DISSERTATION

submitted
to the
Combined Faculty for the Natural Sciences and Mathematics
of
Heidelberg University, Germany
for the degree of
Doctor of Natural Sciences

Put forward by
Diplom-Ing. José Domingo Esparza García
Born in: Fuente Álamo de Murcia, Spain

DISSERTATION

submitted
to the
Combined Faculty for the Natural Sciences and Mathematics
of
Heidelberg University, Germany
for the degree of
Doctor of Natural Sciences

Put forward by
Diplom-Ing. José Domingo Esparza García
Born in: Fuente Álamo de Murcia, Spain

Oral examination: _____

3D Reconstruction for Optimal Representation of Surroundings in Automotive HMIs, Based on Fisheye Multi-camera Systems

Advisor: Prof. Dr. Bernd Jähne
Prof. Dr. Katja Mombaur

Abstract

The aim of this thesis is the development of new concepts for environmental 3D reconstruction in automotive surround-view systems where information of the surroundings of a vehicle is displayed to a driver for assistance in parking and low-speed manouvering.

The proposed driving assistance system represents a multi-disciplinary challenge combining techniques from both computer vision and computer graphics. This work comprises all necessary steps, namely sensor setup and image acquisition up to 3D rendering in order to provide a comprehensive visualization for the driver.

Visual information is acquired by means of standard surround-view cameras with fish eye optics covering large fields of view around the ego vehicle. Stereo vision techniques are applied to these cameras in order to recover 3D information that is finally used as input for the image-based rendering. New camera setups are proposed that improve the 3D reconstruction around the whole vehicle, attending to different criteria. Prototypic realization was carried out that shows a qualitative measure of the results achieved and prove the feasibility of the proposed concept.

Zusammenfassung

Zielsetzung der vorliegenden Arbeit ist die Darstellung neuer Konzepte zur dreidimensionalen Rekonstruktion der Umwelt für Surround-View Systeme die im Automotive Bereich eingesetzt werden. Dazu wird dem Fahrer Umgebungsinformation angezeigt die ihn bei Park- und anderen Fahrmanövern um niedrigen Geschwindigkeitsbereich unterstützt.

Das vorgeschlagene Fahrassistenzsystem stellt eine interdisziplinäre Herausforderung dar und kombiniert Techniken aus den Bereichen *Computer Vision* und *Computer Graphics*. Diese Arbeit umfasst alle notwendigen Schritte, nämlich Sensor-Setup und Bildaufnahme bis zu 3D- Rendering, um eine umfassende Visualisierung für den Fahrer zu bieten.

Visuelle Information wird mit Hilfe von Standard Surround-View-Kameras mit Fischaugenoptik erzeugt, welche das Sichtfeld rund um das Fahrzeug abdeckt. Für diese Kameras werden Stereo-Vision-Techniken angewendet, um 3D-Informationen zu erhalten, die schließlich als Input für das bildbasierte Rendering verwendet werden. Neue Kamera-Setups werden unter Berücksichtigung verschiedener Kriterien vorgeschlagen, welche die 3D-Rekonstruktion rund um das ganze Fahrzeug verbessern. Im Rahmen dieser Arbeit wurden prototypische Umsetzungen realisiert, die eine erste quantitative Abschätzung der Leistungsfähigkeit der beschriebenen Verfahren erlauben und die Machbarkeit des vorgeschlagenen Konzepts beweisen.

Acknowledgements

I would like to thank all people who contributed to this thesis. In first place I would like to thank Prof. Dr. Bernd Jähne, from the Heidelberg Collaboratory for Image Processing, for his supervision and support during this period. His experience and advice have been extremely helpful not just regarding research, but also in solving organizational issues.

Secondly, many thanks to the Robert Bosch GmbH for its financial and logistic support during these three years. Of all the people within the organization who directly or indirectly contributed to the completion of this thesis, my warmest thank you goes to Dr. Michael Helmle for his continuous support and the uncountable amount of time that he dedicated to supervise this work. His experience as a researcher as well as a member of the industry have proven extremely helpful for the definition of research topics and as guideline to keep working on the right track. Big thanks also to my colleagues and friends Leo Vepa and Raphael Cano. Leo's continuous training for my survival as a foreigner in Germany and his unique ability to relativize things have been largely motivating since our first encounter years ago. Raphael's pragmaticity and wise advice have prevented me from falling into complexity holes which may have been critical for my work. A further thank you goes to Carsten Bregenzer and László Anka for making these years more enjoyable - to Carsten for the many coffees together and to László for those minutes looking at time go by.

In third place I woule like to acknowledge the help of my family at the time of deciding whether starting a PhD was the right thing to do. To my sisters Cristina (for encouraging me) and Raquel (for discouraging me), because all views are equally important in the process of making a decision. Also to my parents, for making sure that quitting was never an option. To all of them, thank you.

Finally, the biggest thank you goes to Katrin. For dealing with my temper, mood and stress throughout the years. Her rational and organized understanding of life are an enourmous source of motivation and focus in times where my impulsivity comes into view. Without her, the way to the completion of this work would have felt much longer.

Contents

El que no llora, no mama[1]

Brought to me by my father

Keep it simple

Raphael Cano

Chapter 1

Introduction

Driver assistance functions aim to support the driver of a vehicle during maneuvering. Among others, the support can be designed by means of displaying a comprehensive model of the near range environment to the driver, in order to avoid collision with obstacles not visible in the current path of view of the driver.

The aim of these systems is to improve the perception and understanding that the driver has about the surrounding of the vehicle. In order to achieve this, depth estimation is one of the big challenges that the automotive industry currently tries to solve. This is commonly achieved by means of a combination of sensors capable of delivering, directly or indirectly, distance measurements within a given field of view.

Typical sensors that are used in driving assistance include, among others, ultrasonic, radar or video sensors. In particular, the use of cameras with fisheye optics has gained extra interest in recent years, due to their very large fields of view together with shrinking costs and sizes, which make them an optimal choice for being mounted on vehicles while being in concordance with the exterior design of these.

1.1 Motivation

In recent years, the automotive industry has focused, in the context of driving assistance, in the need to reduce the number of avoidable accidents, which cause large amounts of damage every year.

The reason for many of these accidents is the driver's lack of information about the surroundings of the vehicle. Especially for big-sized vehicles, where large blind areas for

[1] *No pain, no gain.* The author acknowledges that the meaning is not completely equivalent, but not being a native English speaker makes it extremely difficult to express so much, so simply.

the driver exist, systems that can support the manouvering have proven to be of huge importance.

One of the many ways in which a system can be designed to aid a driver while maneuvering is to offer a visualization of the vehicle's surrounding that contains real visual information about areas which are not within the visual field of view of the driver.

Shrinking sizes and costs of sensors have made it possible for automotive-qualified cameras to reach large resolutions. This opens the door for more advance image processing and computer vision techniques to be applied on real products.

In this context, the opportunity to design a system that can exploit existing sensor setups to enhance safety and user experience was the motivation for this thesis. In particular, the possibility to perform stereo measurements with automotive fisheye surround view cameras to gain 3D spatial information has been analyzed. Furthermore, alternative configurations to current systems have been proposed that can benefit from an extension of the number of cameras, and depth-based enhancements on the graphical visualization have been discussed.

1.2 Contributions

The author considers the following to be the main contributions of this thesis:

- Introduction of the idea to perform stereo measurements based on standard surround view camera configuration. An analysis of the main challenges in the process was conducted, with results published in [Esparza et al., 2014b].
- Proposal of new camera setups, where design criteria was with focus on 3D reconstruction. Two new camera mounting configurations have been proposed to overcome some of the problems initially spotted.
- Evaluation of the proposed camera configurations attending to field of view around the ego vehicle and accuracy of depth measurements. These experiments led to the publication of one conference paper: [Esparza et al., 2014a].
- Concept for a ground truth scheme based on a lidar sensor. A registration method was designed in order to be able to represent 3D measurements from the stereo approach and from the lidar on a common reference frame. In cooperation with other colleagues, a conference paper was published with evaluation on the robustness of the approach: [Esparza et al., 2014c].

- Visualization enhancements were proposed in the context of automotive surround view systems, based on depth measurements.

1.3 Outline

This thesis is organized in 8 chapters including the introduction. Chapter 2 presents an introduction into automotive park and maneuvering systems with special focus on surround view systems. Existing systems are described and some of their limitations are pointed out with a review of possible solutions.

Chapter 3 provides an introduction to the foundations of computer vision and computer graphics used throughout this thesis. Additionally, a special section is dedicated to conventions on coordinates systems.

The main contributions of this thesis are contained in Chapters 4, 5 and 6. In particular, Chapter 4 introduces the reference sensor considered for ground truth generation. A novel method is presented for the registration of the reference sensor with respect to multi-camera systems and an error analysis is conducted on the robustness of the registration process.

Chapter 5 describes the methodology developed to perform 3D measurements with surround view systems. Different camera mounting configurations are analyzed with respect to different criteria. Evaluation and results are also presented.

Chapter 6 describes both the complete algorithmical processing required to give an interpretation of the 3D data acquired by means of stereo vision, and the approaches utilized for enhanced visualization. These enhancements rely on the depth information obtained by means of the methods presented in Chapter 5, but are compatible with occupancy information provided by means of other automotive sensors.

In Chapter 7 results are discussed and conclusions are elaborated. Open points and possible lines of research for future work are also described. Finally, Chapter 8 summarizes the work presented and ends this thesis.

Chapter 2

Automotive Park and Maneuvering Systems

Automotive Driver Assistance (DA) functions aim to support the driver of a vehicle during maneuvering. The support can be designed in the following ways:

- Displaying the near range environment to the driver in order to avoid collision with obstacles not visible in the current path of view of the driver.
- Taking over of some of the driver's activities in order to increase the comfort during maneuvering.
- Supervision of the driver's activities and intervention in dangerous situations.
- Automated driving without requiring to have a driver onboard the vehicle.

Park and maneuvering (PM) systems represent, within the DA family, those driving functionalities which are targeted at low speed maneuvering and aim to provide environment information to the driver in a way that is intuitively understandable. In particular, speeds on the order of 10-20 $^{km}/_{h}$ are a good reference of the maneuvering velocities for which these systems are usually designed. For parking space survey, these may reach up to $40^{km}/_{h}$, approximately.

Given the low speeds under consideration, mainly the very near vicinity of a vehicle is of high relevance for the PM systems. Distance ranges up to 10 meters from the vehicle are usually sufficient to fulfill most of the requirements of these systems. The field of view coverage is, however, a more critical aspect of the system design. A complete 360° horizontal field of view is desirable since every direction in the vicinity is relevant in these scenarios. In this sense, PM systems differ largely from other DA functionalities

targeting higher speeds, which normally focus on much narrower FoVs. Furthermore, during low speed maneuvering narrow drive paths or parking spaces are quite common, which pose very specific challenges.

Despite the low maneuvering speeds, a very fast reaction time is required in maneuvering situations. This requires the system to operate with as low a latency as possible, especially on visualization tasks. On top of functional requirements, severe restrictions regarding the number and size of sensors exist in the automotive industry since these determine costs for the end customer.

2.1 Surround View Systems

Surround view systems represent a subgroup within PM systems aimed at displaying a visualization of the surrounding of a vehicle through the vehicle's internal HMI, based on real camera images. The aim of these systems is to support the driver by displaying areas in the vicinity of the vehicle that are out of reach of his path of view.

Surround view functionalities were initially adopted by the automotive industry in the form of single rear-view cameras. These early functionalities allowed the displaying of a video stream from a camera which was mounted on the rear end of a vehicle, at a height lower than the field of view of the driver. In this manner, blind spots behind the vehicle could be imaged and displayed to the driver.

The video streams acquired by means of these cameras could be simply visualized on the internal display of the vehicle, in order to support the driver in more or less complex maneuvering tasks, like reverse parking. As additional support, extra information is usually added in the form of overlays, eg. reference steering lines. Despite the simplicity of the system, it has proved to give a large degree of support for the driver.

