting priori information from tracking algorithm.

2. **Data Base Association** (DBA): CC list will be merged with the DB list, based on the maximum likelihood probability based on Dempsters Shafer method [WSSY02].

5.1.4.1 Current Cycle Association (CCA)

As mentioned previously, sensors operate in an asynchronous manner, which means that the DFU can receive data from a number of sensors at a time. The received data is saved for processing. By assuming that all the sensors data are available then association will be performed in a cascaded manner, which means that two sensors data sets (for example, from IDIS and camera) will be associated together as a first step and then the result will be used in associating the next data set and so on. Association Metrics will be a combination of position and identity for every pair objects from two sensors. The equation (5.1.16) can be reformulated using the weighting methodology. The resulting validation process based on the weighted approach can be expressed as

$$\begin{bmatrix} W_1 \\ W_2 \\ W_3 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \begin{bmatrix} C_{11} + C_{12} + C_{13} \\ C_{21} + C_{22} + C_{23} \\ C_{31} + C_{32} + C_{33} \end{bmatrix} \tag{5.1.17}$$

$$W = R \times C \tag{5.1.18}$$

where W is the vector of weight identity factor, R is the certainty matrix of the classification, and C_{ij} represents the classification of sensor i to the identity number j.

The equations represent identity data fusion of 3 categories for 3 different sensors. The weights W_i (i = 1, 2, 3) are normalized according to

$$W_1 = W_1 / \sum_{i=1}^{3} W_i \tag{5.1.19}$$

$$W_2 = W_2 / \sum_{i=1}^{3} W_i \tag{5.1.20}$$

$$W_3 = W_3 / \sum_{i=1}^{3} W_i \tag{5.1.21}$$

For example, if IDIS classifies an object as “category 1” C_{11} will be equal to 1, however C_{12} and C_{13} will equal to zero. The confidence of the classification will be included in the factor R_{11}. Furthermore, the confidence factor is a normalized Gaussian distributed factor. For the case of the data received from the C2C module, the confidence factor for category 2 is set to be equal to 1 to indicate that the object, which keeps sending its coordinates, is a vehicle. Such an action gives the highest weight to compensate the false classification from the camera and the IDIS. Weight identity factors of each category will be calculated from equation (5.1.17) and the maximum weight will be the assigned to the category of the requested object.

5.1.4.2 Data Base Association (DBA)

To use priori information from the old cycle, history about the detected objects should be saved to be used in association process. Merging Current Cycle (CC) with Data Base (DB) should take into consideration two basic factors:

- Position.
- Identity.

The same two factors are used in CCA and will be used here as well. But to optimize the association process, priori information from the DB will be used. The association process will be performed in the following steps:

1. Estimating the objects position by using Kalman filter.
2. Associating CC data with DB.
3. Updating Kalman Filter parameters.

Dempster Shafers method [WSSY02] provides a mean to construct a hypothesis based on several evidences to associate the CC data with that of the DB. The method is a generalization of the Bayesian theory that allows a general level of uncertainty. It utilizes uncertainty intervals to determine the likelihood of hypotheses based on multiple evidences. The basic idea behind this method is to assign measures of belief to combinations of hypotheses instead of assigning evidence to a set of mutually exclusive hypotheses.

The method determines that if there exist any set of n exclusive and exhaustive sets of propositions given by

$$\psi = [A^1, A^2, \ldots, A^n] \tag{5.1.22}$$

According to Dempster Shafer's method several random sets can be constructed from the above sets, such as

$$\Theta = [A^1 \cap A^2 \cap A^3] Or [A^2 \cap A^3] \tag{5.1.23}$$

These sets describe the hypotheses based on any combination of evidences. A mass function can also be constructed from the above set to provide a measure of belief. The developed mass function follows the general probability rules, e.g.,

$$m(\Theta) \leq 1 \tag{5.1.24}$$

$$\sum m(\Theta) = 1 \tag{5.1.25}$$

Based on the work described by Dang [Dan03], a realization of Dempster Shafer's theory can be applied in Data Base Association problem as follows.

1. Prediction of the found objects' states using Kalman filter
2. Calculating the mass function between the prediction vector X and the measurement vector Y

3. Calculating the belief Matrix between the track vector X and the measurement vector Y
4. Decision is made, if local maxima from m_i is consistent with local maxima from m_j, otherwise decision will be ignored

To illustrate the above sequence, Kalman filter is used to update the estimation of a tracked object's position (for more details about Kalman filter, refer to section 5.2). This estimation is necessary to reduce the error that may occur while correlating the DB objects with that detected from the CC. The update procedure is performed using following equations.

$$x_k = [X_i, Y_i] \tag{5.1.26}$$

$$x_{k+1}^- = [X_i^-, Y_i^-] \tag{5.1.27}$$

$$y_{k+1} = [X_m, Y_m] \tag{5.1.28}$$

$$x_{k+1} = x_{k+1}^- + G_{k+1}(y_{k+1} - Cx_{k+1}^-) \tag{5.1.29}$$

where (X_i, Y_i) is the position of object i in x, y coordinates and (X_m,Y_m) is the new position after the association process.

Kalman Filter is initialized for each new object of the DB. The current states of the filter X_i and Y_i describe the position of the object in the image plane. The filter assumes that the states do not change dramatically, so a prediction of the position in the next cycle will be made at position X_{i^-} and Y_i^-. The (-) sign indicates the priori belief of the objects position. Thereafter, the filter is updated by the measurements y_{k+1}, which represents the actual position of the tracked filter from CC, but the question arises that, which object of CC is corresponding to that of the DB.

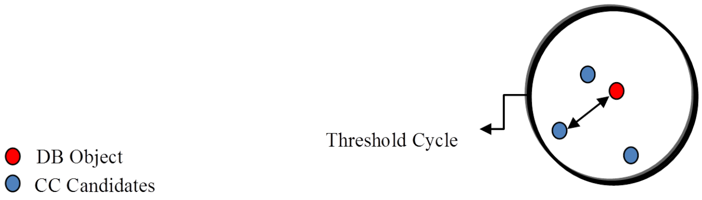

Figure 5.1.10: Candidates Selection From Current Cycle

Figure 5.1.10 shows a DB object surrounded by three detected objects from the CC, which represents the candidates that lie within the threshold cycle (the hypothesis region). The process of choosing the winning candidate is now a matter of constructing the mass function (such as likelihood or matching criteria) and determining the most appropriate object that fulfills the maximum matching score as follows.

$$m(\Theta_{i1}) = \frac{1}{K * Distance} \tag{5.1.30}$$

$$m(\Theta_{i1Norm}) = m(\Theta_{i1}) / \sum_{i=1}^{n} m(\Theta_{i1}) \tag{5.1.31}$$

$$m(\Theta_{i2}) = \begin{cases} 0.5 & \text{if they have the same Category;} \\ 0 & \text{if they have different Category.} \end{cases} \tag{5.1.32}$$

$$m(\Theta_i) = m(\Theta_{i1Norm}) \times m(\Theta_{i2}) \tag{5.1.33}$$

where $m(\Theta_{i1})$ is position mass function of Candidate i, $m(\Theta_{i1Norm})$ denotes the normalized position mass function of candidate i, $m(\Theta_{i2})$ is the identity mass function of candidate i, $m(\Theta_i)$ represents the combined position-identity mass function of candidate i, and K is a weighting constant.

Here, $m(\Theta_{i1})$ describes the position mass function between candidate i and DB object (1). The position mass function is defined as the reciprocal of the difference between the two compared objects. Normalization is necessary to avoid over floating while processing and also to enable the combination when using the identity mass function.

Identity mass function is implemented as a Boolean state. If one of the candidates has the same category of the DB object, it will have the mass function of value of 0.5; otherwise the mass function will be zero. The combined mass function contains the position and identity proposition. The candidate, who has the maximum mass function will be assigned with DB object.

5.2 Lane Fusion

Detecting the Lane lines from the image plane has been introduced in section 4.1.1. Furthermore, digital maps can provide information about the road boundary and also information about the number of lane lines [Ess08]. The main aim of this section is to integrate the lane location information obtained from the vision system and the one estimated from the digital map to get a precise information of the driver path. To achieve this goal, a road boundary from the digital maps has to be reconstructed in the image plane using the projection matrices, which were introduced in the last section. It should be noted that certain requirements should be met to assure that the reconstructed scene matches the one obtained from the camera.

1. **Precise GPS Position and Course Angle of the Vehicle**
   This information is essential while transforming the road boundary from the digital maps to the vehicle coordinates. In addition, an enhancement of the position and orientation of the vehicle has to be performed by integrating GPS with on-board vehicle sensor to minimize the lateral offset (error) in the GPS.
2. **Digital Maps Road Boundary Projection**
   Lane plane has to be estimated at a pre-defined distance ahead of the vehicle. Practically, the lateral offset of the GPS is the offset between the lane plane in the image and the one reconstructed from the digital map. So, it is important to project the digital maps road boundary to the image plane, taking into consideration the orientation angle of the vehicle, as the reconstructed scene should rotate respectively to get a precise simulation of the road from the camera's point of view.

5.2.1 Vehicle Position Enhancement using the Ego Dynamics Data

The primary objective of any navigation system is to combine the navigational data with the route information in order to relate the system position to the surrounding environment. The work in [BTMM04] summarizes the position extraction of the host vehicle into three basic approaches, given below

- **Relative localization**: In this method an in-board vehicle sensors (such as speed, yaw rate, and acceleration sensors) are used to dynamically update the position and the attitude angle of the vehicle.
- **Absolute localization**: This scheme uses GPS to provide the absolute position.
- **Hybrid localization**: It combines the above two approaches together to minimize the localization error.

Estimating absolute accurate position is one of the main goals of many mobile systems (such in robotics) as knowing the position relative to the environment leads to a system's accurate functionality. In spite of the recent progress and advancement in the GPS design, there still exists an error margin of 20 cm that may extend in certain situations to 2 m. Furthermore, GPS signal may not be available in some urban areas (such as tunnels) due to signal shielding or signal unavailability. As a remedy to this problem, many articles have been published in the literature suggesting the inclusion of the on-board vehicle sensors in hybrid form to correct the calculated position [SGKN10, RG04, PDEP01, BD98].

In order to provide the position of the vehicle with a decent accuracy, information from the GPS, on-board vehicle dynamics sensors, and digital maps are integrated to compensate the positioning errors which arise when using the GPS alone. It will be needed to interpolate the positions between the GPS points, since the GPS operates at 1 Hz. The interpolated positions are calculated in terms of longitude and latitude coordinates. In the work by Obradovic [OLS06], a formula is introduced to calculate the new Geodic coordinates by updating the longitudinal and latitudinal position given by

$$\begin{bmatrix} long_{k+1} \\ lat_{k+1} \end{bmatrix} = \begin{bmatrix} long_{k} \\ lat_{k} \end{bmatrix} + \Psi \left\{ \frac{\sin(\Delta\alpha/2)\,(1+Dist)}{\Delta\beta/2} \begin{bmatrix} \cos\alpha_k + \Delta\alpha/2 \\ sin\alpha_k + \Delta\beta/2 \end{bmatrix} \right\} \tag{5.2.1}$$

where α_k stands for the orientation angle, Ψ represents the transformation from the planar to the WGS84 coordinate system, and β denotes the attitude angle of the vehicle.

The formula is an additive process, in which changes in the vehicle's dynamics should be added to the current position. This additive process is known as "dead reckoning" [KHW02], which can be defined as a process in which the system's states (mainly, the position and speed) are updated by the distance between two measures and the change of orientation.

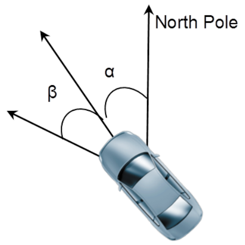

Figure 5.2.1: Difference Between the Orientation and Attitude Angle

Figure 5.2.1 illustrates the difference between the orientation angle α and the attitude angle β of the vehicle. The orientation angle is the angle between the North Pole and the longitudinal axis of the vehicle, while the attitude angle is the angle by which the vehicle tends to move relative to its longitudinal axis. So, change in the attitude angle of the vehicle can be regarded as the change in the orientation angle itself.

Equation (5.2.1) can be re-calculated based on Haversine formula. This formula calculates the distance between two points on the sphere given their geographic coordinates (Latitude and Longitude). Sinnott [Sin84] devised a way to ease the computations in spherical coordinates for the triangles using Haversine formula.

By relating the angle (Θ) with the radius of the earth (R), the distance ($Dist = c$) can be extracted from the formula:

$$Dist = \cos^{-1}\left(\sin(Lat_1) \times \sin(Lat_2) + \cos(Lat_1) \times \cos(Lat_2) \times \cos(\Delta Long)\right) \times R \tag{5.2.2}$$

where the latitudes and longitudes are transformed into radians. In fact, when Sinnott devised the Haversine formula, computational precision was limited. Nowadays, most modern computers

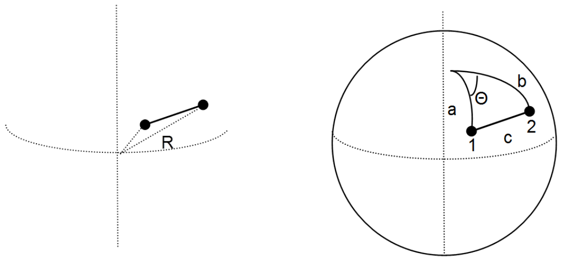

Figure 5.2.2: Calculating Distance in Geographic Coordinates

use IEEE 754 64-bit floating-point numbers, which provide 15 significant figures of precision. With this precision, the simple spherical law of cosines formula gives well-conditioned results down to distances as small as around 1 meter. Furthermore, Sinnott rewrites the equation (5.2.1) as:

$$Lat_{k+1} = \sin^{-1}\left(\sin(Lat_k) \times \cos(\frac{Dist}{R}) + \cos(Lat_k) \times \sin(\frac{Dist}{R}) \times \cos(CoG)\right) \tag{5.2.3}$$

$$Lon_{k+1} = Lon_k + \tan^{-1}\left(\frac{\sin(CoG) \times \sin(\frac{Dist}{R}) \times \cos(Lat_k)}{\cos(\frac{Dist}{R}) - \sin(Lat_k) \times \sin(Lat_{k+1})}\right) \tag{5.2.4}$$

According to the above equations, the new Course over Ground angle (CoG) is needed to estimate the new longitude and latitude values. If change in attitude angle (β) of the vehicle is known, CoG can be updated as:

$$CoG_{new} = CoG_{old} + \beta \tag{5.2.5}$$

The angle β cannot directly be measured, but it can be estimated from simplified bicycle model. The bicycle model [RG04, Tra07] is only an approximation to model the vehicle dynamics in terms of the main vehicle states. The model describes the vehicle motion in terms of the main states (e.g., attitude angle and yaw rate).