In recent years, surround view systems have been extended with additional cameras that allow to cover a larger field of view around the vehicle. The following sections offer an overview of the standard camera configurations employed in surround view systems and introduce the Bird-View visualization, with a review on its strengths and weaknesses [Liu et al., 2008].

2.1.1 Standard Camera Configuration for Surround View Systems

A common layout is to have a camera mounted on each side of the vehicle, usually in the side mirrors, one on the front of the vehicle hidden in the grill and one in the tail

gate or close to it. The mounting position of these cameras is optimized in order to cover close to 360° of the near range vehicle surrounding and to be concord with the vehicle exterior design. Given the large fields of view provided by cameras with fisheye optics [Hughes et al., 2009], this can be achieved with a setup of just four cameras in the described configuration. A common camera setup is depicted in Figure 2.1, where the field of view of each camera is highlighted.

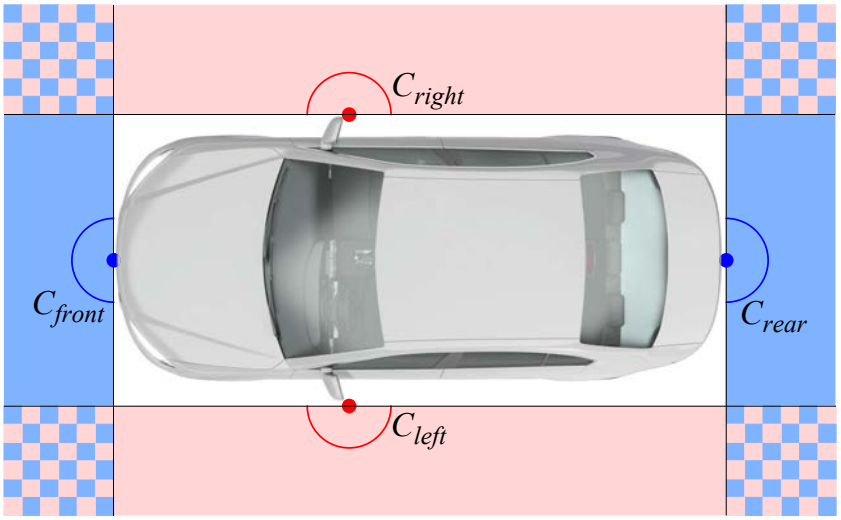


FIGURE 2.1: Standard camera configuration. Four fisheye cameras are mounted on the vehicle in a configuration which commonly allows for 360° visualization of the vehicle's surroundings. The field of view of each camera is assumed equal to 180° horizontally and is represented by a uniform color in this figure. There also exist overlaps on the fields of view of the cameras, that can be used for 3D stereo reconstruction.

2.1.2 2D Bird-View Systems

In order to improve system ergonomy, one characteristic that is desirable for surround view systems is that the visualization includes the vehicle itself, or a model of it, such that an outsider perspective can be achieved. A common example is a Bird-View visualization.

This visualization projects images from the four surround-view cameras on the ground plane, and creates a composite view as seen from a virtual camera positioned at a certain height above the scene, combined with a top-view model of a vehicle. This creates the impression of a camera which is flying on top of the vehicle, on a fixed position about it.

The generation of a bird-view visualization is based on the assumption of a "flat world". Under this assumption, the world is supposed to be flat, thus providing that all objects which are imaged by a camera lie on the same plane. Assuming calibrated cameras, the original images can be projected onto this plane, allowing for a change of perspective by means of a virtual camera. The virtual camera is usually positioned fixed on top of the vehicle, although there is no real restriction in this sense.

An description of the generation of a bird-view is depicted in Figure 2.2.

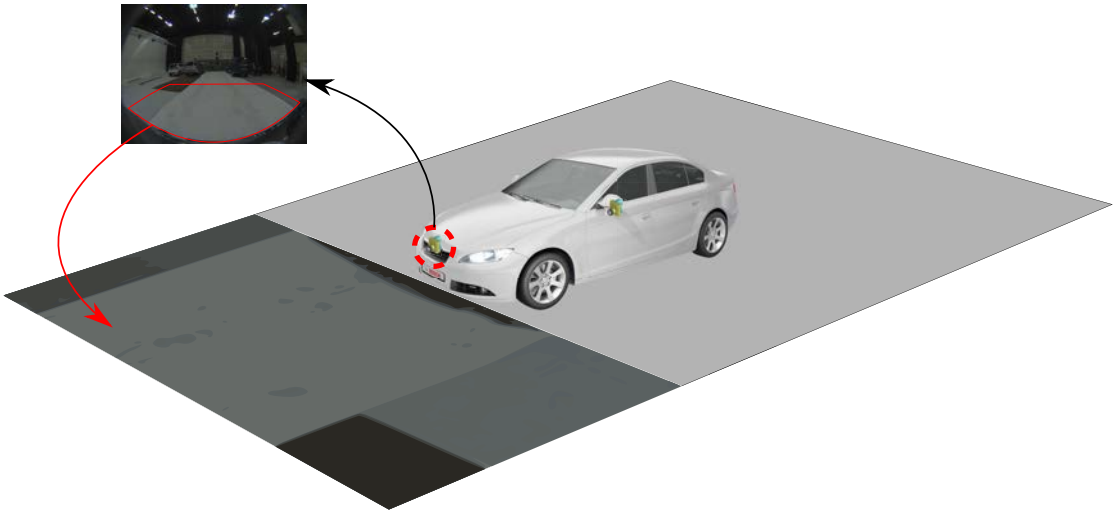


FIGURE 2.2: Bird-view generation. The original images are projected onto the ground plane by considering the intrinsic camera model and the extrinsic calibration. A virtual camera can be positioned within the scene to achieve the desired bird-view.

Figures 2.3, 2.4 and 2.5 show how the virtual view is composed from the real images for different example scenarios.

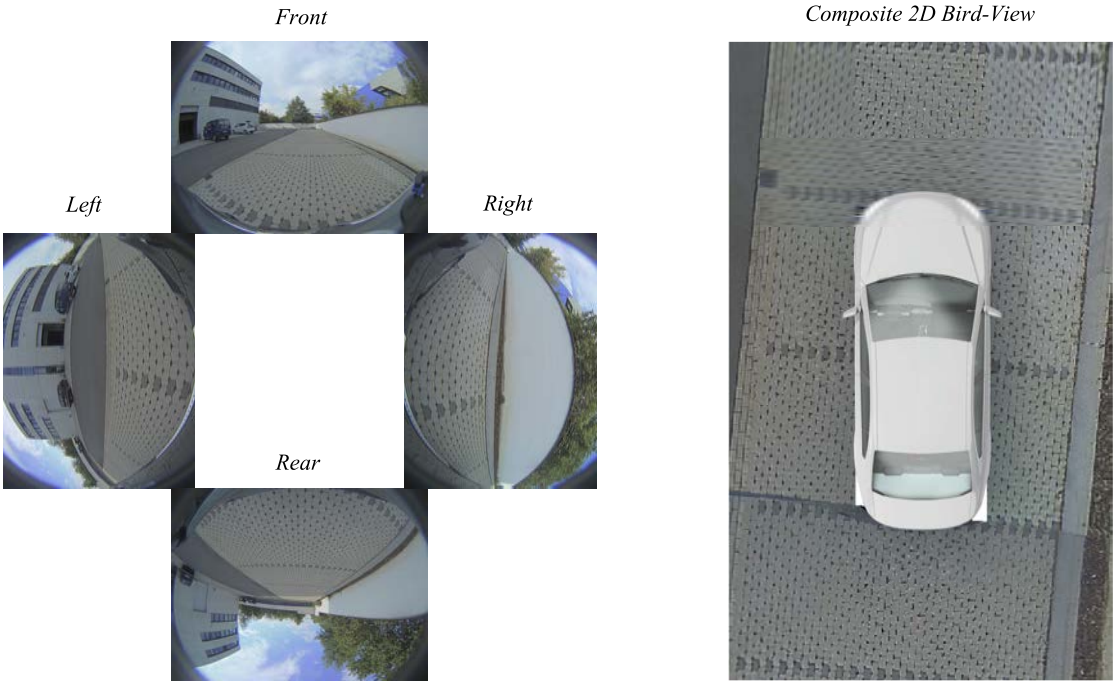


FIGURE 2.3: Bird-view composition from original fisheye images - Scene 1. Left: Original fisheye images as acquired by the cameras. Right: Composite view after projecting the images onto the ground plane.

FIGURE 2.4: Bird-view composition from original fisheye images - Scene 2. Left: Original fisheye images as acquired by the cameras. Right: Composite view after projecting the images onto the ground plane.

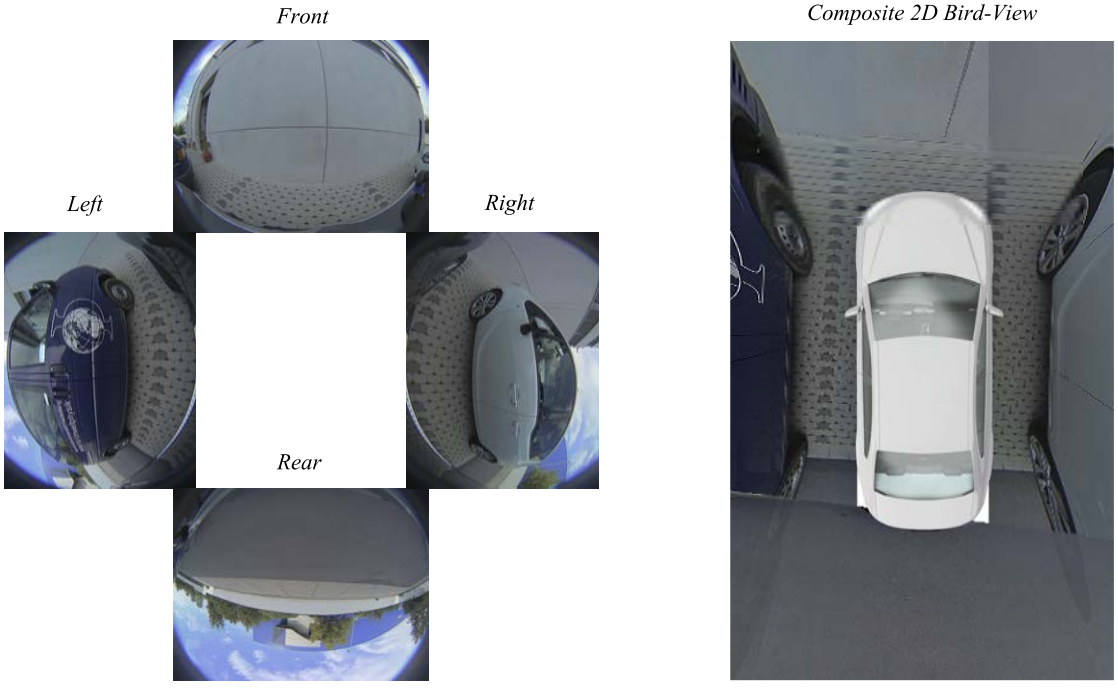


FIGURE 2.5: Bird-view composition from original fisheye images - Scene 3. Left: Original fisheye images as acquired by the cameras. Right: Composite view after projecting the images onto the ground plane.

Although for certain parking situations a bird-view may provide with good support for a driver, there are certain limitations to the system. Most of these limitations are well

understood and continuous research on the field aims at tackling them in the best way possible. In the following, some known weaknesses of the system are presented together with solutions that have been previously proposed in order to overcome them.

2.1.2.1 Narrow Field of View

Given the relatively low mounting position of the surround-view cameras, there are some restrictions regarding the change of perspective that is achievable between real and virtual bird-view cameras even when the flat world assumption is approximately valid, eg. when no other vehicles or obstacles are present in the vicinity of the ego vehicle. In particular, due to the finite resolution of the cameras, only a reduced fraction of the imaged field of view can be projected on the floor. In this way, large changes of pixel size between real and virtual images can be avoided, which is usually a common requirement.

In Figure 2.6 a situation is depicted where the spatial resolutions of the real and virtual cameras can be compared.

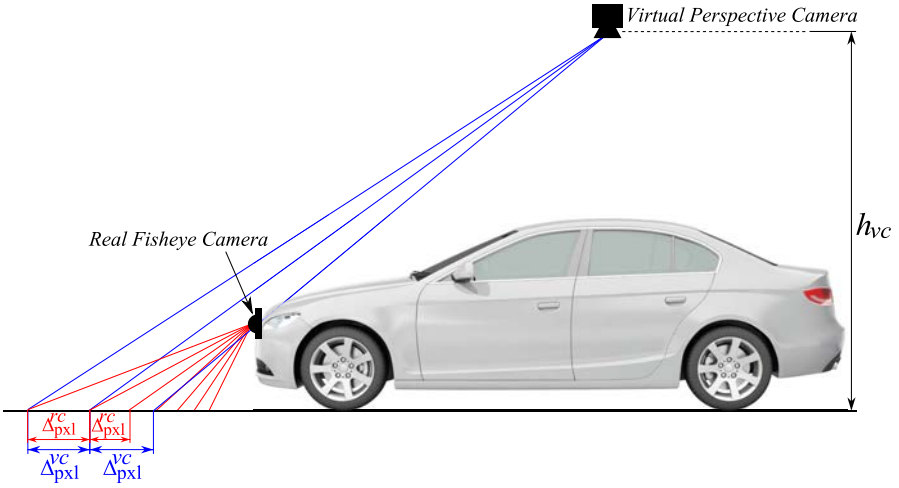


FIGURE 2.6: Comparison of spatial resolution for real and virtual cameras. While the virtual perspective camera has a uniform spatial resolution over the ground plane (assuming it parallel to its image plane), the resolution of the real vehicle fisheye cameras decreases with the distance. Ideally, the field of view of the bird view should be restricted to the areas around the vehicle where both resolutions are comparable, in order to avoid large changes of pixel size. Blue and red rays represent a constant amount of pixels for the virtual and real cameras, respectively.

In the depicted situation, the virtual image plane and ground plane are parallel, thus the spatial resolution of the virtual camera is uniform over the whole field of view. For the real vehicle cameras, however, the spatial resolution decreases with the distance (in the far range, every pixel covers a very large floor area).

Figure 2.7 evaluates a one-dimensional model of the spatial resolution function (in $[pel/m]$) for both real and virtual cameras. This shows, for different virtual camera heights, what the area around the vehicle is that ensures a minimum real resolution on the floor plane equal to the resolution of the virtual camera.

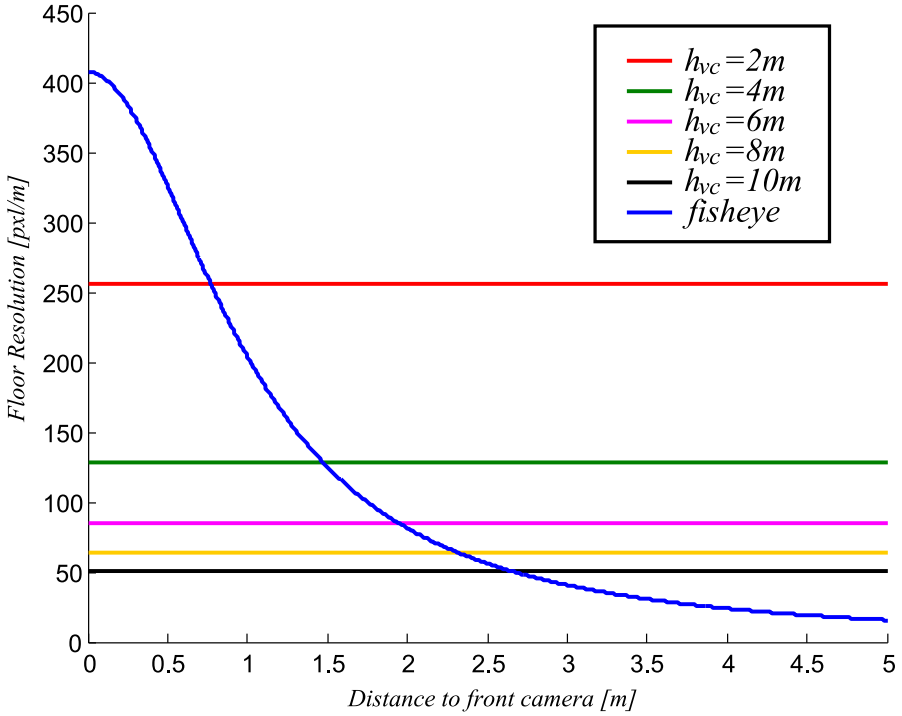


FIGURE 2.7: Spatial resolution on the ground plane with respect to the virtual camera height. Calculated for a perspective virtual camera covering a field of view of 90° and an image width of 1024 pixels, and a vehicle fisheye camera mounted 1 meter above the floor with a FoV of 180° and image width 1280 pixels. The constant resolution is due to parallel image and ground planes. In blue: spatial resolution of the real fisheye camera over the ground plane. Decreasing spatial resolution with respect to the distance can be observed.

From the graph in Figure 2.7 it can be observed that even for very high virtual cameras ($\sim$ 10 meters) the maximum coverage around the vehicle is reduced to under 3 meters to avoid large changes in pixel size. Considering larger FoVs would incur in fisheye pixels being stretched over very large surfaces. As an addition to the reduced field of view, new problems arise for elevated objects (where the flat world assumption is not met). These effects are described in detail in the next section.

2.1.2.2 No Elevated Objects

As a result of the flat world assumption used in the bird-view generation, all objects are visualized as projections on the floor. The shape of the projection corresponds to the shadow that would be casted by a light source situated exactly on the actual camera

position. Additionally, in a perfectly flat world the horizon is located infinitely far away from the ego vehicle and, therefore, any object with an elevation higher than the camera itself cannot be projected on the floor, thus cannot be present in a virtual perspective view.

As a consequence of both previous characteristics, no accurate height information can be inferred from the bird-view directly. This effect is described in more detail in Figure 2.8.

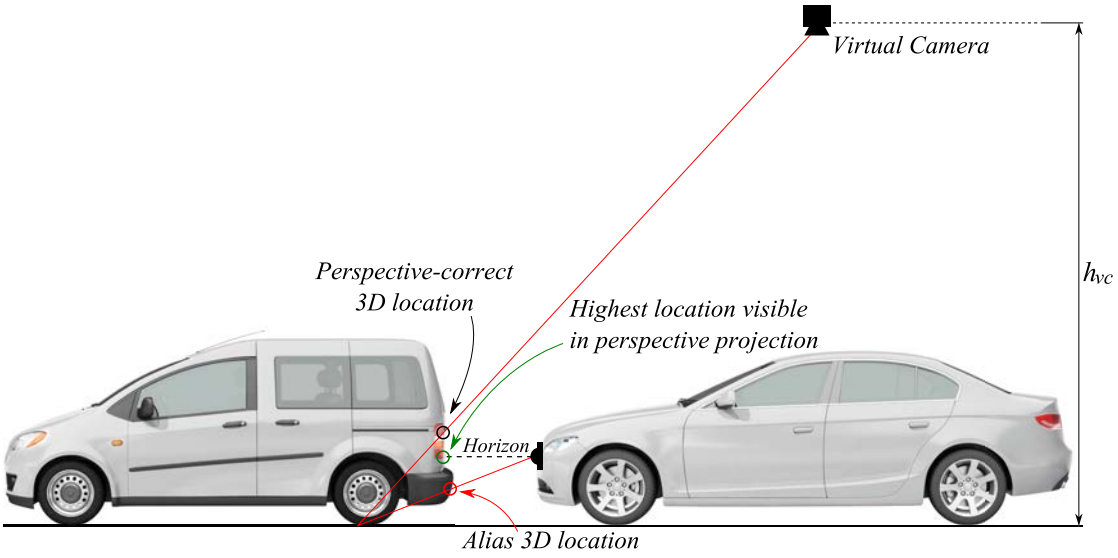


FIGURE 2.8: Restrictions on elevated objects. In situations where the flat-world assumption is not fulfilled, elevated objects find themselves projected on the floor. As a result, texture information is retrieved from alias locations, which are always lower than the real perspective-correct 3D location. As a limit, there is the height of the real camera, which corresponds to the infinitely-far horizon. Any object higher than this limit cannot be visualized on the perspective virtual view.

A solution that has been adopted in some systems in order to include height information in the visualization is to change the flat world assumption for an alternative semi-spherical one, which has a 3D model of the ego vehicle inside it [Shimizu et al., 2010]. A representation of this world is depicted in Figure 2.9. Different models can be considered, based on different basic shapes, eg. circular, elliptical, etc.

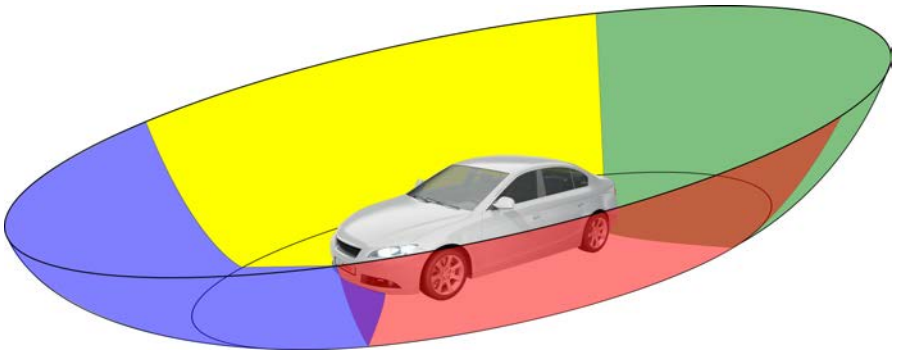


FIGURE 2.9: New projection surface. The images obtained via external cameras are projected onto the semi-spherical projection surface that allows for elevated objects to be seen. The surface is normally divided in as many sectors are cameras are mounted in the car. Each sectors is textured with the corresponding camera image. Each color in this figure corresponds to one of the four cameras (blue: front, red: left, green: rear, yellow: right).

This new assumption allows for virtual displaying of elevated objects, which brings huge benefit with regard to spotting, for example, pedestrians, traffic signs or buildings in the vicinity of the vehicle.

Another benefit of this world assumption is that it opens the door for virtual camera positions different to the above-described birds-view. In particular, elevated objects can be imaged by means of oblique views. The range of possible views that can be selected is unlimited, providing a big degree of flexibility for supporting the driver on meneuvering. Despite the mentioned benefits, there are still problems which this approach cannot avoid. In the following, a review of the most significant limitations of this solution is described:

- Scale magnification factor. This effect occurs when the assumed fixed depth does not fit the real one. Depending on the magnitude of the error, this effect can cause very large scale distortions, as shown in Figure 2.10.

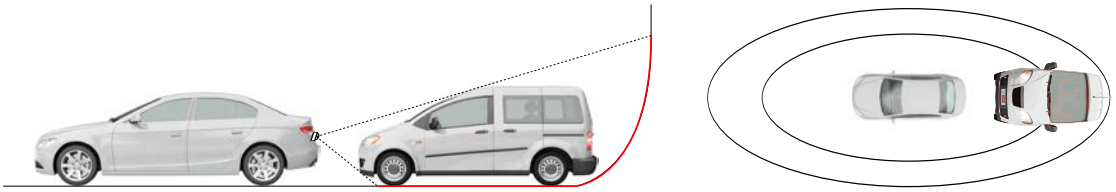


FIGURE 2.10: Given the depth ambiguity resulting from the 2D imaging process, and without any external information, an assumption has to be made about the distance between objects and the camera. Based on the validity of this assumption, a large scaling difference may occur when a real-size model of the vehicle is considered.

- Same object seen multiple times. This effect happens when the real distance to an object is larger than the assumed one and the object is visible in more than

one camera image simultaneously. The areas of the texture corresponding to the same object on different images are used multiple times for rendering, thus multiple instances of the objects are generated. This effect is depicted in Figure 2.11.