Figure 5.2.3 introduces the bicycle model described in [RG04], which approximates the vehicle movement by using the symmetry approach. Thus, the main equations of motion are given by

$$ma_y = F_{sv} cos\delta + F_{sh} \tag{5.2.6}$$

$$J_z \omega^{\cdot\cdot} = F_{SV} cos\delta l_v - F_{SH} l_H \tag{5.2.7}$$

$$F_{SV} \cong C_{\alpha V} \alpha_V \tag{5.2.8}$$

$$F_{SH} \cong C_{\alpha H} \alpha_H \tag{5.2.9}$$

$$\alpha_V \cong \delta - \beta - \frac{l_V}{v_x \omega^{\cdot}} \tag{5.2.10}$$

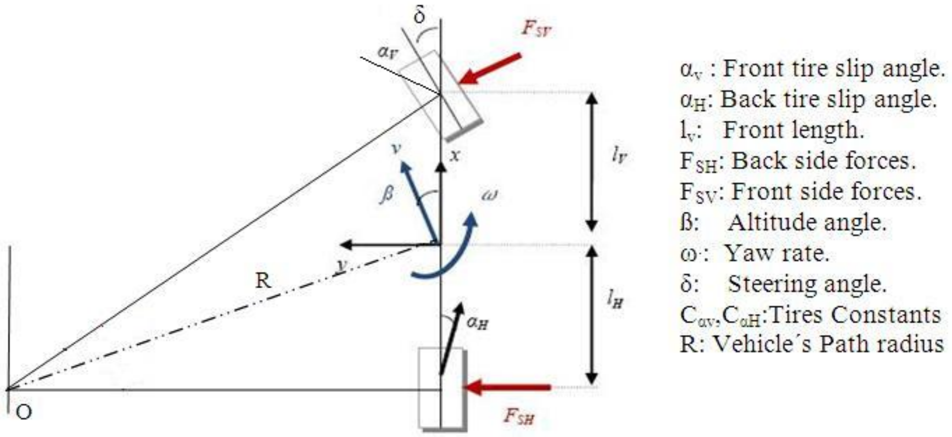
δ
F_{SV}
α_V
l_V
v
x
β
ω
R
y
l_H
α_H
O
F_{SH}
α_v : Front tire slip angle.
α_H: Back tire slip angle.
l_v: Front length.
F_{SH}: Back side forces.
F_{SV}: Front side forces.
ß: Altitude angle.
ω·: Yaw rate.
δ: Steering angle.
$C_{\alpha v}$,$C_{\alpha H}$:Tires Constants
R: Vehicle´s Path radius

Figure 5.2.3: Bicycle Model [Tra07, RG04]

The attitude angle (β) can be estimated by solving these above dynamic equations. The equations can be rewritten in the state space form as:

$$\begin{bmatrix} \beta^{\cdot} \\ \omega^{\cdot\cdot} \end{bmatrix} = \begin{bmatrix} -\frac{C_V+C_H}{mv} & \frac{C_{\alpha H}l_H-C_{\alpha v}l_v}{mv^2}-1 \\ -\frac{C_{\alpha H}l_H}{J_z} & \frac{C_{\alpha v}l_v^2-C_{\alpha H}l_H^2}{J_z v} \end{bmatrix} \begin{bmatrix} \beta \\ \omega^{\cdot} \end{bmatrix} + \begin{bmatrix} \frac{C_{\alpha v}}{mv} \\ \frac{C_{\alpha v}l_V}{J_z} \end{bmatrix} \delta \qquad (5.2.11)$$

Table 5.2 shows the vehicle constants mentioned in the above equations. These constants were obtained from the BMW 530i data sheet.

Parameter	Value
$C_{\alpha v}$	75000 N/rad
$C_{\alpha H}$	150000 N/rad
Jz	28000 Kg. m^2
$L_v = l_H$	1.44 m
m	1555 kg

Table 5.2: Test-Vehicle Parameters

As can be noticed from the above state equations, the model is not linear, since the system's matrix has the velocity (v) in its coefficients. It is assumed that v is constant, while solving the above state space model. The problem could arise e.g., when the value of the velocity leads to an unstable system, which means unmatched results from the stable real model. In order to ensure adequate stability of the proposed model, the characteristics equation is written in the following form as

$$mv^2\left(C_{\alpha H}l_H - C_{\alpha v}l_v\right) + C_{\alpha v}C_{\alpha H}l_v^2 \qquad (5.2.12)$$

According to the table 5.2, the term $C_{\alpha H}l_H - C_{\alpha v}l_v$ is always greater than zero, which produces negative poles and thus a stable model for all values of v. This means that the vehicle will have a stable value of the attitude angle regardless of the driving conditions. This assumption does not consider the fact that the vehicle under extreme driving conditions, such as high speed or slippery road surface may suffer from an unstable rotation and may turn over. But in normal driving situations, the model can be verified to provide a good estimation of the vehicle's states.

Since, $\omega^{\cdot}$ can be measured directly from the vehicle sensor, it can be excluded from the state space equations and handled as an input to the model. Thus the above model will be reduced to the attitude angle as a linear function of (ω, δ, v).

Using the model in the discrete state, $\beta^{\cdot}$ can be replaced by β_{k+1}. This discretization is reasonable under the condition that the interval sampling time T_s is relatively small with respect to the time constant of the proposed model.

$$\beta_{k+1} = -\frac{C_{\alpha V} + C_{\alpha H}}{mv}\beta_k + \left(\frac{C_{\alpha H} l_H - C_{\alpha V} l_V}{mv^2} - 1\right)\omega_k + \frac{C_{\alpha v}}{mv}\delta_k \tag{5.2.13}$$

It should be noted that the proposed model is only an approximation and the error in estimating the attitude angle could be accumulative, unless the attitude angle is corrected by integrating the measurements from the GPS. Kalman Filter is used as an integrating filter between the bicycle model and the measurements from GPS.

5.2.1.1 Kalman Filter

Kalman filter is an estimation filter that is used widely in many applications, ranging from multi-object tracking to data fusion, which also is the case here. The filter consists of two main phases:

1. Prediction phase, in which the new states values are predicted based on the old states.
2. Correction phase, in which Kalman gain and prediction matrices are modified based on the actual measurement values.

Given a dynamic system, its equation of motion can be written as follows

$$x_{k+1} = Ax_k + Bu_k + w_k \tag{5.2.14}$$

$$y_{k+1} = Cy_k + Du_k + v_k$$

where x_k is the state of the system, u_k denotes the inputs vector, y_k represents the measurement vector, A is the dynamic matrix, B and D are the input matrices, C output matrix, and (w_k, v_k) are the process and measurement noise, respectively.

The equations are written in discrete form and describe the states x_{k+1} as a function in previous state observations x_k. The matrices A, B, C, and D are constants for linear systems and represent the system model.

$$x^-_{k+1} = Ax_k + Bu_k + w_k \tag{5.2.15}$$

$$P^-_{k+1} = AP_kA^T + Q \tag{5.2.16}$$

where Q is the process noise covariance and can be derived as a PDF in w_k. The (-) sign indicates that the correspond to the prediction phase of the filter.

$$G_{k+1} = P^-_{k+1}\left(CP^-_{k+1}C^T + R\right)^{-1} \tag{5.2.17}$$

$$x_{k+1} = x_{k+1}^{-} + G_{k+1}\left(y_{k+1} - Cx_{k+1}^{-}\right) \tag{5.2.18}$$

$$P_{k+1} = \left(1 - G_{k+1}C\right)P_{k+1} \tag{5.2.19}$$

where G_{k+1} is the Kalman gain and R denotes the measurement noise covariance.

As shown from the above equations, the prediction is made using the current states as well as the value of the Kalman gain. This gain will be corrected with deviation between the prediction and the actual measurements. Finally, next states x_{k+1} will be corrected by the actual measurements. These two phases of operation make Kalman a self-corrected filter, able to follow the system dynamics.

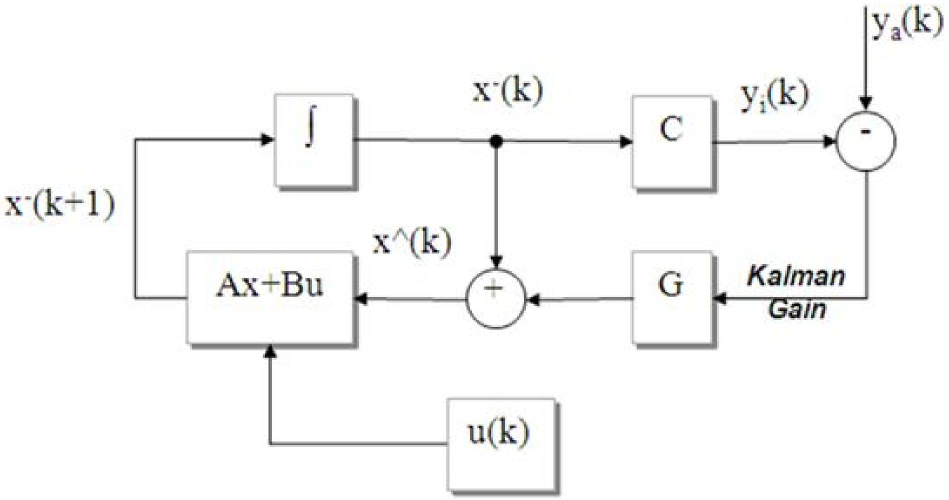

Figure 5.2.4: Graphical Presentation of the Kalman Filter

Figure 5.2.4 shows a graphical representation of the Kalman filter. The variables of the filter are given by:

$$x = [v, \alpha] \tag{5.2.20}$$

$$y_i = [v, \alpha] \tag{5.2.21}$$

$$u = [\Delta v, \beta] \tag{5.2.22}$$

$$y_a = [SoG, CoG] \tag{5.2.23}$$

where y_a is the measurement vector from the GPS, y_i denotes the estimated output vector of the filter, α is the orientation angle of the vehicle measured from the North Pole, Δv and β are the changes in the velocity and the attitude angle, respectively, and β represents the attitude angle.

The Kalman filter may be considered to have three main phases of operation:

1. Extrapolation Phase: The GPS data is not available in this phase, so the measurements are updated using the vehicle dynamics input vector (u) with the help of the prediction matrices given by
$$x_{k+1}^{-} = Ax_k + Bu_k + w_k \tag{5.2.24}$$
$$y_{ik+1} = Cx_k + Du_k \tag{5.2.25}$$
$$P_{k+1}^{-} = AP_kA^T + Q \tag{5.2.26}$$
It should be noted, that no correction is applied in this phase because performing the correction without real measurements will force the states to follow the measurements providing incorrect results.

2. Correction Phase: In this phase, the GPS data is available, however input vector (u) is not. Thus, the filter gain (G) will be corrected as:
$$G_{k+1} = P_{k+1}^{-}\left(CP_{k+1}^{-}C^T + R\right)^{-1} \tag{5.2.27}$$

3. Integration Phase: In this phase, both data are available, so the priori prediction will be used to correct the filter gain by comparing the measurements from the GPS with the prediction made in the last cycle. As a final stage, a prediction matrix is adjusted using the Kalman Gain in order to provide better states estimation for the next cycles.
$$G_{k+1} = P_{k+1}^{-}(CP_{k+1}^{-}C^T + R)^{-1} \tag{5.2.28}$$
$$x_{k+1} = x_{k+1}^{-} + G_{k+1}\left(y_{k+1} - Cx_{k+1}^{-}\right) \tag{5.2.29}$$
$$P_{k+1} = \left(1 - G_{k+1}C\right)P_{k+1} \tag{5.2.30}$$

Figure 5.2.5 illustrates the results when using Kalman filter to interpolate the vehicle orientation angle. The GPS data is shown in the red crosses, while the vehicles states steering angle, yaw rate, and velocity are shown in black, green, and magenta, respectively. The sub graphs are illustrated as follows.

In a), the vehicle's dynamics data is provided to the Kalman filter, where measurements (GPS-data) were absent. The orientation is initialized using the location of the GPS orientation at the test beginning. Therefore, the filter tries to predict the new orientation angle based on the bicycle model as mentioned previously. The GPS points illustrate the deviation between the model output and the real measurements form the GPS.

The sub graph b) shows the test with providing the measurements of the GPS to the Kalman filter. It can be noticed that the deviation is too small, when the measurements are provided to the filter.

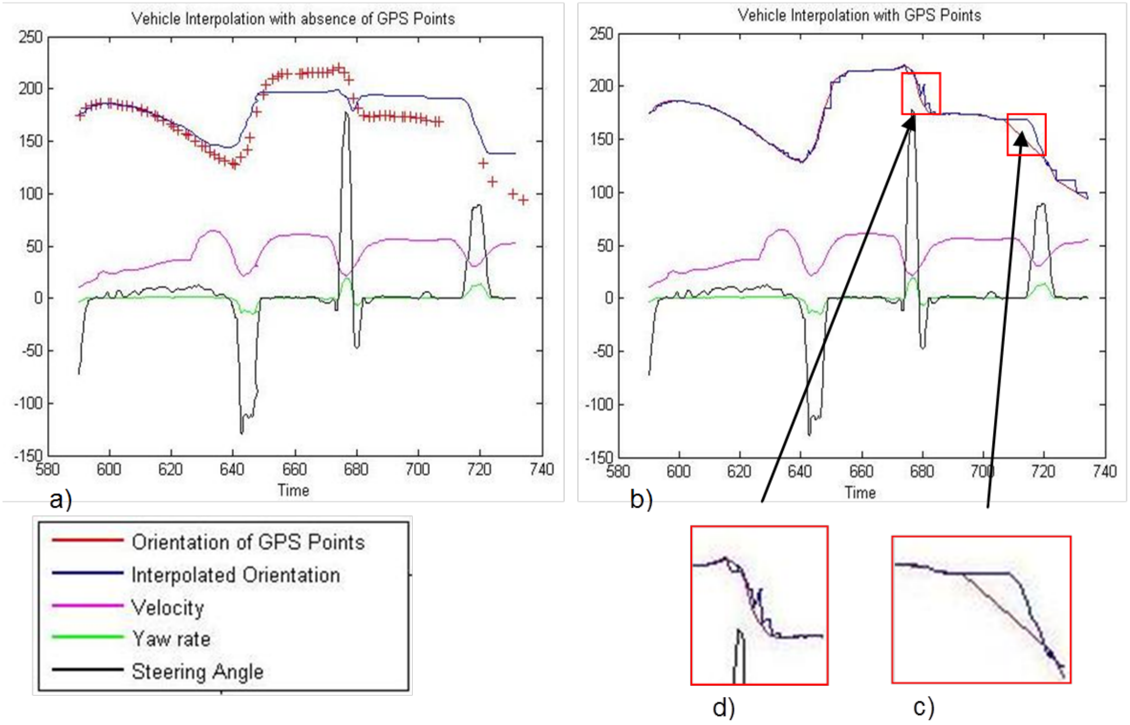

Figure 5.2.5: a) Updating the Orientation Angle in the Absence of GPS Signals. b) Updating the Orientation Angle with Integration of the GPS Signals.

In addition, c) shows a section of the interpolated path, where the GPS data were absent in one curve. The interpolated path is fitted with the vehicle dynamics to match the next value of GPS measurement.

Meanwhile, d) presents the ripples in the interpolated path; this unusual behavior is due to a sudden change of the steering angle at the specified point, as can be noticed from the black line.

As shown in equation (5.2.4), the attitude angle and the measured distance are used in estimating the new geographical position of the vehicle. The tests in the Fig. 5.2.5 are used to map the interpolated GPS position on the digital map.

Figure 5.2.6 shows the interpolated path in the absence of GPS points. It can be shown that the simplified vehicle's model cannot keep a track of the actual position of the vehicle in the absence of GPS signals. Thereafter, the test was repeated to update the vehicle's position by integrating the GPS data; the obtained results are represented in Fig. 5.2.7.