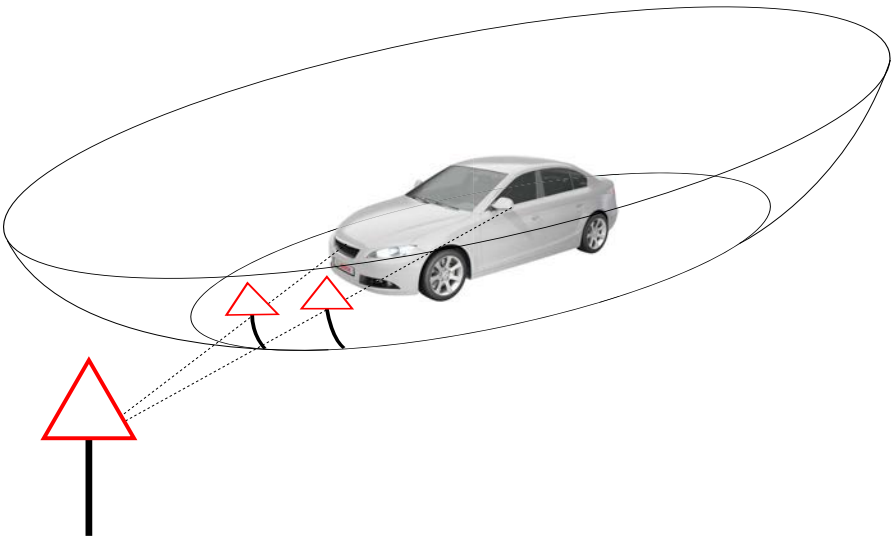


FIGURE 2.11: An object being visible multiple times. When the assumed depth is smaller than the real one, on the areas where the fields of view of more than one camera overlap, more than one instance of the same object may become visible.

- One object is not seen at all. This effect happens when the real distance to an object is shorter than the assumed one. Since the projection surface is divided in areas that correspond to each of the different cameras, there may exist areas of the images that are not used for texturing any geometry. This is depicted in Figure 2.12.

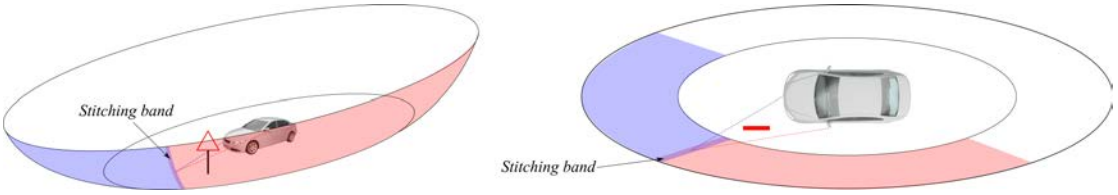


FIGURE 2.12: Disapearing objects. For elevated objects, there is a certain volume in the vicinity of the vehicle that is not projected onto the surface. This effect may happen when the distance to objects is less than the assumed one. Nevertheless, the footprint of the object may still be visible, since at the floor level it can be guaranteed that every location is appropriately textured with images from at least one of the cameras.

These problems are a result of the fact that the shape of the world that is being used as support for generating the virtual view remains an assumption - either flat or spherical, but still an assumption. In order to avoid the kind of problems that have their origin in the incorrectness of these assumptions, real depth measurements would be required. In the next section, a review is made of different approaches that can be considered in order to estimate 3D information in the vicinity of the vehicle.

2.1.3 Depth Information for 3D Surround View Systems

Since depth cannot be directly recovered from a single image, certain approaches are proposed in the following, in order to obtain this information in the context of automotive surround view systems. In particular, it is proposed to gain depth information either by means of external sensors, or by applying computer vision techniques to the surround view images. The next sections explain these approaches in detail. A schematic overview of the processing pipeline required for such a system is depicted in Figure 2.13.

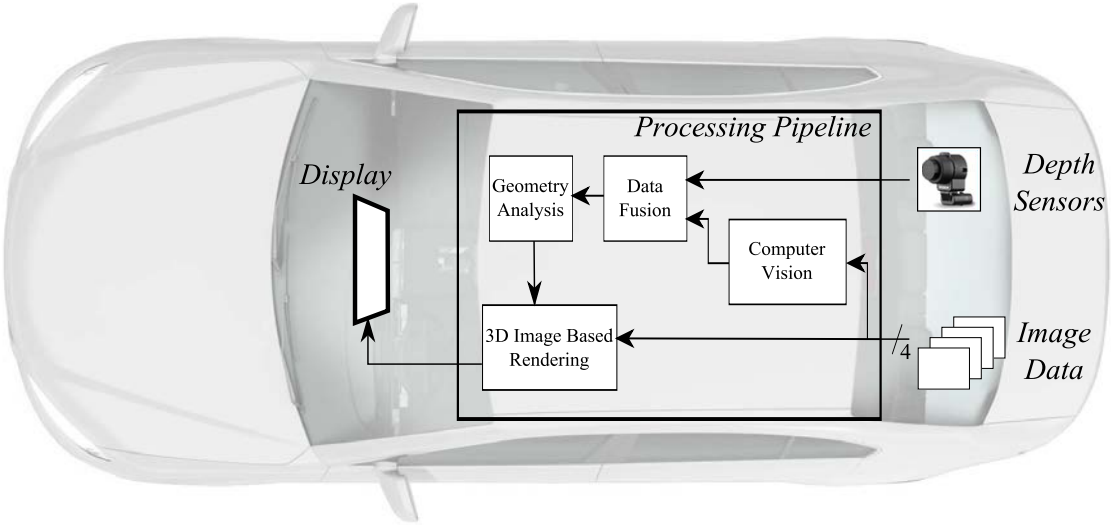


FIGURE 2.13: Depth-based surround view visualization. Depth is obtained by means of Computer Vision and other sensors on a vehicle. Occupancy information is processed and image-based rendering techniques are conducted and shown to the driver by means of a display. Different kinds of sensors in different configurations can be considered. For the visualization tasks in this work, a data fusion layer is assumed that makes the depth analysis independent from the type of sensor.

2.1.3.1 Fusion with External Sensors

The first option proposed to gain 3D depth information is by means of external sensors that already exist on vehicles. In the automotive industry there exist different families of sensors that are capable of measuring distances through different principles.

Two sensors that are commonly installed in vehicles are Ultrasound and Radar sensors. Both sensors work as a combination of actuators and receptors that can detect a reflected signal, ie. sound in the case of USS and electromagnetic waves in the case of radar, and estimate distance based on the time since of flight and the propagation speed on the corresponding medium. These sensors can already be found in most medium and high end vehicles and are, therefore, realistic candidates to support the enhancement of surround view systems.

Another sensor which is lately gaining interest within the automotive community is the 3D laser range finder (LRF). An LRF is capable of gathering 3D information by means of light-based time of flight measurements to reflecting surfaces with a high accuracy. Some models are already available on the market which allow for a full 360° horizontal field of view around the ego vehicle, although multiple alternatives exist that have a narrower FoV.

All these sensors were originally designed with the goal of measuring distances, which makes them a good candidate to be combined with surround-view systems. A sensor data fusion approach is usually required in order to integrate measurements from different sensors. One solution that is commonly used in the field of robotics is the occupancy grid [Elfes, 1989]. An occupancy grid is a discrete spatial representation of the world that contains information about the probability for each spatial element to be occupied by an object. In Section 6.2, a detailed description is given of how an occupancy grid can be utilized to describe geometries on which IBR can be conducted.

In the context of this thesis, focus has been shifted towards depth estimation based on computer vision techniques. The next section introduces the topic and presents a review on previous existing work.

2.1.3.2 Application of Computer Vision Techniques

A second alternative for gaining depth information is to apply computer vision techniques to the different cameras available in the system. The methods mentioned in the following take two different approaches in order to perform depth measurements.

- A first method would be to apply **structure from motion** (SfM) techniques to the cameras on the system. SfM makes use of a sequence of images acquired from a single camera while following a trajectory at different instants in time to build a 3D reconstruction of the scenery, under the assumption that every object was static, except for the camera itself. For every new frame, motion estimation has to be conducted prior to the structure reconstruction phase. This method has some drawbacks, like the need of ego motion before 3D measurements are possible, and reconstruction being possible only up to a scale factor. In surround view systems these translate into uncertainties at system startup (since there is no previous motion) and large latencies due to low maneuvering speeds.
- A different method that can be used in order to gain depth information from the camera images is **stereo vision**. Stereo vision makes use of different images taken simultaneously from different positions with an overlapping field of view. Stereo

vision presents certain benefits with respect to structure from motion, namely no ego motion is required and the 3D reconstruction is scale-correct, among others. Given the independence with respect to ego motions, system latency depends exclusively on the processing steps and no uncertainties are given at system startup.

The core contribution of this thesis is the study of stereo vision applied to surround-view systems. Previous work exists in the field, where authors have also opted for different camera configurations as well as a variety of algorithmical approaches. A review of the related work known to the author is presented in the following.

Several systems have been proposed in literature that make use of different omnidirectional cameras for the purpose of stereo vision. However, cameras with fisheye optics are generally considered a better alternative than catadioptric cameras, due to their large fields of view and compact shape [Hughes et al., 2009].

The work of [Gehrig, 2005] proposed a stereo vision system with fisheye optics which relies on a pin-hole model rectification prior to the feature detection-matching, thus being limited to camera setups that are placed to the left and right of the rear view mirror. In [Gandhi and Trivedi, 2005] the authors used two catadioptric cameras mounted on the mirrors of a vehicle and tried to perform motion-stereo measurements, although results were poor due to low image resolution. A monocular view from each omni camera was obtained on the respecive sides of the car, and stereo matching was applied to consecutive frames from the same camera. The authors did not match features across the camera pair.

In the work of [Suhr et al., 2007], the authors performed detection of free parking spaces by making use of a structure from motion approach, based on fisheye images. They used the camera height to the floor plane as scale reference. The authors further extended their work in [Suhr et al., 2010] by performing a perspective rectification prior to the optical flow computation. A vacant parking slot detection and tracking system that fuses camera and ultrasonic sensor information was presented in [Suhr and Jung, 2014]. The authors detected the parking slots by looking for road markings on a composite bird's eye view and classified them as occupied or vacant, based on ultrasonic measurements. The work of [Unger et al., 2014] introduced a parking assistance system which relies on dense motion-stereo to compute depth maps of the observed environment. By detecting the ground plane, the authors built up silhouettes which limit free space and accumulate them over time. The authors of [Kaempchen et al., 2002] considered a dedicated front stereo camera pair for detection of parking spots and applied a 3D model of a vehicle to the reconstructed data.

The work of [Heng et al., 2013] made use of a 4-camera rig surround view configuration similar to the one considered in this thesis. The aim of the authors was the extrinsic calibration of the camera rig. Although they did not consider the overlap on the fields of view, they performed a time analysis that allowed them to match common features across different camera views based on a local history and motion estimation. In [Knorr et al., 2014], extrinsic calibration of a camera rig was also performed, considering only the overlaps on the fields of view of adjacent cameras.

The work of [Esparza et al., 2014b] was developed as a contribution to this thesis and it first introduced the idea of performing stereo measurements by means of surround view systems. The authors considered a 4-camera setup on a standard configuration and evaluated the amount of existing overlap on the fields of view.

More detailed literature review is provided in following chapters, according to the specific topics.

Chapter 3

Foundations

This chapter presents a comprised collection of concepts which set the base for the work carried out throughout this thesis. In particular, two main sections cover topics related to Computer Vision (CV) and Computer Graphics (CG). Prior to the introduction into CV and CG, a common section is dedicated to describe the convention of coordinate systems used in this thesis.