5.2.2 GPS Position Adjustment using Lane Fusion

In the previous section, a GPS position interpolation has been introduced using the vehicle model, by estimating the change in the orientation angle of the host vehicle and the distance traveled between two measurements. However, this approach cannot compensate the (bias) errors in the GPS position, since the GPS measurements are assumed to be accurate and are used as trusted outputs to correct the estimated states within the Kalman filter. In order to compensate the bias

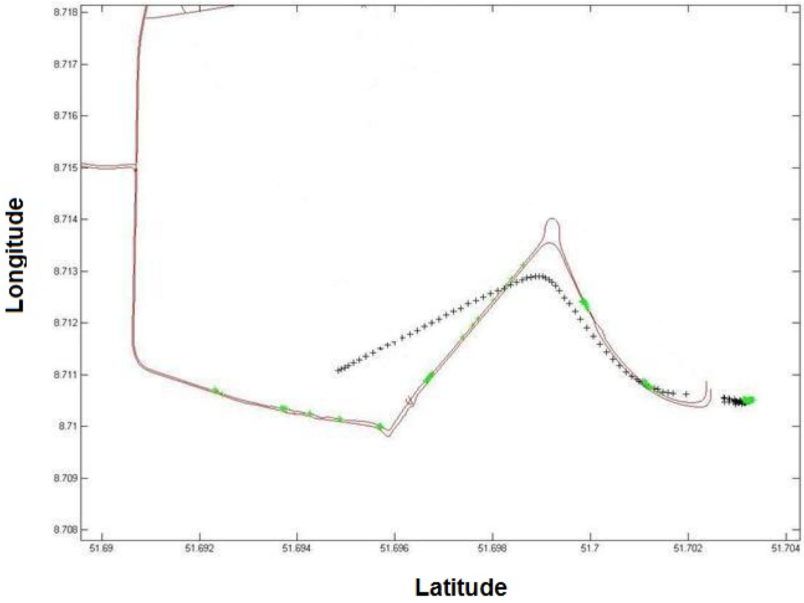

Figure 5.2.6: Updating the Vehicle Position in Absence of GPS Signals

in the GPS positioning, an attempt is made to match lane lines detected from the vision system to those indicated from the road boundary of the digital map, resulting in a better estimate of the correct GPS position and the course angle. If no lane markings are detected, the position and the course will be solely obtained via GPS until lane markings re-appear.

Matching the vehicle position using digital map is discussed in many papers [KHW02, OLS06]. The basic idea behind this approach is to assign the position of the vehicle to the most probable nearby route, to match the perception that the vehicle is always located on the road. The problem still remains that which lane should the vehicle be assigned to?

In this section, a matching algorithm will be described based on the lane fusion concept. This algorithm integrates the lane coordinates obtained from the camera module to that projected by the digital map. Digital maps include sets of control points, which are presented in longitude and latitude values, to describe the road geometry ahead. To illustrate how a digital map can be projected into the image plane, the Sinnott formula, which makes use of the distance between two GPS points, is considered. The formula is used to obtain digital map points in polar coordinates, related to the vehicle as its origin. These points can be projected into the image plane to provide a good measure of the deviation in the road boundary between digital map and that detected from the image. This deviation (offset) is used to correct the GPS position by comparing the actual lane coordinates with that extracted from the digital map as well as while using digital map to find relevant objects based on the road geometry.

The road boundary can be constructed from the digital map points by getting the distance as well as the angle between the vehicle and each control point ahead. The distance ($Dist$) is obtained by using equation (5.2.2), while the angle is given by

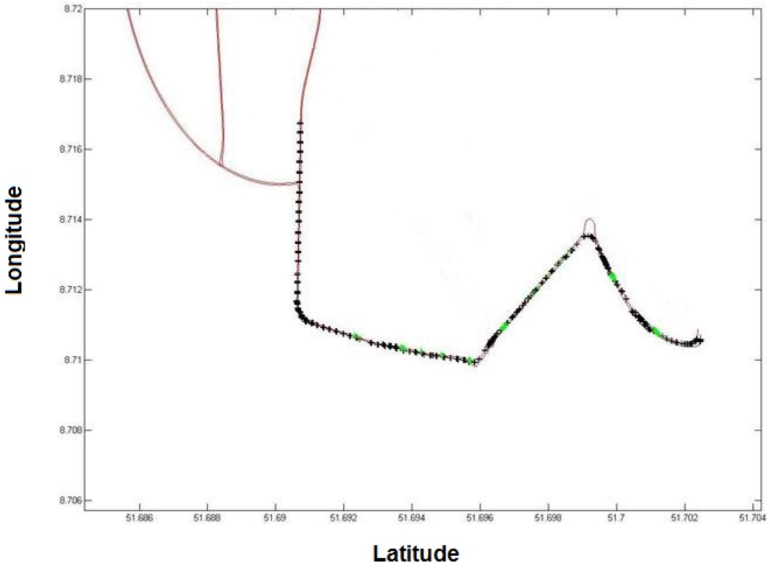

Figure 5.2.7: Updating Vehicle Position with GPS Data

$$\Im = \cos^{-1}\left(\frac{(Lat_2 - Lat_1) \times \pi \times R \times Dist}{180}\right) \tag{5.2.31}$$

where $\Im$ is the angle between the two GPS points. According to the vehicle coordinates, the above equation represents the digital map points in polar coordinates, which can be transformed into vehicle coordinates using

$$\begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix} = Dist \begin{bmatrix} \cos\Im \\ \sin\Im \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \Delta height \end{bmatrix} \tag{5.2.32}$$

where $\Delta height$ is the change in the altitude value between the host vehicle and the corresponding digital map point. Once the digital map points are represented in the real world, the projection matrix can be applied and the road boundary can be reconstructed with respect to the camera's point of view as follows.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = AM \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ \Delta height \end{bmatrix} \tag{5.2.33}$$

Figure 5.2.8 shows a bird eye view of digital map and its corresponding projection in the image plane. The yellow rectangle is the estimation of the vanishing point. It should be noted that the digital map is measured in the host vehicle coordinates, while the digital map is rotated with respect to the heading angle of the vehicle to get the geometry of the road with respect to the driving direction.

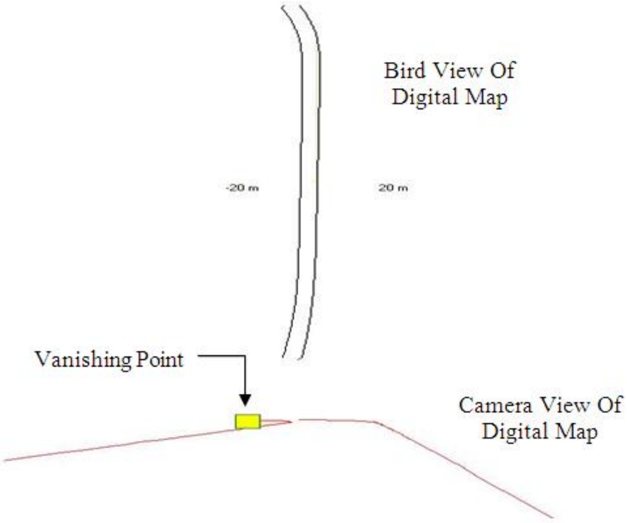

Figure 5.2.8: Projecting Digital Map

Detecting the lane lines in the image is done only in the lower part of the image frames. This part is used in re-modifying the calibration matrix, which was used to match the camera view with the reconstructed virtual one obtained from the digital map.

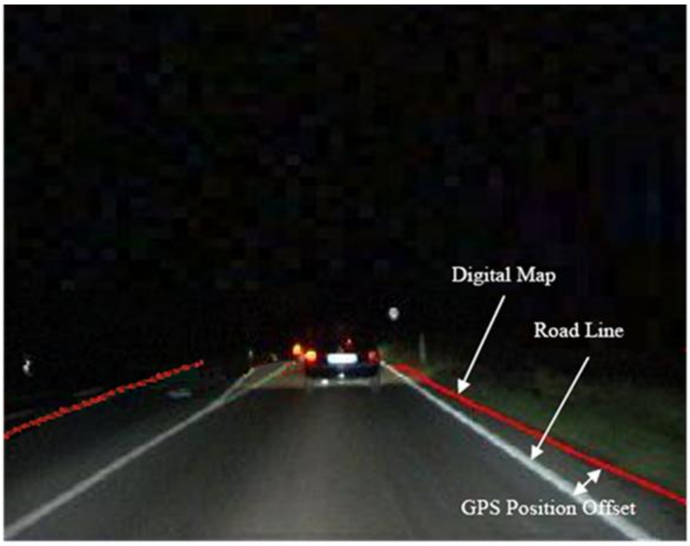

Figure 5.2.9: Position Adjusting using Lane-Lines

Figure 5.2.9 shows the lateral offset in the GPS position. While projecting the boundary road lines obtained from the digital map to the image, it should coincide with road line of the real image. The adjustment algorithm is illustrated in Fig. 5.2.10.

As shown in Fig. 5.2.10, the GPS position data will be used to calculate road points from the digital maps with respect to the vehicle coordinates, and then the points will be projected to the image plane. Lane lines can be represented by four points, since we are only interested in calculating the lateral offset of the GPS position with that of the real lane.

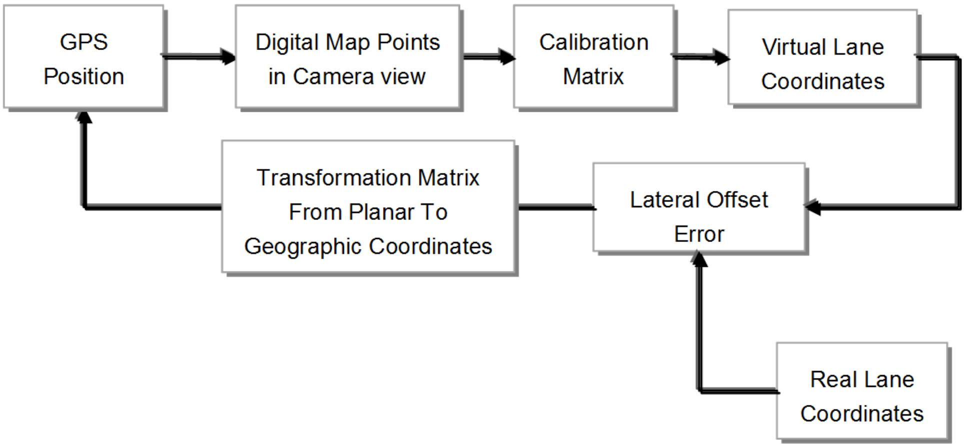

Figure 5.2.10: Position Adjusting Algorithm

The lateral offset should be transformed back to the geographic coordinates in order to be added to the position of the GPS. The main problem is that the projection matrix is not symmetric and the loop ends when the offset is less than 10 cm, or the number of loops exceeds 30. The chosen four points that represent the lane points are assumed to be located at a fixed distance of 10 m, 40 m from the driver. Points from the digital map will be extracted at these locations and offset will be calculated as follows:

$$Error_x = \frac{\sum(X_i - x_i)}{4} \tag{5.2.34}$$

$$Error_y = \frac{\sum(Y_i - y_i)}{4} \tag{5.2.35}$$

where $[X_i, Y_i]$ and $[x_i, y_i]$ are the positions of the lane points in the image and digital maps, respectively.

Finally, the offset error between the points of the trapezoidal, corresponding to the real image and the virtual image (obtained by digital maps), is calculated.

Chapter 6

Situation Analysis and Threat Assessment

Situation analysis is defined as a process of examining the main elements governing a situation and their inter-relations to provide and maintain a state of situation awareness for the decision maker. In the current system, these relationships can be classified as vehicle to surroundings relationship (e.g., orientation and ego-velocity), vehicle to object relationship (e.g., visibility, distance, and relative velocity), and object to object relationship (e.g., inter-distance and inter-relative velocity).

$$\forall r \in S \; \exists \; r_i = f(p_1^k, p_2^k, ..., p_n^k) \tag{6.0.1}$$

where r is the relationship, S is the situation, r_i $(0, 1, 2, ...m)$ are the relations, and p^k are the controlling parameters.

The aim of this Situation Analysis and Threat Assessment Unit (SATAU) is to estimate the parameters $[p_1, p_2, ..., p_n]$ and to establish the various relationships $[r_1, r_2, ..., r_m]$ of the current driving situation from the available input data. The SATAU receives a stream of time-stamped information events about the state of the vehicle (e.g., ego velocity, yaw rate, and orientation), the road type (e.g., highway or city road, number of available lanes, and road curvature), road participants and obstacles (e.g., vehicle and pedestrian), and the actions of the driver (e.g., system activation / deactivation, overwriting the decision of the controller). During a drive, SATAU builds a probability domain P_i by comparing the received information with that stored situations data. If a relevant situation is found in the database, it is assigned to the current situation and its relevant light distribution will be extracted.

The knowledge database used here is constructed from two layers. The first layer which is based on the general parameters and is used to estimate a base light distribution (in this study it is constrained to either a high beam or a dipped beam). In the second layer, which is based on the object specific parameters, the situation is divided into various sub-situations. Each sub-situation represents a distinct object condition (e.g., marking or blinding out). The masking of the output of the second layer (object specific) with the output of the first layer (base) splits the current situation to bright and dark zones, which will be used latter on by the headlight controller to model the light distribution. To develop the database generic situations, a collection of illumination strategies were defined. Each strategy describes a possible base light distribution followed by an intelligent agent dedicated to that situation. Moreover, it was assumed that these illumination strategies can be generated by observing a finite set of key parameters associated with each situation.

6.1 Parameters for Estimating the Base Light Distribution

6.1.1 Ego-Velocity and Steering Angle Rate

Driving at low speed and high steering angle rate characterizes a typical driving behavior on a city road. Therefore, dipped beam is chosen as the base light distribution to avoid unnecessary glare to the pedestrians and cyclists on the road side. This decision is confirmed with the GPS and digital maps data (if available). If the driving speed increased over a specific value, the base light distribution is changed to high beam. The parameter can be computed with the equation 6.1.1. In order to achieve stability in the projected light distribution, a hysteresis was implemented to avoid the rapid switching between the various base light distributions due to the fluctuation in the driving velocity.

$$P_1^B = \begin{cases} 1 & \forall (v \geq v_{\tau 2}) \\ 1 & \forall (v \geq v_{\tau 1}),\ (P_1^{B_{t-1}} = 1) \\ 0 & otherwise \end{cases} \tag{6.1.1}$$

where v is the ego-velocity, $v_{\tau 1}$ and$v_{\tau 2}$ are the lower and upper thresholds, respectively. P^B is the probability of the base light distribution (1 for high beam and 0 for dipped beam), and $P^{B_{t-1}}$ is the base light distribution of the previous cycle.

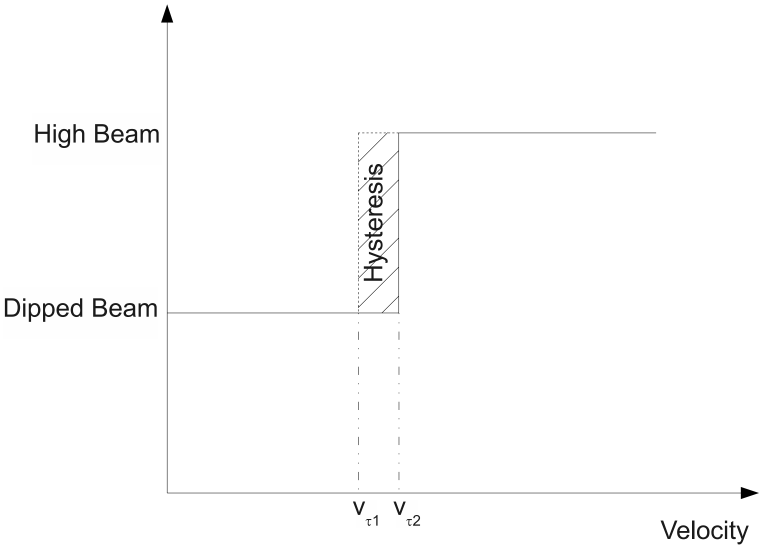

Figure 6.1.1: Activation Velocities and Hysteresis

Figure 6.1.1 shows the activation and switching velocities and the hysteresis limits as well. It should be noticed that the described velocity thresholds can be experimentally adjusted and fine-tuned based on the traffic conditions.

6.1.2 Yaw Rate and Road Curvature

Driving with high beam in a sharp curve can produce hazards due to the appearance of the oncoming vehicle in a short distance (critical zone) which may cause a disability glare to the driver.

$$P_2^B = 1 - \{\min(1, \frac{\psi^{\cdot}}{\tau_{\psi^{\cdot}}})\} \tag{6.1.2}$$

where $\psi^{\cdot}$ is the yaw rate of the host vehicle and $\tau_{\psi^{\cdot}}$ is the allowed threshold.

Car to car (C2C) communication sub-module plays a vital role in such situations by providing the system with the essential information about the vehicles on the curves enabling the system to take reasonable decisions. Therefore, the probability computed in the above equation can be represented as follows:

$$^{C2C}P_2^B = P_2^B(1 - P_{C2C}) \tag{6.1.3}$$

where P_{C2C} is the probability gain obtained from the C2C communication module. If the relative heading angle and the radial distance of oncoming/leading vehicle are smaller than the allowed thresholds, the zone of the vehicle where it is expected to appear will be dipped immediately (i.e., $P_{C2C} = 1$).

If the data of the C2C is not available ($P_{C2C} = 0$) and a high yaw rate is detected, the dipped beam will be used as base light distribution, otherwise the high beam will be recommended.

6.1.3 Light Sources and Average Image Brightness

A combination of a large number of the detected light sources and high image brightness is used as indication of driving in a city road or in a traffic jam, which recommends to use dipped beam as a base light.

6.2 Parameters for Estimating the Object Illumination Strategy

Since, each object should be either marked or blinded out, therefore one probability for each controlling parameter will be estimated, which can be described as follows:

$$S_i = \begin{cases} mark & \forall P_i^O = 1 \\ shade & \forall P_i^O = 0 \end{cases} \tag{6.2.1}$$

where S is the situation decision and P^O is the object's illumination probability.

6.2.1 Object Class

Principally, all the detected vehicles should be blinded out and all non-vehicle objects which are found on the driving road will be marked.

$$P_1^O = \begin{cases} 0 & \forall \text{vehicles} \\ 1 & \forall \text{non-vehicles} \end{cases} \tag{6.2.2}$$

6.2.2 Object Perceptibility Probability

The mean gray value of the object is used to estimate its visibility grade. Non self-illuminating objects have always small perceptibility probability except the objects with highly reflecting surfaces.

$$P_2^O = 1 - \min(\frac{G_{avg}}{\tau_{grey}}, 1) \tag{6.2.3}$$

where G_{avg} is the average gray value of the object and τ_{grey} is the minimum gray value to see non self-illuminating object.

In order to avoid unnecessary glare to the road users, the threshold τ_{grey} is kept high enough to filter most of traffic signs, road reflectors, and pedestrians wearing reflecting clothes. The average value is computed using the method proposed in section 4.1.3.2.

6.2.3 Time to The Closest Point of Approach

A crucial parameter in many threat evaluation techniques is the Closest Point of Approach (CPA). In this method, for stationary objects, threatening targets are prioritized based upon which ones will reach their CPA first. Since, we are dealing with dynamic as well as stationary objects another term called Time-to-CPA (TCPA) or Time To Collision (TTC) is introduced. It is the time taken to reach CPA of an object. For simplification, it can be assumed that while the computation of the CPA/TCPA parameters the object velocity will remain constant during the sampling time. The TCPA can be computed with the following equation.

$$TCPA = \begin{cases} \frac{d}{v_{ego} - v_{obj}} & \text{for } (v_{ego} - v_{obj}) > 0 \\ \infty & \text{otherwise} \end{cases} \tag{6.2.4}$$

where d is the object distance from the host vehicle, v_{ego} is the ego velocity, and v_{obj} is the absolute velocity of the object. It should be noted that, the above equation is valid as long as $v_{ego} - v_{obj} > 0$ where a collision is possible. However, for zero and negative values, the $TCPA$ gets a very large number.

When the driver wants to change the lane and turns on the indicator, a time offset $T_{indicator}$ will be subtracted from the $TCPA$ when the host vehicle will drive in the direction of the object lane, giving the driver more visibility distance to decide if it makes sense to change the lane or not. Otherwise the time offset will be added TCPA when the host vehicle drives away from the object's lane.

$$TCPA = \frac{d}{v_{ego} - v_{obj}} \pm T_{indicator} \tag{6.2.5}$$

Object position on the road produces different threats values, for instance objects on the ego driving lane are more important than those which are long way off the road sides. Therefore, another parameter T_{lane} is considered to modify the estimation of TCPA in equation (6.2.5), where $0 \leq T_{lane} \leq 1$. The total TCPA can be calculated using the next equation.

$$TCPA = (\frac{d}{v_{ego} - v_{obj}} \pm T_{indicator}) * T_{lane} \tag{6.2.6}$$

Hence, if the TCPA of an object is smaller than a threshold τ_{ttc}, the object would be regarded as a hazard and should be marked.

$$P_3^O = 1 - \min\{(\frac{TCPA}{\tau_{ttc}}), 1\} \tag{6.2.7}$$

where τ_{tcc} is the maximum permissible time to collision so that the object should not be marked.

6.3 Decision Making

Decision making can be considered as an outcome of the cognitive process leading to the selection of a sequence of actions among several alternatives [BA01, Hug09, KT79]. Making a decision implies that there are alternative choices to be thought and in such a case it is not desired only to identify as many of these alternatives as possible but to choose the one that best fits with system goals, objectives, and values [Ful05]. In the literature [TML03, SDA06, KMM96, KT79], there exist several techniques to solve a decision problem. The selection of an appropriate method is not an easy task and depends on the concrete decision problem, as well as on the objectives of the decision makers. Sometimes "the simpler the method, the better", but complex decision problems may require complex methods.

In the current problem, we are dealing with a large number of nominal scale dependent variables. Moreover, the system should produce a binary output, either marking or blinding out decision for each object. Thus, the decision tree model [Qui86] was regarded as the most suitable technique to generate the rules which will be used to assess the object state.

6.3.1 Generating the Knowledge Base Rules

For generating the intellectual rules of the knowledge base, a simple learning mechanism was used to build the internal model of the decision tree. As shown in Fig. 6.3.1 the work flow of the training steps can be summarized as follows.

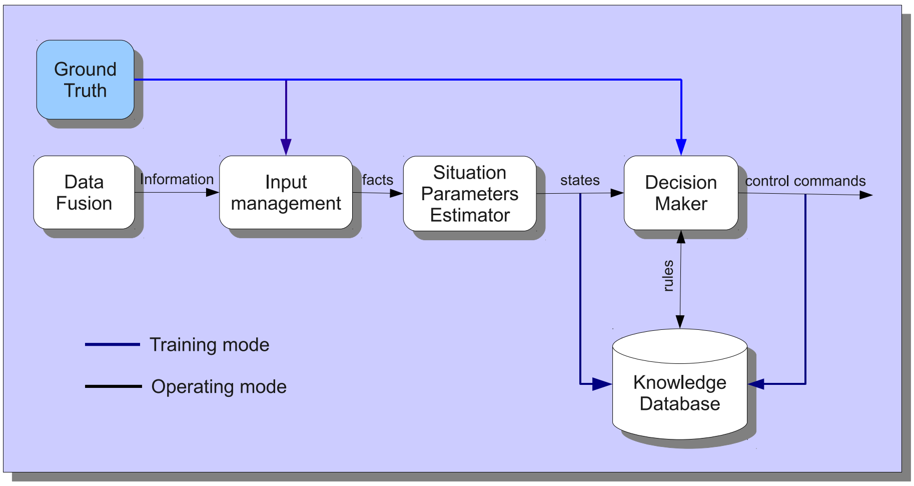

Figure 6.3.1: Work Flow of the Situation Analysis and Threat Assessment Submodule

When a new ground truth example $[x_1, x_2, x_3....x_n]$ of category Y is presented for learning, the decision tree first attempts to recognize it. If the example is not recognized by any existing rule, a new rule is automatically added to the tree in order to store the new example and its category value Y. If the example is recognized by one or more rules and they all agree that it matches a category Y, then the new example is discarded since it does not add any new information to the existing knowledge base. If the example is recognized by several rules where one or more identify it with a category other than Y, these rules which are in disagreement with the category to learn automatically reduce their similarity domain to exclude the new example. This corrective action changes the knowledge base by making certain rules more conservative in their classification process. As a result, a learning operation can have the following impact on a knowledge base:

- Add a new rule.
- Reduce the similarity of existing rules.
- Reduce the similarity domain of existing rules and add a new rule.
- Do nothing.

It is important to realize that when the similarity domain of rules are reduced, it might very well happen that an example which was recognized with the correct category at an earlier time is now no longer recognized because the rule which originally recognized the said example now excludes it. Therefore, repeating the learning procedure for all examples until the number of rules reaches a constant is a robust method to build the knowledge.

The above mentioned procedure is used to generate two knowledge bases, the first one is dedicated to recognize the base light distribution; however the second one is used to identify the illumination action for the objects

6.3.2 Situation Interpretation and Recognition

After generating the rules, the knowledge bases are used to recognize the situation. Firstly, the parameters of background estimation are fed to the decision tree classifier to determine the suitable base light distribution for the current situation. Thereafter, for each object, its set of the parameters is used to assess the threat and to identify whether the object should be marked or it should be shaded. The decided action for the object is added to its information to be used in the next assessment cycle to compensate the change in the light distribution. For example, if an object is confirmed in the current cycle as a relevant object to be marked, then the headlight will illuminate it, which means in the next cycle it will be recognized with a higher perceptibility probability than its original one, that can affect the decision making negatively. Therefore, the change in the object status is regarded during calculating the perceptibility probability by replacing the new calculated value with the original one which was estimated before the status change.

6.4 Illumination Modeling and Headlight Control

6.4.1 Light Distribution Optimization

Based on the simulation of the light distribution strategies found via the software used in [Ros05] and the practical observations, the following facts have been put into consideration while modeling the light distribution, which was studied in the previous section.

1. All the vehicles, which are less than 4° apart from each other, should be merged together.
2. For oncoming traffic, a left buffer (about 5% of the shading width) is added in order to avoid the dazzling in high relative velocity situations.
3. To compensate the high dynamics of the vehicle (e.g., pitching), a shading buffer of size approximately 5 % of the shading width is added.
4. Objects which are found in the ego lane and may produce threat to the driver should be marked for less than one second to avoid dazzling the other road participants even if there is a vehicle in the marking range.
5. If a pedestrian is detected and classified by the system, it can be, for example, continuously marked taking in consideration the above mentioned points.

6.4.2 Headlight Control

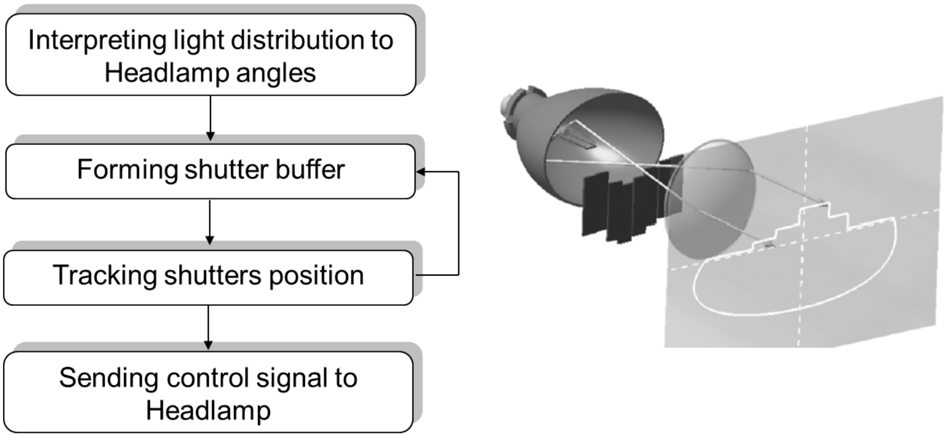
Interpreting light distribution to Headlamp angles
Forming shutter buffer
Tracking shutters position
Sending control signal to Headlamp

Figure 6.4.1: Functional Flow of Headlight Control Submodule

This function is dedicated to issue the required signals to control the shutters of the headlight to produce the computed light distribution. The procedure is carried out as follows:

1. Acquiring the current status of the headlight's shutters and compare it with the status saved from the previous cycle.
2. Convert the calculated light distribution to a sequence of physical headlight's angles, indicating which areas should be illuminated.

3. Decide if swiveling the headlights right or left is affordable to realize the the required light distribution.
4. Update the status of each shutter (up/down).
5. Send the control signal to headlight control unit.
6. Check the status of the shutters to assure that each shutter is in the required position. If any deviation detected, reset all the shutters and send the control signal again.

Chapter 7

System Evaluation and Results

System evaluation is a crediting process in which the outcomes of the system are compared to the predefined requirements. This section presents results from a series of on-road verification tests performed to determine the performance of the prototype, as well as to identify areas of system's weak points that should be improved in the next development cycle. Data was collected from tests conducted on the public roads using a BMW 5 series vehicle equipped with the LBDAS prototype. The objectives of the on-road tests were to drive the test vehicle in an uncontrolled driving environment to measure the system's susceptibility to nuisance alerts, assess alerts in perceived threatening situations, and evaluate system availability. The testing procedure is based on the guidelines published by NHTSA [NHT08] for testing the so-called "Integrated Vehicle-Based Safety Systems".

7.1 Recognition Range

Detection range is considered as the main parameter which affects dramatically the system performance; therefore this parameter has been tested exclusively. In practice, the two light distributions namely glare-free high beam and the marker light have different detection range requirements; consequently each light function has been evaluated separately. Since we are dealing with a multi sensors system, in which all the hardware components interact together and each influences the system's performance, the evaluation of the recognition range has been measured in form of the final outcomes represented in the control strategy.

7.1.1 Case Study: Glare-Free High Beam

Glare-Free function requires a long detection range (more than 800 m), which cannot be evaluated directly from the output of the recognition unit. Thus in order to measure the effectiveness recognition range of the LBDAS, the experiment (for the evaluation of the first proptype [Ros05]) has been adapted to cover the enhanced detection range by using GPS position to estimate the distance continuously.

In addition to the test vehicle, another target vehicle was equipped with a Luxmeter to measure the illuminance projected on the driver's eyes. A correct recognition of the target and a right decision taken by the LBDAS causes the target's area to be shaded out. This is perceived by the driver as if the host vehicle is driving with dipped beam light distribution; hence the illuminance detected by the Luxmeter should also be in the allowance range of the dipped beam. The detection range of the light sources in various traffic situation will be illustrated in the following section.