3.1 Reference Coordinate Systems

In systems where multiple sensors work together, the definition of reference coordinate systems plays an important role for the system description. Throughout this thesis, a global origin of coordinates is considered, which corresponds to a distinctive position on a vehicle. In particular, the norm DIN70000 is considered, which defines the origin of coordinates in the middle of the vehicle rear axis at ground level. The orientation is defined such that the $x-$coordinates are positive in the forward-driving direction and $z-$coordinates are positive in the direction of the floor normal, as shown in Figure 3.1.

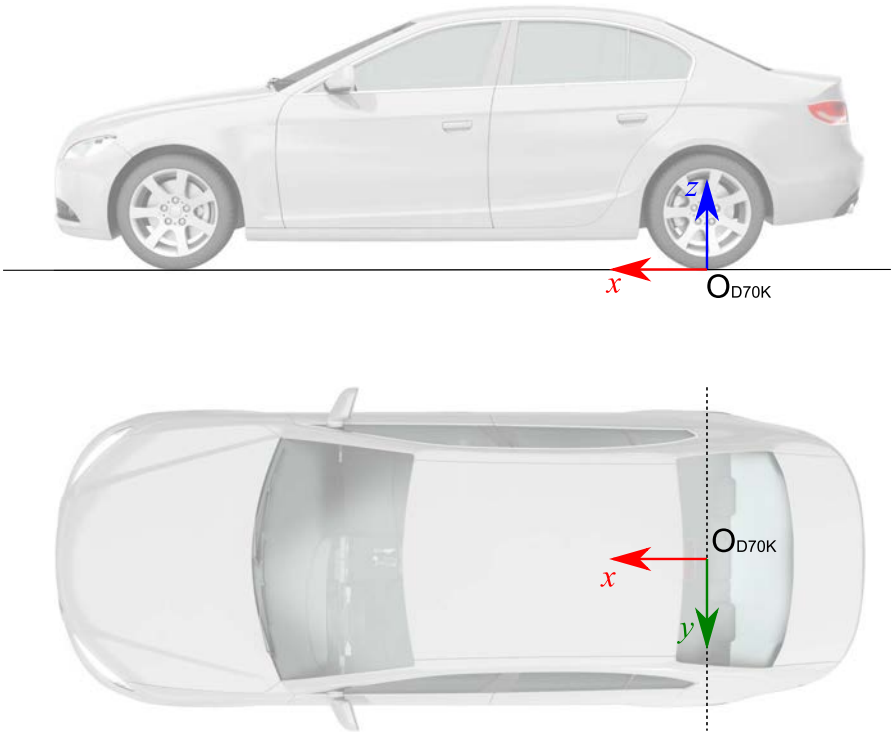


FIGURE 3.1: Global coordinate system DIN70K. The origin of coordinates is defined in the middle of the rear axis, at ground level. The $x-$coordinates are positive in the forward-driving direction and $z-$coordinates are positive in the direction of the floor normal.

Given the large amount of sensors utilized in the setups proposed in this thesis, together with the different conventions existing in the fields of CV and CG, a hierarchy of coordinate systems is used, where the pose of each sensor is always described with respect to a reference that occupies a higher position in the hierarchy, being D70K the top reference. In the following sections, a description is given on how measurements can be transformed between different origins of coordinates, as well as the local conventions employed for each kind of sensor.

3.1.1 Sensor Frame & Sensor Pose Description

For this work the sensor frame is defined as the reference of coordinates which origin is coincident with the virtual center of the sensor (virtual center of projections, in the case of central cameras), and its three axes are parallel to the axes of its parent coordinate system. In Figure 3.2 an example is depicted.

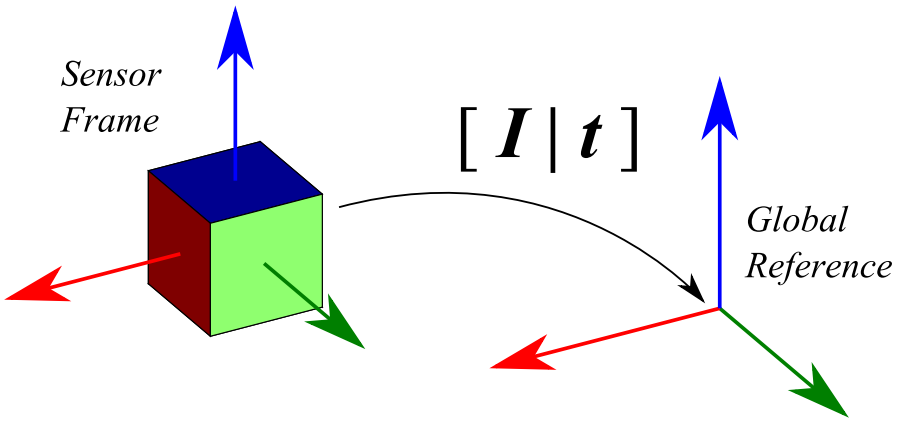


FIGURE 3.2: Definition of the Sensor Frame. The sensor frame reference coordinate system is defined as a pure translation from the global reference of coordinates to the sensor's virtual center (center of projections in the case of ideal central cameras). This definition is independent from the sensor orientation.

Considering the sensor frame reference, the pose of any sensor can be described with respect to its parent coordinate system in terms of six parameters. Of these six paremeters, three describe the position of the sensor, and three the orientation. In this thesis, the orientation of the sensor is described in terms of extrinsic rotations over the sensor frame axes. The magnitude of the rotations is represented by the Tait-Bryan angles (α, β, γ), with a body-fixed convention (rotations are applied about successively rotated axes) [Paul, 2008]. The result of the consecutive rotations is depicted in Figure 3.3.

As a convention for the rest of the thesis, the orientation of each sensor will be described assuming $x-$coordinates to be positive along the principal direction (optical axis in the case of cameras) and the $z-$coordinates positive with the sensor's up-vector. This convention will be used to describe sensor poses and a conversion to local sensor coordinates will be subsequently applied in each case.

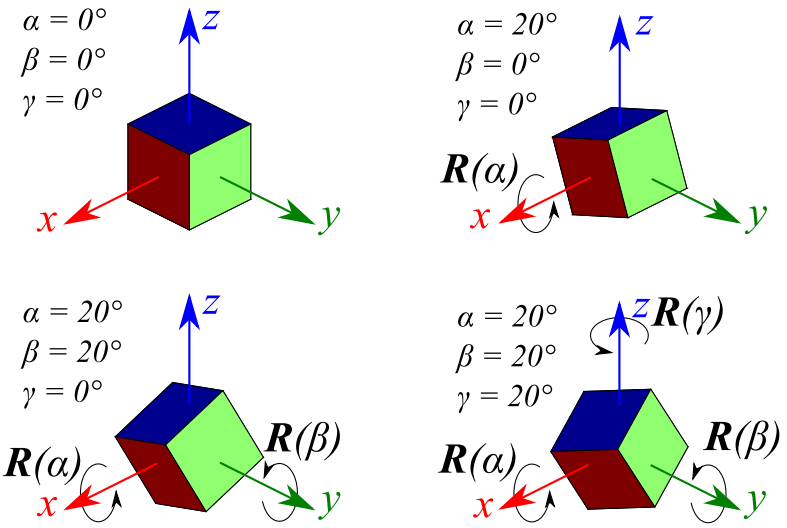


FIGURE 3.3: Sensor pose description. The orientation of a sensor is described with respect to the sensor frame coordinate system. Upper left: parallel orientation to parent reference. Upper right: pure roll rotation. Lower left: roll and pitch rotations. Lower right: all roll, pitch and yaw rotations.

Once the hierarchy of the coordinate systems is described, conversion between different references is possible so that metric measurements from a sensor A can be expressed with respect to a different sensor B, and vice versa. The transformations to be applied to the coordinates for this purpose are presented in the following section.

3.1.2 Rigid Body Transformations

In order to describe rigid transformations between different coordinate systems, homogeneous 4×4 matrices can be utilized. In particular, distinction between two different transformations has to be made: transformation from local sensor coordinates to global reference coordinates, and transformation from global reference coordinates to local sensor coordinates.

3.1.2.1 Local to Global Frame Transformation

A transformation matrix $\mathbf{M}_{G,L}$ can be defined that transforms the homogeneous representation of a point $\boldsymbol{P}$ in local sensor coordinates $\boldsymbol{P}_L$ to global reference coordinates $\boldsymbol{P}_G$. As support, the sensor frame (SF) is considered as an intermediate coordinate system, as in Eq. 3.1.

$$\boldsymbol{P}_{SF} = \begin{pmatrix} \mathbf{R}(\alpha, \beta, \gamma) & \mathbf{0} \\ \mathbf{0}^T & 1 \end{pmatrix} \boldsymbol{P}_L \tag{3.1}$$

The transformation from sensor frame to global coordinates (G), as defined in the previous section involves a pure translation $\mathbf{t}$, as in Eq. 3.2.

$$\boldsymbol{P}_G = \begin{pmatrix} \mathbf{I} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \boldsymbol{P}_{SF} \tag{3.2}$$

Expressions 3.1 and 3.2 can be combined to form Eq. 3.3, where $\mathbf{M}_{G,L}$ is the 4×4 homogeneous matrix describing the transformation between a local sensor coordinate system and its parent reference.

$$\begin{aligned} \boldsymbol{P}_G &= \begin{pmatrix} \mathbf{I} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \begin{pmatrix} \mathbf{R}(\alpha,\beta,\gamma) & \mathbf{0} \\ \mathbf{0}^T & 1 \end{pmatrix} \boldsymbol{P}_L \\ &= \begin{pmatrix} \mathbf{R}(\alpha,\beta,\gamma) & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \boldsymbol{P}_L \\ &= \mathbf{M}_{G,L} \boldsymbol{P}_L \end{aligned} \tag{3.3}$$

The rotation matrix $\mathbf{R}(\alpha,\beta,\gamma)$ is defined as in Eq. 3.4

$$\mathbf{R}(\alpha,\beta,\gamma) = \mathbf{R}_z(\gamma)\mathbf{R}_y(\beta)\mathbf{R}_x(\alpha) \tag{3.4}$$

where

$$\mathbf{R}_z(\gamma) = \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{R}_y(\beta) = \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix} \tag{3.5}$$

$$\mathbf{R}_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix}$$

3.1.2.2 Global to Local Frame Transformation

This transformation converts the homogeneous representation of a point $\boldsymbol{P}$ in global reference coordinates $\boldsymbol{P}_G$ to local sensor coordinates $\boldsymbol{P}_L$. It is represented by the matrix

$\mathbf{M}_{L,G}$, which can be described based on the previous local-to-global transformation, as in Eq. 3.6.

$$\begin{aligned}
\mathbf{M}_{L,G} = \mathbf{M}_{G,L}^{-1} &= \left[\begin{pmatrix} \mathbf{I} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \begin{pmatrix} \mathbf{R}(\alpha,\beta,\gamma) & \mathbf{0} \\ \mathbf{0}^T & 1 \end{pmatrix} \right]^{-1} \\
&= \begin{pmatrix} \mathbf{R}(\alpha,\beta,\gamma) & \mathbf{0} \\ \mathbf{0}^T & 1 \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{I} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix}^{-1} \\
&= \begin{pmatrix} \mathbf{R}(\alpha,\beta,\gamma)^T & \mathbf{0} \\ \mathbf{0}^T & 1 \end{pmatrix} \begin{pmatrix} \mathbf{I} & -\mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix}
\end{aligned} \tag{3.6}$$

These transformations are important both in CV and in CG, but especially so for applications combining both of them, as in Image Based Rendering (see Section 3.3.2). Furthermore, once the extrinsic relations between sensors are defined, there still exist different local axes conventions that are used for each different kind of sensor. The next section describes the axes conventions typically used in CV and in CG, that are also considered in this thesis.

3.1.3 Axes Convention in Computer Graphics and Computer Vision

In order to transform the representation of coordinates between different systems, the global transformations described in the previous sections are used. Nevertheless, both in Computer Vision and in Computer Graphics specific conventions exist for the local camera axes definition.

These coordinates are considered locally, for example, in order to describe camera projection models. In Figure 3.4 the conventions utilized in Computer Vision and OpenGL-based Computer Graphics are compared.