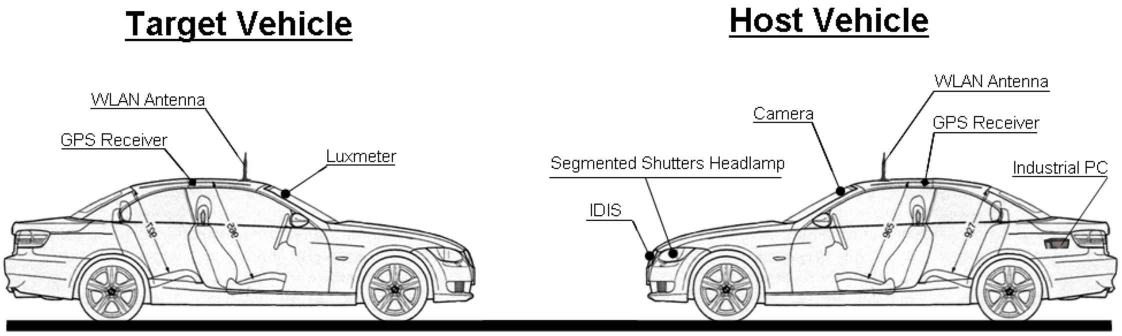

Figure 7.1.1: Schematic of the Experiment used to Measure the Projected Illuminance on the Driver Eyes of an Oncoming Vehicle

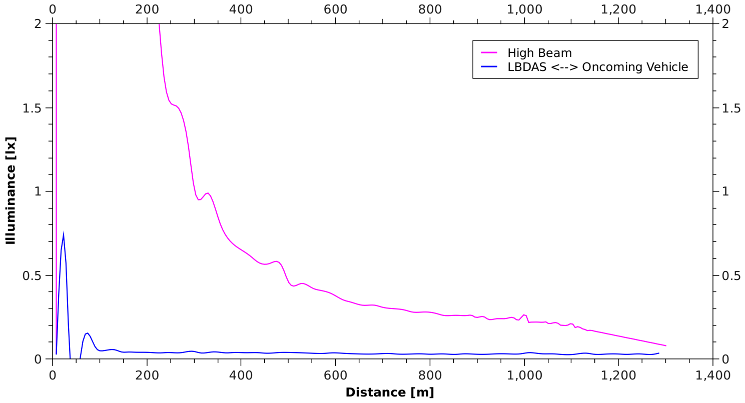

Figure 7.1.2: Illuminance Measured by Luxmeter Mounted on the Front Windshield of an Oncoming Vehicle

Figure 7.1.2 demonstrates a situation where an oncoming vehicle is detected from the system at a distance of about 1300 m and is tracked continuously. The system has modeled the light distribution of the host vehicle so that the area of the target vehicle is excluded from the illumination space, thus the illuminance measured by the Luxmeter is like that of the dipped beam. The ripples in the measured illuminance of LBDAS at distance of 100 m are resulted from the vehicle dynamics (e.g., pitching) of both the host vehicle and the target vehicle.On the other hand, Fig. 7.1.3 shows that the leading vehicle is recognized and tracked up to 650 m then the system has lost the vehicle, therefore the decision making unit has commanded the headlamps to illuminate the area of the leading vehicle with the high beam assuming that there is no glare threat anymore for the leading vehicle. Thus, the illuminance measured after losing the target vehicle is equal to that one of the high beam light distribution.

The above mentioned experiment has been carried out with two different target vehicles, the first one is Mercedes E class equipped with Xenon headlight, while the second is VW Sharan equipped with halogen headlight. For each vehicle, the average as well as the minimum and maximum recognition range for the headlights and the tail lights has been measured.

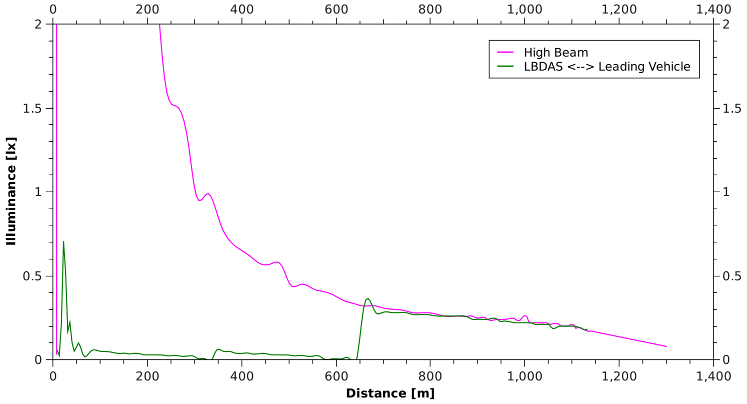

Figure 7.1.3: Illuminance Measured by Luxmeter Mounted on the Back Windshield of a Leading Vehicle

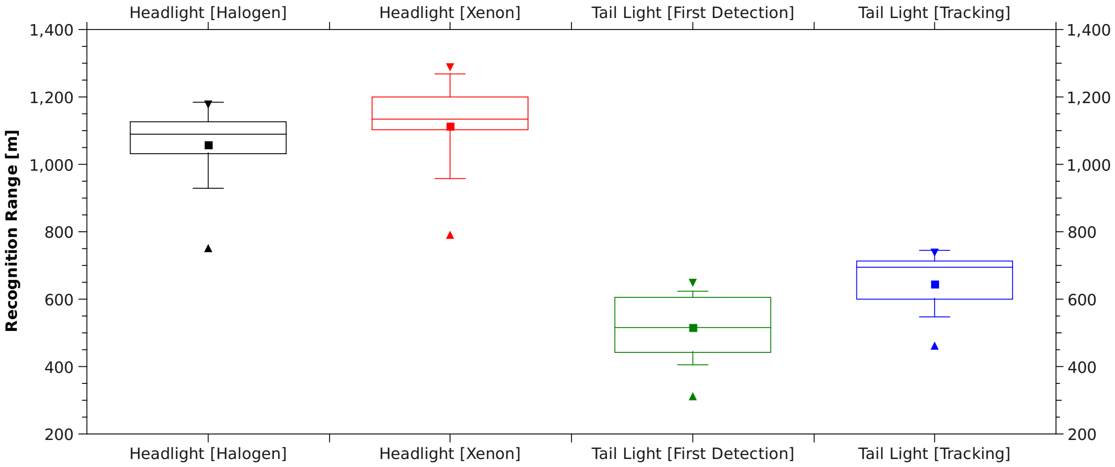

Figure 7.1.4: Recognition Distance of Light Sources in the Standalone Mode

Figure 7.1.4 shows the detection range of the system in the standalone mode; where the symbols ▲, ▼, and ■ denote the minimum, the maximum, and the average recognition range, respectively. The results demonstrate that the obtained average recognition range fulfills the system requirements which are 800 m for oncoming vehicles and 400 m for leading vehicles. The Xenon headlamps are detected early at an average distance of more than 1100 m while the halogen headlamps are registered foremost at an average distance of 1050 m. In addition, it can also be noticed that the first detection of the tail lamps fall sometimes to 350 m, however when the leading vehicle is taken over and is tracked from the system, the minimum achieved recognition range is about 450 m.

The results obtained in the cooperative mode will be illustrated and discussed later in this section.

7.1.2 Case Study: Marker Light

In contrast to the glare-free evaluation methodology, the marker light function has been directly evaluated. The recognition distance of the object has been noted when it is illuminated from the LBDAS headlight. In the experiment, three types of objects have been considered to be investigated, which are a car, a pedestrian, and a dummy obstacle in form of a carton box. The objective of the experiment is to determine the variation of maximum range of the marker light against different speeds of the host vehicle. The tests have shown that the host vehicle velocity influences dramatically the detection performance of the small objects, namely pedestrians.

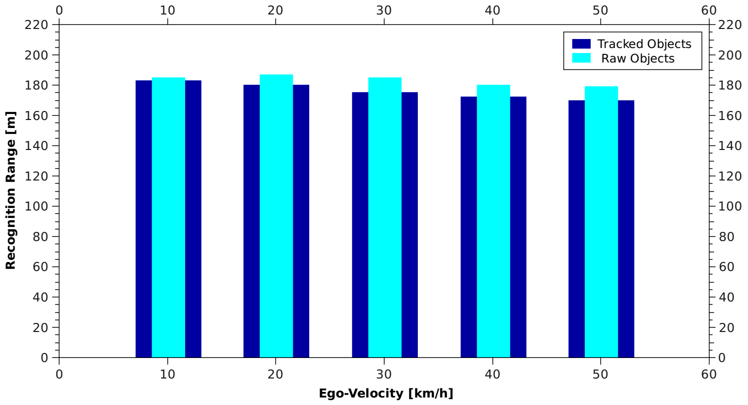

Figure 7.1.5: Range of the Marker Light Covered by IDIS for a Car

Figure 7.1.5 shows that the system can recognize the cars up to 180 m. The slight performance drop at ego-velocity of 50 km/h can be compensated by considering the raw data. However, the detection performance of pedestrians drops dramatically down from 110 m obtained at ego-velocity of 10 km/h to about 30 m at velocity of 50 km/h. Even by using the raw data, the detection range has not been increased impressively as shown in Fig. 7.1.6. The dummy object has been robustly detected from the system. Figure 7.1.7 shows that the raw object information has increased the detection performance with more than 33%.

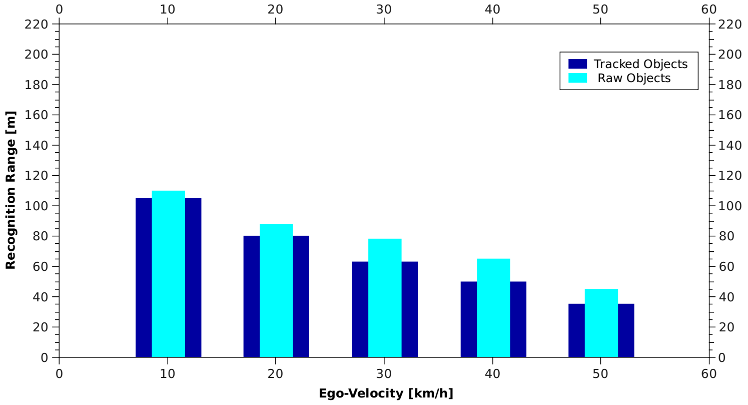

Figure 7.1.6: Range of the Marker Light Covered by IDIS for a Pedestrian

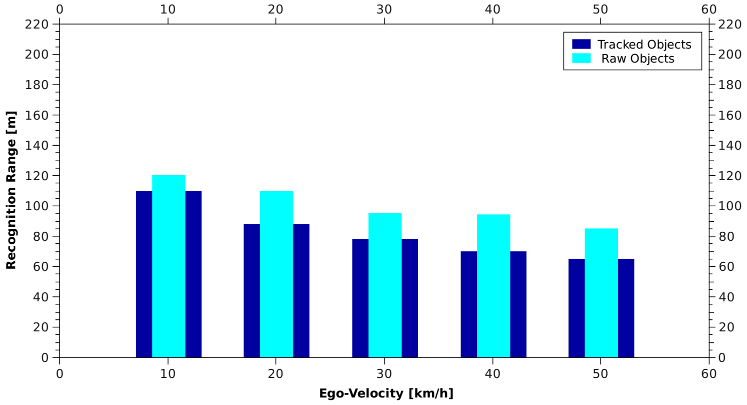

Figure 7.1.7: Range of the Marker Light Covered by IDIS for a Dummy Obstacle

7.2 Object Classification Performance

In order to describe the classification performance of the system for the various types of objects, a couple of parameters are defined as follows.

- Positive object: This is the object which is relevant to the classification module. For example, light sources are the positive objects of the camera system.
- Negative object: This denotes those ones other than the positive object. If the classification module deals with more than one positive category, the selected category is treated as the positive object, and all other categories are grouped as the negative objects.
- True positive (TP): This refers to the number of positive objects where the classification module correctly predicts their type..
- False positive (FP): In this case, the classification module assigns the positive object category to a negative object. For instance, a reflector is classified as a light source.
- True negative (TN): This represents the number of negative objects where the classification module predicts their types as a non-positive object. The correct classification of the negative objects is not counted (e.g., a traffic sign is classified as a reflector).
- False negative (FN): This denotes the case when a negative object category is assigned to a positive object.

Actual Class	Prediction Class	
	Postive Object	Negative Object
Postive Object	**TP**	**FN**
Negative Object	**FP**	**TN**

Table 7.1: Illustration of the Classification Parameters

Based on the above mentioned definitions of TP, TN, FP and FN, the following performance measures can be calculated.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7.2.1}$$

$$sensitivity = \frac{TP}{TP + FN} \tag{7.2.2}$$

$$specificity = \frac{TN}{TN + FP} \tag{7.2.3}$$

$$precision = \frac{TP}{TP + FP} \tag{7.2.4}$$

$$FMeasure = 2 \cdot \frac{precision \cdot sensitivity}{precision + sensitivity} \tag{7.2.5}$$

where accuracy indicates how close is the quality of classification process to the true value, sensitivity denotes the proportion of the correctly predicted positive objects, specificity is the proportion of correctly identified of negative objects, precision represents the repeatability of the classification, and F-Measure [Sas07] is the subcontrary mean of precision and sensitivity which gives an overall measure of the quality of the prediction.

7.2.1 Light Sources Classification Performance

The classification quality of the light sources is measured as the capability of the system to distinguish between the light sources and the other infrastructure reflections. The results represented in table 7.2 show a robust classification accuracy of more than 95% with a precision of about 93%. It is worth to mention that the accrediting process is independent of the type of the light source (e.g., a headlight or a tail light); i.e. if a bright tail light is classified as white headlight, it has been considered as a correct classification one since both are light sources.

Performance Measure	Value
Accuracy	95.10%
Sensitivity	91.48%
Specificity	96.78%
Precision	92.95%
F-Measure	0.9221

Table 7.2: Classification Performance Measures of the Light Sources

Figure 7.2.1 demonstrates the Receive Operating Characteristic (ROC) curve of the light sources classification performance. The ROC [Han89] is a graphical plot of the system sensitivity against the false positive rate which is equivalent to (1- specificity). The results presented in the figure show that the system can achieve more than 99% true positive rate when 10% false alarms is allowed.

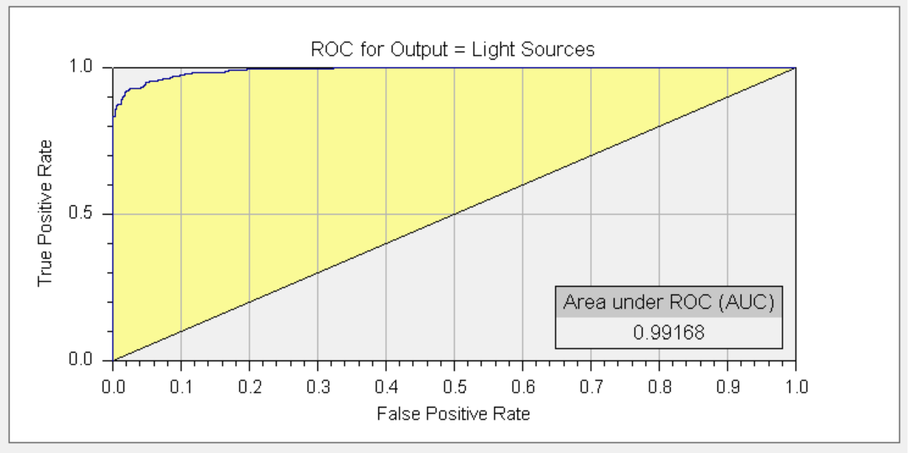

Figure 7.2.1: Light Sources Receive Operating Characteristic Curve

7.2.2 IDIS Objects Classification Performance

The IDIS object recognition sub-function has shown an accurate classification performance of more than 99 % for all non-stationary objects as shown in table 7.3.

	C_{SD}	C_{OD}	P_{SD}	P_{OD}	B_{SD}	B_{OD}
Accuracy	100.00%	99.82%	99.64%	99.94%	99.82	99.70%
Sensitivity	100.00%	97.56%	98.43%	100.00%	100.00%	100.00%
Specificity	100.00%	100.00%	100.00%	99.92%	99.80%	99.64%
Precision	100.00%	100.00%	100.00%	99.76%	98.11%	98.29%
F-Measure	1	0.9877	0.9934	0.9988	0.9905	0.9914

Table 7.3: IDIS Classification Performance Measures

Category	C_{SD}	C_{OD}	P_{SD}	P_{OD}	B_{SD}	B_{OD}
C_{SD}	315	0	0	0	0	0
C_{OD}	0	120	0	0	0	3
P_{SD}	0	0	376	1	5	0
P_{OD}	0	0	0	413	0	0
B_{SD}	0	0	0	0	288	0
B_{OD}	0	0	0	0	0	156

Table 7.4: IDIS Classification Confusion Matrix

where C_{SD} denotes the cars driving in the same direction of the host vehicle while C_{OD} represents the cars driving in the opposite direction. Analogue, P denotes pedestrians and B is for the bicycles.

The classifier is able to identify leading cars without any false alarms, however slow distant oncoming cars are miss-classified as oncoming bicycles as can be concluded from the confusion matrix shown in table 7.4. In addition, a number of fast pedestrians running in the same direction of the host vehicle have been classified as ongoing bicycles.

7.3 Performace Evaluation of Different System Configurations

Different hardware configurations have been tested to estimate the influence of each component on the overall system performance. The results can be summarized as follows:

7.3.1 IDIS

Classification of IDIS objects has improved the performance by more than 5%. Such improvement is obtained by filtering the camera's miss-classified stationary objects, like guiding reflectors and traffic signs as well as detecting the vehicles which are too close to the host vehicle and their hypotheses have been filtered out in the blob extraction procedure. However, this is valid as long as the driving road is straight. In sharp curves, stationary objects had been detected from IDIS as movable ones with high lateral velocity, which increases the system nuisance alert rates by about 30%. Therefore, in absence of valid road boundaries from the data fusion module, distant objects detected by IDIS, while the host vehicle driving in the curvature lane, should be neglected and therefore do not contribute in the decision making regarding the glare free high beam. However, this drawback has not proved a serious effect on the marker light, since each object will be checked if it is bright enough or not to be marked. Practically, it has been observed that illuminating the obstacles in curvature can increase the driver safety by illuminating the guiding rails, for example, by warning the driver about the road boundaries. Nevertheless, this situation should be investigated further in details.

7.3.2 Camera

Camera has shown a significant contribution in reducing about 60% of the false alarm rates of the marker light. The image information (for the calculation of the perceptibility) and the detected lane position (for the adjustment of the extracted road boundary from the digital maps) have increased the system reliability by avoiding the unnecessary illumination of objects, like vehicles standing in red traffic light and miss-allocated guiding reflectors.

7.3.3 Digital Maps

Accurate lane detection and robust driving path estimation have played an essential role in increasing the system performance for both glare-free and marker light. Right associating objects to the road lanes made it possible to correct the miss-classified light sources and to determine accurately the relevant objects for marker light. The altitude information has improved the object matching and association process in the data fusion by adjusting the vanishing point of the camera which has a great effect in projecting the IDIS objects and the road boundaries. However, the inaccurate GPS position influences the use of the digital maps dramatically, as with the current GPS accuracy it is hardly possible to determine the absolute position of the host vehicle on the road, especially in multiple lanes roads. Therefore, adjusting road boundaries via the lane information obtained from the vision system has increased impressively the benefits of using the digital maps (for detailed information, refer to section 5.2).

7.3.4 Car-to-Car Communication

Independent of the driving direction (e.g., oncoming or ongoing), a detection range of 800 m as well as a 0.5 degree heading angle accuracy have been successfully achieved by using C2C communication.

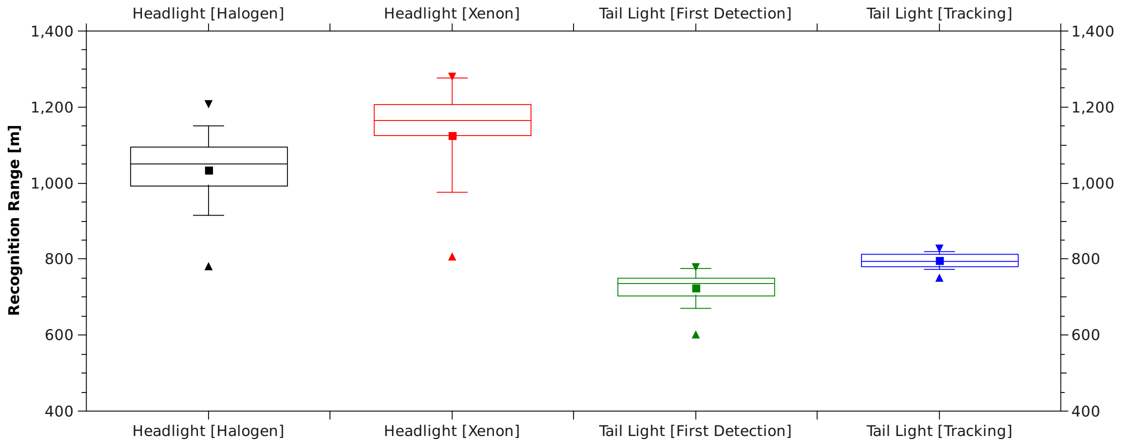

Figure 7.3.1: Recognition Distance of Light Sources in the Cooperative Mode

As shown in Fig. 7.3.1, C2C has enhanced the detection range of the distant leading vehicles by about 35% . However, it offers no enhancement in detection of the oncoming vehicles, since this issue is already addressed from the camera quite effectively. In addition, C2C has increased the use of the glare free function in curves by about 70% in situations when no other vehicles are detected.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

The presented system can be considered as a new step on the road to solve the conflict between improvement of driver visibility and dazzling of other road participants. The developed LBDAS in some aspects could be considered as an extension to the first prototype introduced in [Ros05]. The main function of the system "Glare-Free high beam" has been inherited. Otherwise, the following enhancements have been added. Hardware and software architectures have been designed which enabled testing various hardware configurations as well as evaluating the contribution of each component on the overall system performance. New object detection sensors and recognition techniques have been investigated and utilized to fulfill the requirements of the system. New methodologies for situation analysis and target selection have been implemented which supported the system in correctly producing an optimal light distribution. The provisional DMD headlight has been replaced by a new automotive apt addressable headlamp (Segmented-Shutters Headlamp) which enabled testing the system in uncontrolled environments and in real traffic situations. Last but not least, a new light function "Marker Light" has been integrated. The new light distribution aims at drawing the driver's attention to any potential hazard as early as possible by directing marker lights to these objects that are of particular relevance to the visual perception of the driver. Marker light is intended to be used wherever the glare-free high beam is not permitted; for example, on the city roads. However, marker light is considered as an opposite implementation to the glare-free high beam, in this thesis a new strategy has been introduced showing that merging the glare-free high beam with the marker light in one system provides the driver with superior flexible light distribution, which can be called a full speed LBDAS.

The system is designed according to the flow of information as follows. Firstly, both internal and external sensors are needed to monitor the host vehicle status and the environment in the vicinity of it. Afterwords, the acquired information is processed. Among others, this mechanism involves disciplines such as object recognition to gain the necessary knowledge about the host vehicle's surrounding as well as driving dynamics calculations to take the host vehicle status into account. In the central information processing, all the data collected by the sensor systems is analyzed, sorted, and interpreted. Merging all the necessary data yields the system's internal representation of the current situation. Consequently, an appropriate strategy is chosen for the given situation. The strategy includes information on the basis of which objects are blinded out or marked and how to do so. Finally, the chosen strategy is applied to the situation and the desired light distribution is projected. The developed solution works in two operating modes. The first operating mode is a standalone mode, where the host vehicle has no physical contact with other road participants. The other operating mode is a cooperative mode, where all vehicles are elements in a cooperative communication network, and each of them has the ability to get information about other vehicles in its surrounding. It is worth mentioning that for the time being, the cooperative mode cannot replace the standalone mode, nevertheless, it is actually an extension to it, simply because building secure vehicles networks and preparing the required infrastructure to support DAS are still requiring further research. The aim of investigating the cooperative communication network mode was to measure the benefits and the performance gain when these techniques are available in the market.

LBDAS demand fast and robust object detection methods in order to perform efficiently. The thesis introduced new techniques to detect vehicles at night, based on their light sources for a distance up to 1000 meters as well as new methodologies to classify objects of the infrared sensor (IDIS) based on object's dynamics. C2C communication has been investigated and showed a robust performance within a communication range of 800 meters.

Assessing threat possibility depends mainly on classification quality of sensors data. In order to fulfill the system requirements concerning the detection range and classification accuracy, methods for sensors data fusion have been implemented. The thesis presented hybrid data fusion architecture by constructing a communication network using WinSocket programming techniques. The architecture was chosen to ensure the Independence of the remote-sensors (e.g., IDIS and camera) modules from each other in order to minimize the false alarms in the system, which are produced by classification uncertainty of each sensor. The CPU time of the main data fusion unit has been used as a global timing reference to solve data synchronization issues in the adapted communication network. The correlation of the different sensors data is performed in the two-dimensional image space as a unified plane to project all sensors data on. The sensors data association is realized by using an optimized approach based on constructing association metrics which depends on correlating the sensors information via spatial and identity matching.

GPS position of the host vehicle is used to extract the road's coordinates in front of the host vehicle from the data base of the digital maps. The low update rate of GPS receiver (1 Hz) imposes the need for extrapolating the host vehicle position up to 20 Hz in order to obtain an adequate performance. Therefore, a linear vehicle model has been integrated with the GPS signals through Kalman filter to update the geographical position based on the dynamics of the host vehicle. Digital maps provide the system with means to estimate the relevancy of objects on the road by relating their position to the road, which is used in assessing the threat of a possible collision.

The system performance has been evaluated in various traffic situations. The results have demonstrated a promising performance for the proposed LBDAS regarding detection range, real-time capability, and false alarm rate.

8.2 Future Work

All objectives of the thesis have been realized and a prototype was successfully implemented based on the objects data as well as environment information obtained from various sensors. However, the prototype has been tested qualitatively under limited conditions in relatively simple situations due to the time constraints as well as the limited resources such as availability of the digital maps for all roads during the development phase. Therefore, extending the system and testing it in more complex situations are highly recommended. In addition, the system activation/deactivation parameters (e.g., ego velocity) should be fine-tuned based on an evaluation from test drivers.

This research has demonstrated that an accurate system for a real-time classification of objects in traffic situations based on their dynamics can be implemented using Hella ACC IDIS sensor. IDIS has shown a robust performance in detecting of the vehicles as well as the bulky objects. However, combining IDIS with a passive night vision system can enhance the detection range of pedestrians and it can also be used to verify the classified IDIS objects, which increases the usability benefits of the marker light function.

Distant street lights as well as traffic lights are main sources of the system false alarms. Therefore, developing algorithms to filter such objects can increase the robustness of the system.

As mentioned previously, the digital maps, which are used currently in the system, were created for the development purpose. Thus, integrating the developed GPS position-correction algorithm with professional digital maps can improve the performance of marker light.

The resolution of the Segmented-Shutter headlamp is acceptable taking in consideration that it is an automotive apt addressable headlamp solution; however using a headlamp with a resolution less than one degree can enhance obviously the system accuracy.

Bibliography

[ABJ+08] P. F. Alcantarilla, L. M. Bergasa, P. Jimenez, M. A. Sotelo, I. Parra, D. Fernandez, and S. S. Mayoral, *Night time vehicle detection for driving assistance lightbeam controller*, Proc. IEEE Intelligent Vehicles Symposium, 2008, pp. 291–296.

[ACC04] L. C. G. Andrade, W. F. M. Campos, and R. L. Carceroni, *A video-based support system for nighttime navigation in semi-structured environments*, Proc. 17th Brazilian Symposium on Computer Graphics and Image Processing, 17–20 Oct. 2004, pp. 178–185.

[BA01] E. Borgonovo and G.E. Apostolakis, *A new importance measure for risk-informed decision making*, Reliability Engineering & System Safety **72** (2001), no. 2, 193 – 212.

[BB] Thomas Bachmann and Stephan Bujnoch, *Connecteddrive - driver assistance systems of the future*, Commercial Applications of Satellite-Navigation, 1–14.

[BBC+07] M. Bertozzi, A. Broggi, C. Caraffi, M. Del Rose, M. Felisa, and G. Vezzoni, *Pedestrian detection by means of far-infrared stereo vision*, Computer Vision and Image Understanding **106** (2007), no. 2-3, 194 – 204, Special issue on Advances in Vision Algorithms and Systems beyond the Visible Spectrum.

[BBF00] Massimo Bertozzi, Alberto Broggi, and Alessandra Fascioli, *Vision-based intelligent vehicles: State of the art and perspectives*, Robotics and Autonomous Systems, 2000.

[BBG06] J. Berssenbriigge, J. Bauch, and J. Gausemeier, *A night drive simulator for the evaluation of a predictive advanced front lighting system*, Proc. ITI 4th International Conference on Information & Communications Technology ICICT '06, 2006, pp. 1–2.

[BBK+03] G. Bierleutgeb, M. Burg, W. Kesseler, M. Kleinkes, and J. Locher, *Infrared-based driver assistance for enhanced perception at night*, 5th International Symosium on Progress in Automobile Lighting, 2003.

[BD98] Z. Berman and J. D.Powell, *The role of dead reckoning and inertial sensors in future general aviation navigation*, IEEE Position Location and Navigation Symp., 1998.

[BdMRJ04] M Bellino, Y Lopez de Meneses, P Ryser, and J Jacot, *Lane detection algorithm for an onboard camera*, SPIE proceedings of the first Workshop on Photonics in the Automobile, 2004.

[BFT06] A. Broggi, R. L. Fedriga, and A. Tagliati, *Pedestrian detection on a moving vehicle: an investigation about near infra-red images*, Proc. IEEE Intelligent Vehicles Symposium, 2006, pp. 431–436.

[Bis02] Bishof, *The mechatronics handbook.*, United States: Commercial, 2002.

[BL07a] M. Boehm and J. Locher, *Lichtbasierte fahrerassistenzsysteme: Gestaltung und bewertung im hinblick auf akzeptanz und kundennutzen*, Tagungsband des 16. Aachener Kolloquiums Fahrzeug- und Motorentechnik, 2007.

[BL07b] ———, *Lighting based driver assistance systems: Design and evaluation with regard to customers acceptance and benefit*, Aachen colloquium Automobile and Engine Technology, 2007, pp. 861–869.

[Boe05] Ch. Boehlau, *Multibeam lidar acc approaching the start of production*, Proceedings of ISAL 2005, 6th International Symposium on Automotive Lighting, 2005, pp. 82–90.

[BSW99] Christopher L. Bowman, Alan N. Steinberg, and Franklin E. White, *Revisions to the jdl data fusion model*, Architectures,Algorithms and Applications III (Orlando, FL), Processdins of the SPIE Sensor Fusion, March 1999, pp. pp 430–441.