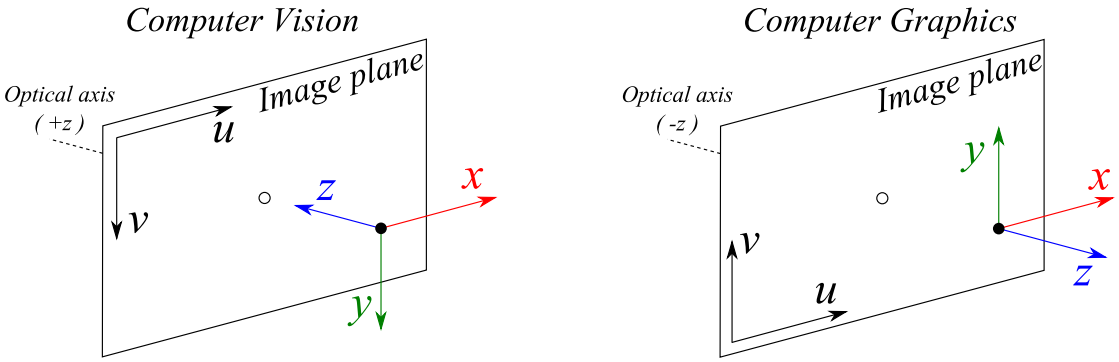


FIGURE 3.4: Axes Convention in Computer Vision and Computer Graphics (OpenGL convention). Left: CV - the optical axis is along the positive $z-$direction. Right: CG - the optical axis is along the negative $z-$direction.

In order to apply changes of axes convention between CV and the global reference used in this thesis (as described in Section 3.1.1), the 3×3 matrices 3.7 and 3.8 can be applied.

$$\mathbf{M}_{Global,CV} = \begin{pmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix} \tag{3.7}$$

$$\mathbf{M}_{CV,Global} = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{pmatrix} \tag{3.8}$$

In the same manner, change of axes convention in the context of CG can be carried out by applying Eqs. 3.9 and 3.10.

$$\mathbf{M}_{CG,Global} = \begin{pmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \tag{3.9}$$

$$\mathbf{M}_{Global,CG} = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \end{pmatrix} \tag{3.10}$$

3.1.4 Camera Registration

In order to estimate the pose of multiple cameras with respect to a common global reference, several approaches have been proposed in literature [Hartley and Zisserman, 2000],

[Esquivel et al., 2007], [Heng et al., 2013], [Pless, 2003]. Different solutions exist depending on the camera rig restrictions: there are offline and online solutions, as well as algorithms that require calibration targets or can work on natural scenes.

In the experiments described in this thesis, the extrinsic registration of the cameras was done offline by means of special calibration targets as well as additional cameras and bundle adjustment. The registration was assumed static for the duration of each experiment, ie. no online calibration algorithm was considered. For each different camera configuration that was proposed and evaluated, a new extrinsic registration session was conducted.

Open source libraries exist that allow for camera extrinsic registration. In particular, the author invites the reader to see the OpenCV [Bradski, 2000] and OpenGV [Kneip and Furgale, 2014] libraries.

3.2 Computer Vision

The field of Computer Vision comprises all the steps required in the process of gaining information about the world by means of camera images. Generally speaking, it can be understood as the process of taking images of the world as an input, and generating higher level understanding of the world as an output.

In this section, basic Image Processing and Computer Vision concepts are presented. In particular, a review of camera models is introduced together with basic image transformations and interpolation techniques. Finally, an introduction to multiple view geometry is given.

3.2.1 Camera Models

Image acquisition represents the first step within a complete CV system. In the process of acquiring an image, the 3D world is projected onto a 2D image plane, thus it is not possible to recover depth information for the imaged objects [Jähne, 1997]. Depending on the characteristics of the cameras and optics employed, different models have been proposed in literature that can explain how a 3D point is projected onto the image plane. These models normally account not only for ideal projections, but also for other distortions introduced by the camera lenses. This section introduces a review of both projection and distortion models for standard and omnidirectional cameras.

As for omnidirectional cameras, there exist multiple solutions that allow for large FoV to be imaged, some involving the use of exposed mirrors, eg. catadioptric cameras. Given their exposed nature, catadioptric cameras are not commonly used by the automotive industry. For this reason, fisheye lenses are the only omnidirectional cameras considered in this review. The reader is invited to find further literature on this topic in the works of [Geyer and Daniilidis, 2000] and [Barreto and Araujo, 2001]. A detailed survey on omnidirectional camera models can also be found in [Sturm et al., 2011].

3.2.1.1 Ideal Pinhole Model

The well-known pinhole projection model for standard cameras is presented in this section. The ideal pinhole model describes the camera aperture as a single point, through which all light rays pass. An ideal pinhole camera is illustrated in Figure 3.5.

FIGURE 3.5: Pinhole camera model. The ideal pin-hole camera model assumes all light rays passing through a single common point and projecting onto a focal plane.

The image plane is defined bering parallel to both camera axes at a focal distance f from the pin hole. In this model, a 3D point $\boldsymbol{P} = (X, Y, Z)^T$ is imaged in normalized coordinates $\boldsymbol{p} = ({}^X/_Z, {}^Y/_Z, 1)^T$. Therefore, the tangent of the angle defined by the view ray and the optical axis, together with the focal length f, characterize the distance between the principal and projection points on the image plane.

The relation between the angle θ described by the view ray and the optical axis, and the distance ρ to the principal point is given by Eq. 3.11.

$$\rho = f \tan \theta \tag{3.11}$$

This model can approximate the imaging characteristics of many standard cameras but does not account for distortions due to imperfections on the lenses.

3.2.1.2 Fish-Eye Models

Fish-eye lenses offer the possibilty to acquire images with an ultra-wide field of view, by means of a special mapping between viewing rays and pixel coordinates. This special mapping involves the bending of viewing rays and, for fields of view of 180° or above, large barrel distortions are unavoidable. In order to model the mapping between the 3D world and the 2D image plane for cameras with fisheye optics, a separation between projection and distortion models is commonly used [Geyer and Daniilidis, 2000], [Barreto and Araujo, 2001].

Ideal Fish-Eye Projection Models

In the following, some of the most commonly used projection models for fisheye optics are presented. They differ from each other in the projection function considered to map θ and ρ, which stand for the angular distance to the optical axis (in $[rad]$) and the distance over the image plane to the principal point (in $[pel]$), respectively. The coordinates (u', v') represent normalized image coordinates [Abraham and Förstner, 2005].

- *Equidistant model*

This model considers a projection function of the kind represented in Eq. 3.12.

$$\rho = c\theta \tag{3.12}$$

The projection of a world point $\boldsymbol{P} = (X, Y, Z)^T$ to normalized image coordinates is described by Eqs. 3.13, 3.14.

$$u' = \frac{X}{\sqrt{X^2 + Y^2}} \arctan \frac{\sqrt{X^2 + Y^2}}{Z} \tag{3.13}$$

$$v' = \frac{Y}{\sqrt{X^2 + Y^2}} \arctan \frac{\sqrt{X^2 + Y^2}}{Z} \tag{3.14}$$

The inverse projection is described by Eqs. 3.15, 3.16, 3.17.

$$X = \frac{u'}{\sqrt{u'^2 + v'^2}} \sin \sqrt{u'^2 + v'^2} \tag{3.15}$$

$$Y = \frac{v'}{\sqrt{u'^2 + v'^2}} \sin \sqrt{u'^2 + v'^2} \tag{3.16}$$

$$Z = \cos \sqrt{u'^2 + v'^2} \tag{3.17}$$

- *Stereographic model*

This model considers a projection function of the kind represented in Eq. 3.18.

$$\rho = c \tan \frac{\theta}{2} \tag{3.18}$$

The projection of a world point $\boldsymbol{P} = (X, Y, Z)^T$ to normalized image coordinates is described by Eqs. 3.19, 3.20.

$$u' = \frac{X}{\sqrt{X^2 + Y^2 + Z^2} + Z} \tag{3.19}$$

$$v' = \frac{Y}{\sqrt{X^2 + Y^2 + Z^2} + Z} \tag{3.20}$$

The inverse projection is described by Eqs. 3.21, 3.22, 3.23.

$$X = \frac{2u'}{1 + u'^2 + v'^2} \tag{3.21}$$

$$Y = \frac{2v'}{1 + u'^2 + v'^2} \tag{3.22}$$

$$Z = \frac{1 - (u'^2 + v'^2)}{1 + u'^2 + v'^2} \tag{3.23}$$

- *Orthogonal model*

This model considers a projection function of the kind represented in Eq. 3.24.

$$\rho = c \sin \theta \tag{3.24}$$

The projection of a world point $\boldsymbol{P} = (X, Y, Z)^T$ to normalized image coordinates is described by Eqs. 3.25, 3.26.

$$u' = \frac{X}{\sqrt{X^2 + Y^2 + Z^2}} \tag{3.25}$$

$$v' = \frac{Y}{\sqrt{X^2 + Y^2 + Z^2}} \tag{3.26}$$

The inverse projection is described by Eqs. 3.27, 3.28, 3.29.

$$X = u' \tag{3.27}$$

$$Y = v' \tag{3.28}$$

$$Z = \sqrt{1 - (u'^2 + v'^2)} \tag{3.29}$$

- *Equisolid model*

This model considers a projection function of the kind represented in Eq. 3.30.

$$\rho = c \sin \frac{\theta}{2} \tag{3.30}$$

The projection of a world point $\boldsymbol{P} = (X, Y, Z)^T$ to normalized image coordinates is described by Eqs. 3.31, 3.32.

$$u' = \frac{X}{\sqrt{2(X^2 + Y^2)}} \sqrt{1 - \frac{Z}{X^2 + Y^2 + Z^2}} \tag{3.31}$$

$$v' = \frac{Y}{\sqrt{2(X^2 + Y^2)}} \sqrt{1 - \frac{Z}{X^2 + Y^2 + Z^2}} \tag{3.32}$$

The inverse projection is described by Eqs. 3.33, 3.34, 3.35.

$$X = 2u' \sqrt{1 - (u'^2 + v'^2)} \tag{3.33}$$

$$Y = 2v' \sqrt{1 - (u'^2 + v'^2)} \tag{3.34}$$

$$Z = \sqrt{1 - (u'^2 + v'^2)} \tag{3.35}$$

Nonideal Fish-Eye Projection Models

The projection models above were aiming at describing the projection ideally, without considering deviations due, for example, to lens imperfections. These are normally accounted for in the distortion model, being radial and tangential the main distortion components. This is valid both for standard projective and fisheye cameras. In Figure 3.6 the effect of the main distortion components is depicted.

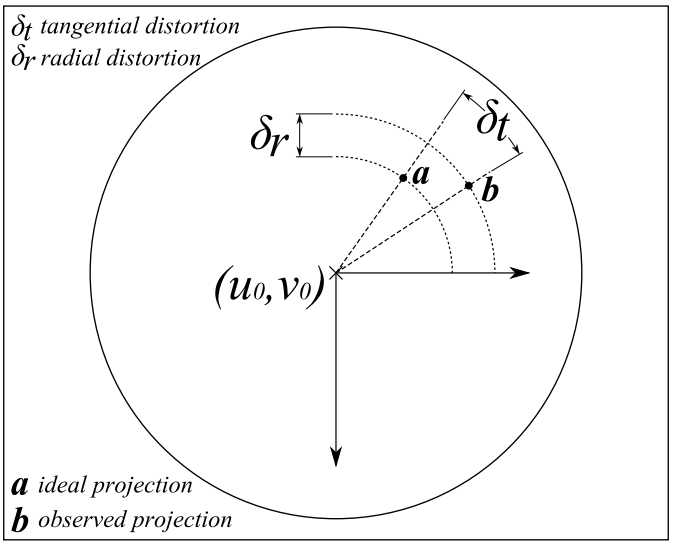


FIGURE 3.6: Main distortion components. The most commonly considered distortions are radial and tangential. As expressed by their names, radial distortions refer to the difference in radial distance to the center of projections between the ideal and observed projections. Tangential distortions refer to the angular deviation from the ideal location on the image plane.

In this thesis, a modified version of the general projection model originally proposed by [Geyer and Daniilidis, 2000] and [Barreto and Araujo, 2001] is considered to model the projection of a world point onto the image plane. This modification was initially proposed by [Mei and Rives, 2007] and it is described in the following, extracted from their original paper.

Let $\boldsymbol{P}$ be a point expressed in camera coordinates and $\boldsymbol{P}_S$ its projection on the unit sphere. The camera model proposed by [Mei and Rives, 2007] firstly introduces a change of reference frame corresponding to a translation of magnitude ξ over the optical axis, as in Eq. 3.36. This is illustrated in Figure 3.7.

$$\boldsymbol{P}_P = (P_{S,x}, P_{S,y}, P_{S,z} + \xi)^T \tag{3.36}$$

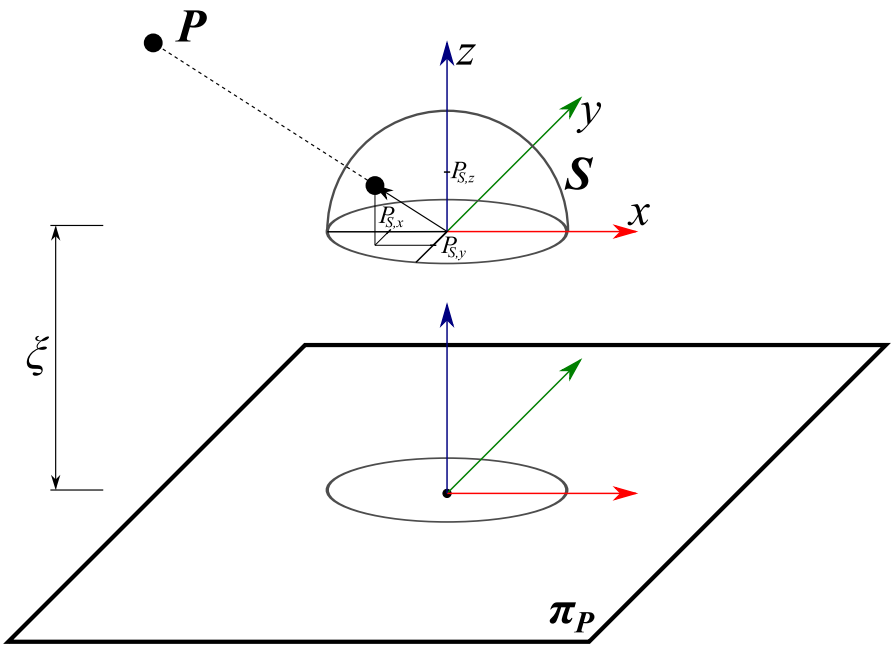


FIGURE 3.7: Change of reference frame as described in the model from [Mei and Rives, 2007], which is given by a translation of magnitude ξ over the optical axis.

Following the previous transformation of coordinates, the normalized undistorted ideal pinhole projection $(x'_u + y'_u)$ of point $\boldsymbol{P}$ onto the image plane can be described by means of Eqs. 3.37, 3.38.

$$x'_u = \frac{P_{S,x}}{P_{S,z} + \xi} \tag{3.37}$$

$$y'_u = \frac{P_{S,y}}{P_{S,z} + \xi} \tag{3.38}$$

Based on this ideal projection, a distortion model is considered with both radial and tangential components. In particular, the radial distortion $\mathcal{L}(\rho)$ is modeled by three parameters k_1, k_2, k_3 by means of Eq. 3.39, where $\rho = \sqrt{{x'_u}^2 + {y'_u}^2}$.

$$\mathcal{L}(\rho) = 1 + k_1\rho^2 + k_2\rho^4 + k_3\rho^6 \tag{3.39}$$

The tangential distortion dx, dy is modelled by two parameters t_1, t_2 as in Eq. 3.40.

$$\begin{bmatrix} dx \\ dy \end{bmatrix} = \begin{bmatrix} t_1(\rho^2 + 2{x'_u}^2) + 2t_2 x'_u y'_u \\ t_2(\rho^2 + 2{y'_u}^2) + 2t_1 x'_u y'_u \end{bmatrix} \tag{3.40}$$

Both distortions are applied by means of Eq. 3.41, obtaining the normalized distorted point $(x'_d + y'_d)$.

$$\begin{pmatrix} x'_d \\ y'_d \end{pmatrix} = \begin{pmatrix} x'_u \mathcal{L}(\rho) + dx \\ y'_u \mathcal{L}(\rho) + dy \end{pmatrix} \tag{3.41}$$

Finally, the unnormalized observed projection (x, y) of point $\boldsymbol{P}$ on the image plane can be obtained by considering focal lengths γ_x, γ_y and skew factor α, as in Eq. 3.42, where (x_0, y_0) represents the principal point or center of distortions.

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \gamma_x(x'_d + \alpha y'_d) + x_0 \\ \gamma_y y'_d + y_0 \end{pmatrix} \tag{3.42}$$

In Figure 3.8 the angular resolution in terms of angle per pixel along the lines passing through the center of distortions is shown for the cameras considered in this thesis. The cameras have been intrinsically calibrated based on the model proposed in [Mei and Rives, 2007].

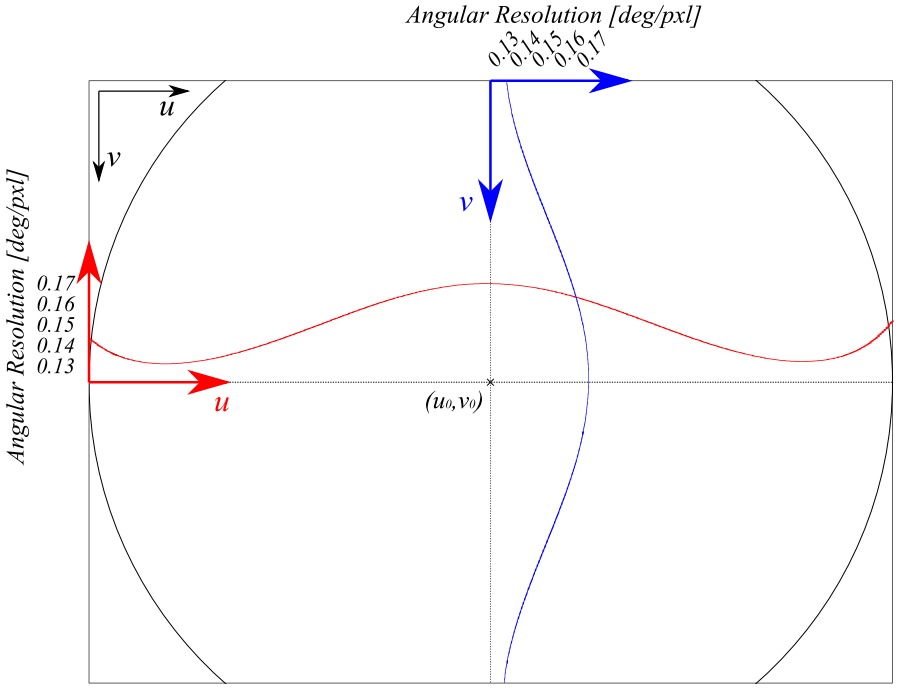


FIGURE 3.8: Angular resolution for the fisheye cameras used throughout the experiments of this thesis, which have a total resolution of 1280 × 960 pixels and horizontally cover 185° approximately. The resolution is represented in terms of angle per pixel, along the vertical and horizontal lines passing through the principal point. The graphs have been generated considering the camera calibration based on the model of [Mei and Rives, 2007].

The camera models that have been presented in this section are those relevant for the work carried out in this thesis. Further literature on the topic can be found in [Hartley and Zisserman, 2000] and [Ray, 2002].

3.2.2 Image Transformations

In this section a series of standard image processing transformations are presented. In particular, an introduction into the concept of image warping is given. Different interpolation techniques are also presented that offer different tradeoff levels between quality and complexity, which are typically chosen depending on application requirements.

3.2.2.1 Image Warping

Image warping is a process that implies image resampling at specific locations which are defined by the warping functions [Heckbert, 1989]. The intensities at the given locations are used to raster a new image that meets certain geometrical characteristics, eg. new size, distortion free, projective transformation, etc. These locations are not necessarily integral, thus different interpolation techniques are often used. Warping functions can be represented by planar motions [Szeliski, 2006], but also by nonlinear functions. A typical example of image warping is the use of cylindrical or spherical projections for panning cameras. Other relevant applications of image warping include:

- Resizing: Either to enlarge or reduce the size of an image, intensity values not corresponding to a single pixel are usually required. The grid of sample locations for the resize images can be considered a warping function.
- Distortion correction: The warping function is defined by the correspondences between ideal projection and real observed image coordinates. The distortion-corrected image is obtained by sampling the image at the coordinates that correspond to the ideal raster positions [De Villiers et al., 2008].
- Transformation of coordinates: The mapping of images between coordinates systems is a common problem that usually implies nonlinear warping functions. One of the most typical examples is the 2D projection of the earth's surface used in cartography [Tobler, 1973], [Snyder, 1997].

An example of image warping is depicted in Figure 3.9.

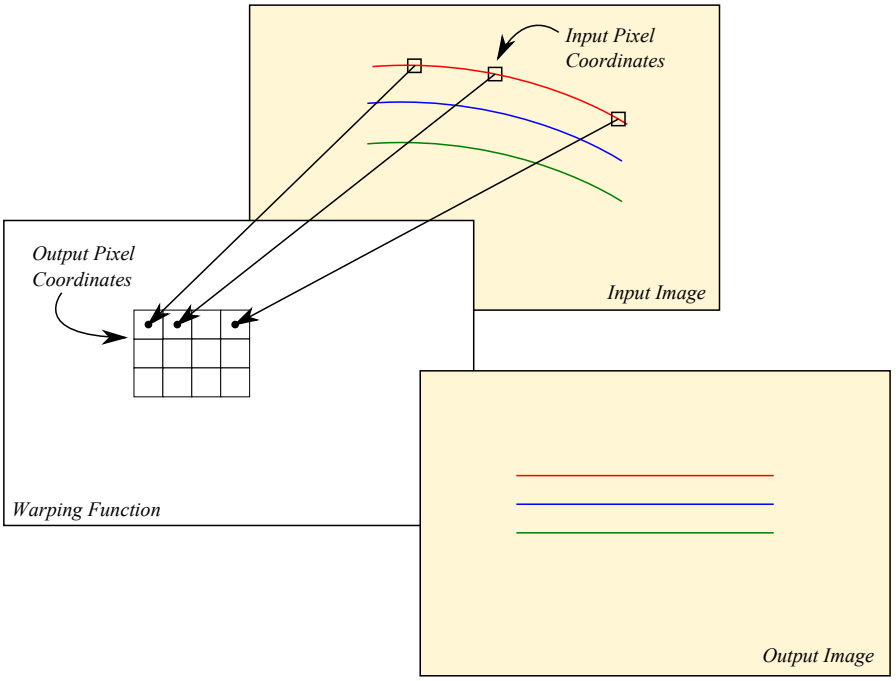


FIGURE 3.9: Warping function. For each output pixel coordinate, the warp function describes the corresponding coordinates on the original image. The warping function can be defined in order to perform distortion correction or any transformation of image coordinates.

Depending on how severe the warping is, large changes in pixel sizes may occur between the original and warped images. This can cause *jaggies* or other image artefacts [Blinn, 1989] due to nonadequate interpolation techniques. The works of [Turkowski, 1990] and [Van Ouwerkerk, 2006] evaluate these effects in detail.

3.2.2.2 Interpolation Techniques

Image interpolation works in two dimensions and aims to achieve a best approximation of a pixel intensity value, based on the values of the surrounding pixels. In the following, some of the most commonly used interpolation techniques are described.