[BTB05] Sabri Boughorbel, Jean-Philippe Tarel, and Nozha Boujemaay, *Generalized histogram intersection kernel for image recognition*, International Conference on Image Processing, 2005.

[BTMM04] C. Blanc, L. Trassoudaine, Le Guillloux Y. Moreira, and R. Moreira, *Track to track fusion method applied to road obstacle detection*, Processdings of the 7th International Conference on Information Fusion (Stockholm, Sweden), June 2004, pp. pp:775–782.

[Bur02] Road Bureau, *Its - a key social technology initiative for the 21st century*, Tech. report, Supervised by Road Bureau, The Ministry of Land, Infrastructure and Transport, 2002.

[CCCW06] Yen-Lin Chen, Yuan-Hsin Chen, Chao-Jung Chen, and Bing-Fei Wu, *Nighttime vehicle detection for driver assistance and autonomous vehicles*, Proc. 18th International Conference on Pattern Recognition ICPR 2006, vol. 1, 2006, pp. 687–690.

[CCHF09a] C. K. Chan, K. W. E. Cheng, S. L. Ho, and T. M. Fung, *Development of electric vehicle with advanced lighting system and all electric drive*, Proc. 3rd International Conference on Power Electronics Systems and Applications PESA 2009, 2009, pp. 1–8.

[CCHF09b] ———, *Simulation of the control method for the adaptive front lighting system*, Proc. 3rd International Conference on Power Electronics Systems and Applications PESA 2009, 2009, pp. 1–4.

[CH03] Ming-Yang Chern and Ping-Cheng Hou, *The lane recognition and vehicle detection at night for a camera-assisted car on highway*, Proc. IEEE International Conference on Robotics and Automation ICRA '03, vol. 2, 14–19 Sept. 2003, pp. 2110–2115.

[CHTW06] Yinghao Cai, Kaiqi Huang, Tieniu Tan, and Yunhong Wang, *Context enhancement of nighttime surveillance by image fusion*, Proc. 18th International Conference on Pattern Recognition ICPR 2006, vol. 1, 2006, pp. 980–983.

[CLF$^+$08] Yen-Lin Chen, Chuan-Tsai Lin, Chung-Jui Fan, Chih-Ming Hsieh, and Bing-Fei Wu, *Vision-based nighttime vehicle detection and range estimation for driver assistance*, Proc. IEEE International Conference on Systems, Man and Cybernetics SMC 2008, 2008, pp. 2988–2993.

[Com01] European Commission, *Transport policy white paper.*

[CWF09] Yen-Lin Chen, Bing-Fei Wu, and Chung-Jui Fan, *Real-time vision-based multiple vehicle detection and tracking for nighttime traffic surveillance*, Proc. IEEE International Conference on Systems, Man and Cybernetics SMC 2009, 2009, pp. 3352–3358.

[Dan03] Hongshe Dang, *A new data association approach for automotive radar tracking*, Proceddings of the 6th international conference on information fusion, July 2003, pp. Volume 2,pp 1384–1388.

[DD05] D. Dudley and C. Dunn, *Dlp technologie nicht nur fuer projektoren und fernsehen*, Photonik, vol. 1, 2005, pp. 32–35.

[DGL07] Jianfei Dong, Junfeng Ge, and Yupin Luo, *Nighttime pedestrian detection with near infrared using cascaded classifiers*, Proc. IEEE International Conference on Image Processing ICIP 2007, vol. 6, Sept. 16 2007–Oct. 19 2007, pp. VI–185–VI–188.

[DH72] R O Duda and P E Hart, *Use of hough transform to detect lines and curves in picture*, Communications of the ACM (1972), no. 15, 11–15.

[DLL06] D. Decker, R. Lachmayer, and J. Locher, *Enablers for safe night-time driving*, Convergence International Congress and Exposition On Transportation Electronics, 2006.

[DST92] Michael B. Dillencourt, Hannan Samet, and Markku Tamminen, *A general approach to connected-component labeling for arbitrary image representations*, vol. 39, ACM, April 1992, pp. 253–280.

[Ess08] Abas Essam, *Sensor data fusion for light based driving assistance system*, Master's thesis, Maschinenbau Fakultaet, Uni Paderborn, 2008.

[EWG01] Frank Ewerhart, Stefan Wolf, and Dietrich Gall, *Video based curve light system sensor, system and results*, PAL 2001, 2001.

[FD06] B. R. Fajen and M. C. Devaney, *Learning to control collisions: The role of perceptual attunement and action boundaries*, vol. 32, 2006, pp. 300–313.

[FDEW02] K. Ch. Fuerstenberg, K. Dietmayer, S. Eisenlauer, and V. Willhoeft, *Multilayer laserscanner for robust object tracking and classification in urban traffic scenes*, 9th World Congress on Intelligent Transport Systems, 2002, pp. 7–8.

[FDL03] K. Ch. Fuerstenberg, K. C. J. Dietmayer, and U. Lages, *Laserscanner innovations for detection of obstacles and road*, 7th International Conference on Advanced Microsystems for Automotive Applications, 2003.

[FDW02] K. Fuerstenberg, K. Dietmayer, and V. Willhoeft, *Pedestrian recognition in urban traffic using a vehicle-based multilayer laserscanner*, IEEE Intelligent Vehicles Symposium, 2002, pp. 31–35.

[FHL05] A. Ferrein, L. Hermanns, and G. Lakemeyer, *Comparing sensor fusion techniques for ball position estimation*, RoboCup Symposium, 2005.

[FL02] K. Ch. Fuerstenberg and U. Lages, *Pedestrian detection and classification by laserscanners*, IEEE Intelligent Vehicles Symposium 2002, 2002.

[Fle03] Benoist Fleury, *A high performance night vision system*, 2003, pp. 281–300.

[FP02] David A. Forsyth and Jean Ponce, *Computer vision: A modern approach*, Prentice Hall, 2002.

[Ful05] Janos Fulop, *Introduction to decision making methods*, Laboratory of Operations Research and Decision Systems, Computer and Automation Institute, Hungarian Academy of Sciences, 2005.

[FW06] David J. Fleet and Yair Weiss, *Handbook of mathematical models in computer vision*, Springer. ISBN 0387263713., 2006.

[GE05] Mirco Goetz and Karsten Eichhorn, *Optical technologies for future headlamps*, ISAL 2005 Symposium, Darmstadt University of Technology, 2005.

[GK08] Goetz and Kleinkes, *Headlamps for light based driver assistance*, Optical sensors 2008, vol. 7003, 2008.

[GLT09] Junfeng Ge, Yupin Luo, and Gyomei Tei, *Real-time pedestrian detection and tracking at nighttime for driver-assistance systems*, vol. 10, 2009, pp. 283–298.

[GLX05] Junfeng Ge, Yupin Luo, and Deyun Xiao, *Adaptive hysteresis thresholding based pedestrian detection in nighttime using a normal camera*, Proc. IEEE International Conference on Vehicular Electronics and Safety, 14–16 Oct. 2005, pp. 46–51.

[GPFZ06] Arturo Gil-Pinto, Philippe Fraisse, and Rene Zapata, *Wireless reception signal strength for relative positioning in a vehicle robot formation*, Proc. IEEE 3rd Latin American Robotics Symposium LARS '06, 2006, pp. 100–105.

[GPFZ08] A. Gil-Pinto, P. Fraisse, and R. Zapata, *Wireless communication for secure positioning in multi robot formations of non holonomic ground vehicles*, Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS 2008, 2008, pp. 4198–4198.

[Gro] Jon L. Grossman, *Thermal infrared vs. active infrared: A new technology begins to be commercialized*, Tech. report, IRINFO.

[Han89] J. A. Hanley, *Receiver operating characteristic (roc) curves methodology: The state of the art*, Critical Reviews in Diagnostic Imaging, 1989.

[Hel06] Hella, *Adaptive hell-dunkel-grenze (ahdg) system analysis*, Tech. report, 2006.

[HLL05] Norbert Hoever, Bernd Lichte, and Steven Lietaert, *Multi-beam lidar sensor for active safety applications*, SAE International, 2005.

[HM04] David L. Hall and Sonya A. H. McMullen, *Mathematical techniques in multisensor data fusion*, Artech Print on Demand, 2004.

[Hon07] Hendrik Honsel, *Entwurf und programmierung von modellen zur bewegungsdynamik in ad-hoc-wlan-netzwerken zur fahrzeug-zu-fahrzeug-kommunikation*, Master's thesis, Fachhochschule Gelsenkirchen Abt. Bocholt, Fakultaet Maschinenbau, 2007.

[HRD01] U. Hofmann, André Rieder, and Ernst D. Dickmanns, *Radar and vision data fusion for hybrid adaptive cruise control on highways*, ICVS '01 Proceedings of the Second International Workshop on Computer Vision Systems, 2001.

[HT08] M. A. Hogervorst and A. Toet, *Presenting nighttime imagery in daytime colours*, Proc. 11th International Conference on Information Fusion, 2008, pp. 1–8.

[HTC07] Wei-Lieh Hsu, Chang-Lung Tsai, and Tsung-Lun Chen, *Traffic detection at nighttime using entropy measurement*, Proc. Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing IIHMSP 2007, vol. 1, 26–28 Nov. 2007, pp. 585–588.

[Hug09] Warren R. Hughes, *A statistical framework for strategic decision making with ahp: Probability assessment and bayesian revision*, Omega **37** (2009), no. 2, 463 – 470.

[Hur05] Michael B. Hurley, *An extension of statistical decision theory with information theoretic cost functions to decision fusion: Part ii*, Information Fusion **6** (2005), no. 2, 165 – 174.

[JMF99] A K Jain, M N Murty, and P J Flynn, *Data clustering: a review*, ACM Comput Surv (1999), no. 31.

[JTM+06] D. Jiang, V. Taliwal, A. Meier, W. Holfelder, and R. Herrtwich, *Design of 5.9 ghz dsrc-based vehicular safety communication*, IEEE Wireless Communications, 2006.

[KEW04] R. Kauschke, K. Eichhorn, and J. Wallaschek, *Aktive scheinwerfer zur subtraktiven lichtverteilungserzeugung*, DGaO Proc, 2004.

[KHW02] E. J. Krakiwsky, C. B. Harris, and R. V. C. Wong, *A kalman filter for integrating dead reckoning, map matching and gps positioning*, Position Location and Navigation Symposium (2002), 39–46.

[KMM96] G Kamberova, R Mandelbaum, and M Mintz, *Statistical decision theory for mobile robotics: theory and application*, In Proc. IEEE/SICE/RSJ Int. Conf. On Multisensor Fusion and Integration for Intelligent Systems, 1996.

[Kos03] Walter J. Kosmatka, *Differences in detection of moving pedestrians attributable to beam patterns and speeds*, PAL2003 Symposium, Darmstadt University of Technology, 2003.

[KT79] D Kahneman and A Tversky, *Prospect theory: An analysis of decision under risk*, Econometrica (1979), no. 47, 263–291.

[Kuh06] Patrick Kuhl, *Anpassung der lichtverteilung des abblendlichtes an den vertikalen strassenverlauf*, Ph.D. thesis, Paderborn University, 2006.

[KV05] D. Kliebisch and S. Voelker, *Examinations of the recognition distance of headlamps*, Proceedings of the 6th International Symposium on Automotive Lighting, 2005.

[LB97] Klaus Langwieder and Hans Baeumler, *Charakteristik von nachtunfaellen*, vol. PAL 1997, 1997.

[LFL05] Lakemeyer, Alexander Ferrein, and L.Hermanns., *Comparing sensor fusion techniques for ball estimation.*, RoboCup Symposium 2005, 2005.

[LHB+08] Antonio Lopez, Joerg Hilgenstock, Andreas Busse, Ramon Baldrich, Felipe Lumbreras, and Joan Serrat, *Nighttime vehicle detection for intelligent headlight control*, Advanced Concepts for Intelligent Vision Systems, 2008.

[LHF+08] B. Lucas, R. Held, D. Freundt, M. Klar, and M. Maurer, *Frontsensorsystem mit doppel long range radar*, 5th Workshop Fahrerassistenzsysteme, 2008.

[LKH02] Ilkwang Lee, Hanseok Ko, and D. K. Han, *Multiple vehicle tracking based on regional estimation in nighttime ccd images*, Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '02), vol. 4, 13–17 May 2002, pp. IV–3712–IV–3715.

[LKVB03] J. Locher, M. Kleinkes, S. Völker, and G. Bierleutgeb, *Night vision: Erhoehung der verkehrssicherheit durch infrarot-nachtsichtsysteme*, VDI: Optische Technologien in der Fahrzeugtechnik, 2003.

[LMS85] Leslie Lamport and P. M. Melliar-Smith, *Synchronizing clocks in the presence of faults*, Journal of the Ascciation for Computing Machinery, (JACM) **32, No. 1** (1985), no. 1, pp: 52–78.

[LSH04] Cheol-Hee Lee, Kuk Sagong, and Yeong-Ho Ha, *A method for generating simulated nighttime road images under automotive headlamp lights*, vol. 53, 2004, pp. 695–704.

[LTC+08] Ming-Chih Lu, C. P. Tsai, Ming-Chang Chen, Yin Yu Lu, Wei-Yen Wang, and Chen-Chien Hsu, *A practical nighttime vehicle distance alarm system*, Proc. IEEE International Conference on Systems, Man and Cybernetics SMC 2008, 12–15 Oct. 2008, pp. 3000–3005.

[Lue04] Andreas Luebke, *Car-to-car communication - technologische herausforderungen*, Proceedings VDE Kongress 2004, 2004.

[LV04] J. Locher and S. Voelker, *The influence of vehicle beam patterns on safety and acceptance*, SAE World Congress, 2004.

[Mac67] J. B. MacQueen, *Some methods for classification and analysis of multivariate observations*, vol. 1, 1967, pp. 281–297.

[Mah06] Eslam Abdul Aziz Abdul Aziz Mahmoud, *Development of a vision-based headlamp detection system for intelligent vehicles*, Master's thesis, Faculty of Mechanical Engineering at Paderborn University in partial fulfillment, 2006.

[MDLR07a] M. Maehlisch, K. Dietmayer, O. Loehlein, and W. Ritter, *Advanced microsystems for automotive applications 2007, vdi buch*, ch. Exploiting Latest Developments in Signal Processing and Tracking for Smart Multi-Sensor Multi-Target ACC, pp. 75–91, springer Berlin, 2007.

[MDLR07b] ______, *De-cluttering with integrated probabilistic data association for multisensor multitarget acc vehicle tracking*, Proc. IEEE Intelligent Vehicles Symposium, 2007, pp. 178–183.

[MFS] J McDonald, J Franz, and R Shorten, *Application of the hough transform to lane detection in motorway driving scenarios*, in Proc. of the Irish Signals and Systems Conference, pp. 340–345.

[MFS01] ______, *Application of the hough transform to lane following in motorway driving scenarios*, Proc. of the Irish Signals and Systems Conference, 2001.

[MHRD06] Mirko Mahlisch, Rudiger Hering, Werner Ritter, and Klaus Dietmayer, *Heterogeneous fusion of video, lidar and esp data for automotive acc vehicle tracking*, Proc. IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, 2006, pp. 139–144.

[MS07] Sebastian Mojrzisch and Steffen Strauss, *High dynamic headlamp system based on a segmented shutter*, ISAL 2007 Symposium, Darmstadt University of Technology, 2007.

[MSRD06] M. Mahlisch, R. Schweiger, W. Ritter, and K. Dietmayer, *Sensorfusion using spatio-temporal aligned video and lidar for improved vehicle detection*, Proc. IEEE Intelligent Vehicles Symposium, 2006, pp. 424–429.

[NA02] Mark S. Nixon and Alberto S. Aguado, *Feature extraction and image processing*, Newnes, 2002.

[NHO03] S. Nagumo, H. Hasegawa, and N. Okamoto, *Extraction of forward vehicles by front-mounted camera using brightness information*, Proc. Canadian Conference on Electrical and Computer Engineering IEEE CCECE 2003, vol. 2, 4–7 May 2003, pp. 1243–1246.

[NHT08] NHTSA, *Integrated vehicle-based safety systems light-vehicle on-road test report*, Tech. report, U.S. Department of Transportation, National Highway Traffic Safety Administration, 2008.

[Oda90] T. Oda, *An algorithm for prediction of travel time using vehicle sensor*, Proc. Third International Conference on Road Traffic Control, 1990, pp. 40–44.

[OGWM94] R. A. Olson, R. L. Gustavson, R. J. Wangler, and R. E. McConnell, *Active-infrared overhead vehicle sensor*, IEEE Transactions on Vehicular Technology, vol. 43, 1994, pp. 79–85.

[OJG10] Ronan O'Malley, Edward Jones, and Martin Glavin, *Detection of pedestrians in far-infrared automotive night vision using region-growing and clothing distortion compensation*, Infrared Physics & Technology **53** (2010), no. 6, 439 – 449.

[OLS06] Dragan Obradovic, Henning Lenz, and Markus Schupfner, *Fusion of map and sensor data in a modern car navigation system*, Journal of VLSI Signal Processing, Springer Science **45, No. 1-2** (2006), no. 45, pp 111–122.

[OM99] K Owens and L Matthies, *Passive night vision sensor comparison for unmanned ground vehicle stereo vision navigation*, In Proc. of the IEEE Intl. Conf. on Robotics and Automation, 1999.

[OS86] Paul L. Olson and Michael Sivak, *Perception-response time to unexpected roadway hazards*, vol. 28, 1986, pp. 91–99.

[PDEP01] J.D. Powell, D.Gebre-Egziabher, and P.Enge, *Design and performance analysis of a low-cost aided dead reckoning navigation system*, Gyroscopy and Navigation 4, 2001.

[pjG08] Xu pingping, Song jianguo, and Shen Guangdi, *The design on adaptive front lighting system (afs) based on brushless dc motor*, Proc. International Conference on Electrical Machines and Systems ICEMS 2008, 2008, pp. 1442–1444.

[PM01] P.Viola and M.Jones, *Rapid object detection using a boosted cascade of simple features*, vol. Computer Vision and Pattern Recognition, 2001, pp. 511–518.

[Qui86] J R Quinlan, *Induction of decision trees*, Machine Learning, 1986, pp. 81–106.

[RBMC09] J. Rebut, B. Bradai, J. Moizard, and A. Charpentier, *A monocular vision based advanced lighting automation system for driving assistance*, Proc. IEEE International Symposium on Industrial Electronics ISIE 2009, 2009, pp. 311–316.

[RG04] Jihan Ryu and J. Christian Gerdes, *Integrating inertial sensors with gps for vehicle dynamics control*, Journal of Dynamic Systems, Measurement, and Control, 2004.

[RGW09] Hui Rong, Jinfeng Gong, and Wulin Wang, *Kinematics model and control strategy of adaptive front lighting system*, Proc. Second International Conference on Intelligent Computation Technology and Automation ICICTA '09, vol. 2, 2009, pp. 70–74.

[Ros03] R. Roslak, *A corparative study of mobile ad hoc networks and autonomois systems for collective illumination of the traffic space*, 7th World Multiconference on Systemics and Informatics, 2003.

[Ros05] Jacek Roslak, *Entwicklung eines aktiven scheinwerfersystems zur blendungsfreien ausleuchtung des verkehrsraums*, Ph.D. thesis, Fakultaet fuer Maschinenbau, Universitaet Paderborn, 2005.

[RS05] S. Rezaei and R. Sengupta, *Kalman filter based integration of dgps and vehicle sensors for localization*, Proc. IEEE International Conference Mechatronics and Automation, vol. 1, 2005, pp. 455–460 Vol. 1.

[RS07] ———, *Kalman filter-based integration of dgps and vehicle sensors for localization*, vol. 15, 2007, pp. 1080–1088.

[RSK08] M. Rockl, T. Strang, and M. Kranz, *V2v communications in automotive multi-sensor multi-target tracking*, Vehicular Technology Confernce. IEEE 68th, 2008.

[RW03] J. Roslak and J. Wallaschek, *Aktive kfz-lichtverteilungen zur kollektiven ausleuchtung des verkersraumes*, 1. Paderborner Workshop Intelligente Mechatronische Systeme, 2003.

[RW04] ______, *Active lighting systems for improved road safety*, Intelligent Vehicles Symposium, 2004.

[Rze03] George Rzevski, *On conceptual design of intelligent mechatronic systems*, 1029 – 1044.

[SAMF03] John M. Sullivan, Go Adachi, Mary Lynn Mefford, and Michael J. Flannagan, *High-beam headlamp usage on unlighted rural roadways*, Tech. report, The University of Michigan Transportation Research Institute, 2003.

[Sas07] Y. Sasaki, *The truth of the f-measure*, 2007.

[SAW08] Hatem Shadeed, Essam Abas, and Joerg Wallaschek, *Sensor data fusion platform for light based driver assistance systems*, 5th International Workshop on Intelligent Transportation, 2008.

[SC09] Apiwat Sangnoree and Kosin Chamnongthai, *Robust method for analyzing the various speeds of multitudinous vehicles in nighttime traffic based on thermal images*, Proc. Fourth International Conference on Computer Sciences and Convergence Information Technology ICCIT '09, 24–26 Nov. 2009, pp. 467–472.

[SDA06] Michael Stamatelatos, Homayoon Dezfuli, and George Apostolakis, *A proposed risk-informed decision-making framework for nasa (psam-0368)*, ASME Press, New York, NY, 2006.

[SGKN10] Z. Shen, J. Georgy, M. Korenberg, and A. Noureldin, *Enabling accurate low cost positioning in denied gps environments with nonlinear error models of inertial systems*, CGC10, 2010.

[She03] Phillip H. Sherrod, *Dtreg predictive modeling software*, Tech. report, 2003.

[SHS$^+$06] H. Shadeed, T. Hesse, S. Strauss, J. Wallaschek, and M. Goetz, *Concept of an active front-lighting driver assistance system*, IEEE 4th International Conference on Information & communication Technology, 2006.

[Sin84] R. W. Sinnott, *Virtues of the haversine*, Sky and Telescope 68, 1984.

[SIS$^+$07] M. Sato, M. Izumi, H. Sunahara, K. Uehara, and J. Murai, *Threat analysis and protection methods of personal information in vehicle probing system*, Proc. Third International Conference on Wireless and Mobile Communications ICWMC '07, 2007, pp. 58–58.

[SKK$^+$97] A. Siadat, A. Kaske, S. Klausmann, M. Dufaut, and R. Husson, *An optimized segmentation method for a 2d laser-scanner applied to mobile robot navigation*, 3rd IFAC Symp. Intelligent Components and Instruments for Control Applications, 1997.

[Sko89] Merrill I. Skolnik, *Radar handbook (2nd edition)*, Mcgraw-Hill Professional, 1989.

[SM04] James R. Sayer and Mary Lynn Mefford, *High visibility safety apparel and nighttime conspicuity of pedestrians in work zones*, Journal of Safety Research **35** (2004), no. 5, 537 – 546.

[SRF+08] A. Sergio, F. Rodriguez, V. Fremont, , and P. Bonnifait, *Extrinsic calibration between a multi-layer lidar and a camera*, IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems MFI 2008, 2008, pp. 214–219.

[SSS07] T. Schon, B. Sick, and M. Strassberger, *Hazard situation prediction using spatially and temporally distributed vehicle sensor information*, Proc. IEEE Symposium on Computational Intelligence and Data Mining CIDM 2007, 2007, pp. 261–268.

[Str05] S. Strauss, *Lichtsysteme mit led lichtquellen*, LED in der Lichttechnik, Haus der Technik, 2005.

[SV99] Carl Shapiro and Hal R. Varian, *Information rules*, Harvard Business Press. ISBN 087584863X, 1999.

[SW06] H. Shadeed and J. Wallaschek, *Vehicle front-lighting assistance system to improve comfort and safety of the driver*, IEEE 4th International Conference on Information and communication Technology, 2006.

[SW07a] ______, *Concept of an intelligent adaptive vehicle front-lighting assistance system*, Proceedings of 2007 IEEE Intelligent Vehicle Symposium, 2007.

[SW07b] ______, *On intelligent adaptive vehicle front-lighting assistance systems*, Proceedings of The 10th International IEEE Conference on Intelligent Transportation Systems, 2007.

[SWG+09] JÃŒrgen Seekircher, Bernd Woltermann, Axel Gern, Reinhard Janssen, Dirk Mehren, and Martin Lallinger, *The car learns how to see - camera-based assistance systems*, 2009.

[TA00] Turner and Austin, *Sensors for automotive telematics*, UK: Meas. Sci.Technol., 2000.

[TDH94] R. Taktak, M. Dufaut, and R. Husson, *Vehicle detection at night using image processing and pattern recognition*, Proc. ICIP-94. IEEE International Conference Image Processing, vol. 2, 13–16 Nov. 1994, pp. 296–300.

[TLH04] Q. M. Tian, Y. P. Luo, and D. C. Hu, *Pedestrian detection in nighttime driving*, Proc. Third International Conference on Image and Graphics, 18–20 Dec. 2004, pp. 116–119.

[TML03] J Trommershauser, L T Maloney, and M S Landy, *Statistical decision theory and the selection of rapid, goal-directed movements*, J. Opt. Soc. Am. A (2003), no. 20, 1419–1433.

[Tra06] *Traffic safety facts*, 2006.

[Tra07] A. Traechtler, *Mechatronics systems in automotive industry*, Tech. report, Regelungstechnik und Mechatronik, Paderborn University, 2007.

[TRLZ08] Tuan Hue Thi, Kostia Robert, Sijun Lu, and Jian Zhang, *Vehicle classification at nighttime using eigenspaces and support vector machine*, Proc. Congress on Image and Signal Processing CISP '08, vol. 2, 27–30 May 2008, pp. 422–426.

[TsBM99] C Taylor, J seck, R Blasi, and J Malik, *A comparative study of vision-based lateral control strategies for autonomous highway driving*, International Journal of Robotics Research (1999).

[TT07] Feng Tang and Hai Tao, *Fast multi-scale template matching using binary features*, In IEEE WACV, 2007.

[Vis08] General Vision, *Cognimem reference guide - parallel neural network for pattern recognition*, 2008.

[Voe05a] S. Voelker, *Light emitting diodes - basics and applications*, Tagung der IQPC GmbH, 2005.

[Voe05b] ______, *Werden zukünftige scheinwerfer mehr blenden?*, Tagung der IIR Deutschland GmbH, 2005.

[Vog07] Joerg Vogeler, *Entwurf und implementierung eines objekterfassungssystems im bezug auf cognisight kamera-bausatz*, Master's thesis, Hochschule Fulda, 2007.

[WF01] V Willhoeft and K Fuerstenberg, *Quasi-3d scanning with laserscanners*, 8th World Congress on Intelligent Transport Systems, 2001.

[WFD01] V. Willhoeft, K. Ch. Fuerstenberg, and K. C. J. Dietmayer, *New sensor for 360° vehicle surveillance*, IEEE Intelligent Vehicles Symposium, 2001.

[WHF05] Chun-Che Wang, Shih-Shinh Huang, and Li-Chen Fu, *Driver assistance system for lane detection and vehicle recognition with night vision*, Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005), 2005, pp. 3530–3535.

[WKE03] J. Wallaschek, R. Kauschke, and K. Eichhorn, *Aktive lichtsysteme für das kraftfahrzeug*, 5. Mechatroniktagung Fulda, 2003.

[WLKC07] Wei-Yen Wang, Ming-Chih Lu, Hung Lin Kao, and Chun-Yen Chu, *Nighttime vehicle distance measuring systems*, vol. 54, 2007, pp. 81–85.

[WLS06] J. Wallaschek, J. Locher, and S. Strauß, *Lichttechnische fahrassistenz.*, ATZ Automobiltechnische Zeitschrift, 2006.

[WSSY02] H. Wu, M. Siegel, R. Stiefelhagen, and J. Yang, *Sensor fusion using dempster-shafer theory*, Proceedings of IEEE IMTC, 2002, pp. 21–23.

[WWBH07] Joerg Wallaschek, Burkard Woerdenweber, Peter Boyce, and Donald Hoffman, *Automotive lighting and human vision*, Springer Berlin Heidelberg, 2007.

[YTK+08] A. Yamasaki, Hidenori Takauji, S. Kaneko, T. Kanade, and H. Ohki, *Denighting: Enhancement of nighttime images for a surveillance camera*, Proc. 19th International Conference on Pattern Recognition ICPR 2008, 2008, pp. 1–4.

[Zha00] Zhengyou Zhang, *A flexible new technique for camera calibration*, Transactions on Pattern Analysis and Machine Intelligence IEEE, November 2000, pp. volume 22, issue 11, Nov 2000, pp 1330–1334.

[ZJ95] Weihua Zhuang and J.Tranquilla, *Modeling and analysis for the gps pseudo-range observable*, vol. 31 Issue:2, 1995, pp. 739 – 751.

[ZMQO10] Yi Zhou, Abedalroof Mayyas, Ala Qattawi, and Mohammed Omar, *Feature-level and pixel-level fusion routines when coupled to infrared night-vision tracking scheme*, Infrared Physics & Technology **53** (2010), no. 1, 43 – 49.

[ZTL05] Bin Zhang, Qiming Tian, and Yupin Luo, *An improved pedestrian detection approach for cluttered background in nighttime*, Proc. IEEE International Conference on Vehicular Electronics and Safety, 14–16 Oct. 2005, pp. 143–148.

Curriculum Vitae

Personal Data

Name: Hatem Shadeed
Date of Birth: 10th October, 1976
Email: h.shadeed@gmail.com
Place of Birth: Egypt
Marital Status Married

Education and Qualifications

April 2005 to Date	**Doctoral Candidate in L-LAB (Public-Private Cooperation between Paderborn University and Hella KGaA Hueck & Co.)**
Oct 2002 to Oct 2003	**Paderborn University, Germany** Degree: ***Master of Engineering in applied mechatronics***
Sep 2001 to Oct 2002	**Information Technology Institute, Egypt** Degree: ***Diploma in applied mechatronics***
Sep 1995 to Mai 1999	**Zagazig University, Egypt** Degree: ***Bachelor in mechanical engineering***

Work Experience

Jun 2008 to Date	**Hella - Aglaia Mobile Vision GmbH, Berlin, Germany** Position: ***Development Engineer and System Integrator***
Oct 2003 to April 2005	**Hella - L-LAB Lighting Center, Paderborn, Germany** Position: ***Assistance Student***

Additional Information

Awards:

- 2004: Hella KGaA fellowship to study PhD in Dynamic Intelligent Systems
- 2002: Information Technology Institute fellowship to study diploma in mechatronics

---

Berlin, 14th November 2012