Bilinear Interpolation

Bilinear interpolation is an interpolation technique which takes into consideration the intensity values of the four neighbouring pixels to the desired location (x, y). The estimated intensity value $\widetilde{\mathbf{I}}(x, y)$ is proportional to the proximity to each of the

neighbours and is given by Eq. 3.43, where x_1, x_2, y_1, y_2 describe the coordinates of the four neighbouring pixels.

$$\begin{aligned}\widetilde{\mathbf{I}}(x,y) &= \tfrac{1}{(x_2-x_1)(y_2-y_1)} \\ &\quad (\ \mathbf{I}(x_1,y_1)(x_2-x)(y_2-y) \\ &\quad + \ \mathbf{I}(x_2,y_1)(x-x_1)(y_2-y) \\ &\quad + \ \mathbf{I}(x_1,y_2)(x_2-x)(y-y_1) \\ &\quad + \ \mathbf{I}(x_2,y_2)(x-x_1)(y-y_1) \ \ )\end{aligned} \tag{3.43}$$

Despite allowing to estimate intensities at nonintegral pixel coordinates, bilinear interpolation is known to produce a number of image artefacts (eg. aliasing). More computationally demanding techniques are usually utilized in order to reduce these effects. Figure 3.10 illustrates the operations graphically.

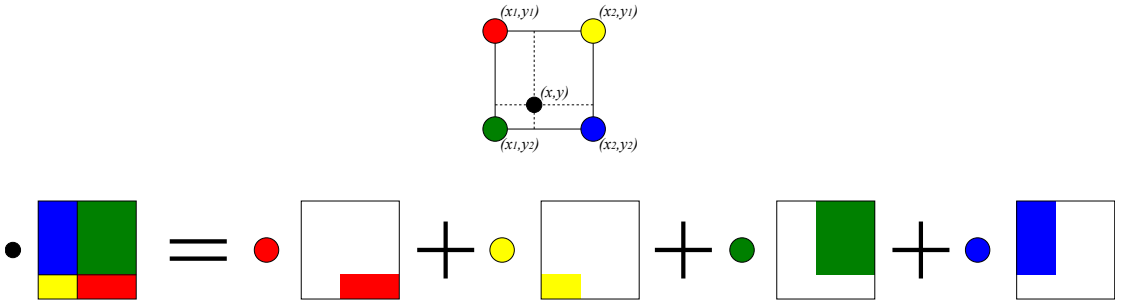


FIGURE 3.10: Bilinear interpolation. The intensity value at a given location is determined by the four nearest neighbours and it is proportional to the proximity of these.

Bicubic Interpolation

Bicubic interpolation is commonly used for image resampling since it outperforms bilinear interpolation with regard to fine detail. This interpolation technique estimates the intensity value at a given location by considering sixteen neighbouring pixels. It is based on the idea of estimating the parameters of a 3-degree polynom which passes through neighbouring pixels and has the same first derivative. Bicubic interpolation can be broken into two one-dimensional cubic interpolations along each direction. One-dimensional cubic interpolation can be computed as in Eq. 3.44.

$$\begin{aligned}\widetilde{\mathbf{I}}_{cubic}(\mathbf{I}(x_{-1}),\mathbf{I}(x_0),\mathbf{I}(x_1),\mathbf{I}(x_2),x) &= \left(-\tfrac{1}{2}\mathbf{I}(x_{-1})+\tfrac{3}{2}\mathbf{I}(x_0)-\tfrac{3}{2}\mathbf{I}(x_1)+\tfrac{1}{2}\mathbf{I}(x_2)\right)x^3 \\ &+ \left(\mathbf{I}(x_{-1})-\tfrac{5}{2}\mathbf{I}(x_0)+2\mathbf{I}(x_1)-\tfrac{1}{2}\mathbf{I}(x_2)\right)x^2 \\ &+ \left(-\tfrac{1}{2}\mathbf{I}(x_{-1})+\tfrac{1}{2}\mathbf{I}(x_1)\right)x \\ &+ \mathbf{I}(x_0)\end{aligned} \tag{3.44}$$

Based on Eq. 3.44, two-dimensional bicubic interpolation in pixel coordinates (x, y) can be conducted as by means of Eq. 3.45

$$\widetilde{\mathbf{I}}_{bicubic}(x,y) = \widetilde{\mathbf{I}}_{cubic}(\mathbf{I}_X(x,y_{-1}),\mathbf{I}_X(x,y_0),\mathbf{I}_X(x,y_1),\mathbf{I}_X(x,y_2),y) \tag{3.45}$$

where

$$\mathbf{I}_X(x,y) = \widetilde{\mathbf{I}}_{cubic}(\mathbf{I}(x_{-1},y),\mathbf{I}(x_0,y),\mathbf{I}(x_1,y),\mathbf{I}(x_2,y),x) \tag{3.46}$$

In Figure 3.11, the complete process is shown graphically.

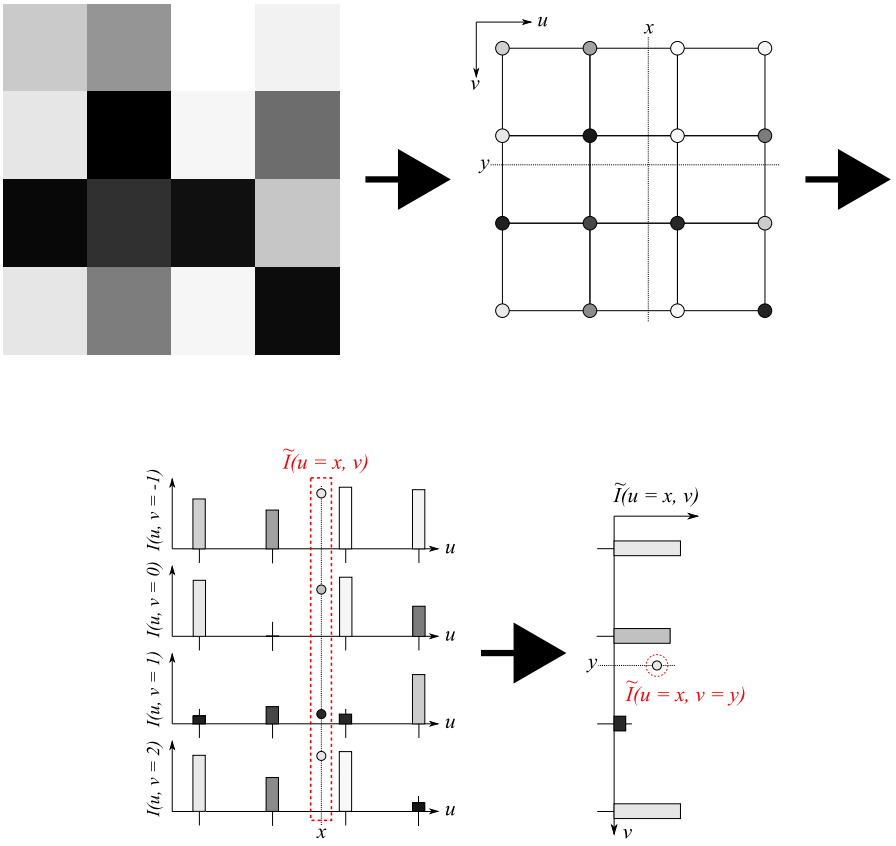


FIGURE 3.11: Bicubic interpolation. Bicubic interpolation can be split into two one-dimensional cubic interpolations anlong both x- and y-directions. Cubic interpolation involves the fitting of a degree-three polynom that meets the intensity values on neighbouring pixels and their first order derivative.

Despite producing better level of detail than bilinear interpolation, certain image artefacts may be still introduced due to negative lobes on the interpolation function [Blinn, 1989].

Lanczos Interpolation

The Lanczos interpolation has been reviewed in literature [Turkowski, 1990], [Blinn, 1989] as an optimal alternative for image interpolation preserving detail and minimizing aliasing artefacts.

In particular, the filtering is applied by means of Eq. 3.47

$$S(x,y) = \sum_{i=\lfloor x \rfloor - a + 1}^{\lfloor x \rfloor + a} \sum_{j=\lfloor y \rfloor - a + 1}^{\lfloor y \rfloor + a} s_{ij} L(x-i) L(y-j) \tag{3.47}$$

where s_{ij} represents the intensity of the original image at position (i, j) and $L(x)$ is Lanczos' kernel with size parameter a, as defined in Eq. 3.48. The term $\lfloor x \rfloor$ is used to represent the floor function.

$$L(x) = \begin{cases} \operatorname{sinc}(x) \operatorname{sinc}(x/a) & \text{if } x < a \\ 0 & \text{if } x \geq a \end{cases} \tag{3.48}$$

After resampling, rastering of the virtual images is possible as in Figure 3.9. The effect of considering different values for the parameter a is shown in Figure 3.12 for a one-dimensional function.

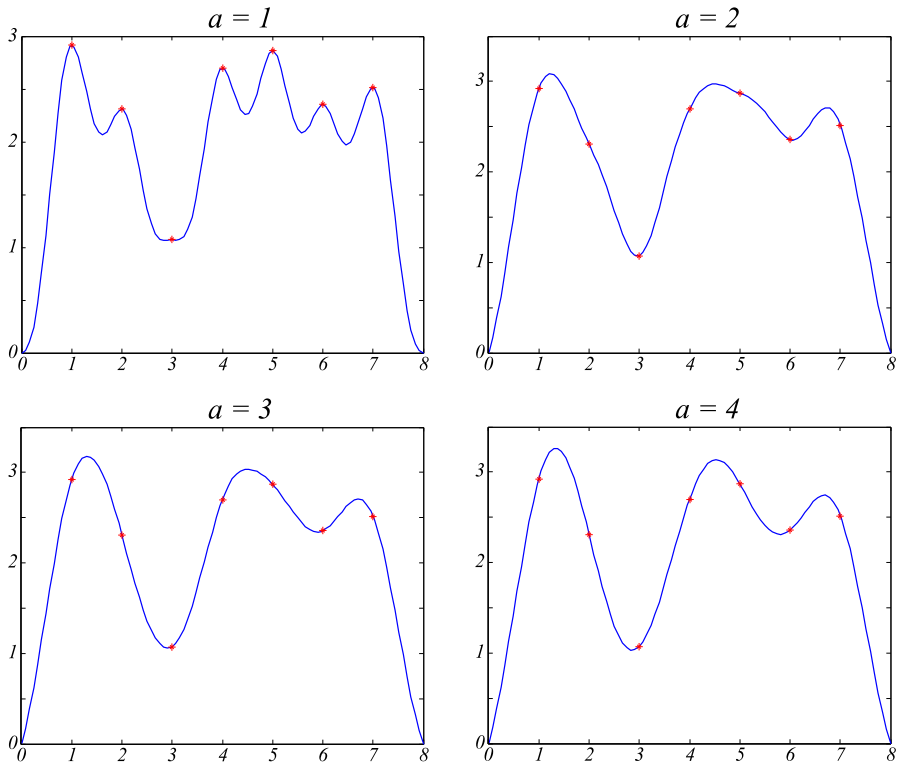


FIGURE 3.12: Lanczos interpolation for different values of the parameter a.

In the experiments presented in this thesis, a Lanczos filter with a size parameter $a = 4$ has been considered to deal with large changes in pixel size. No significant change has been observed by applying different size parameters.

3.2.3 Multiple View Geometry

This section reviews basic concepts in the field of multiple view geometry relevant for the work in this thesis. In particular, stereo vision, epipolar geometry and keypoint-based computer vision are the aspects on which this section focuses. The contents presented in this section can be found in more detail in [Hartley and Zisserman, 2000].

3.2.3.1 Stereo Vision

Stereo Vision is the process of extracting 3D information from several view images of a single scene [Hartley and Zisserman, 2000]. Assuming known camera poses and calibrated cameras, from two corresponding image points $\boldsymbol{x}$ and $\boldsymbol{x}$', the 3D coordinates of the world point $\boldsymbol{X}$ can be estimated by means of simple triangulation, as shown in Figure 3.13.

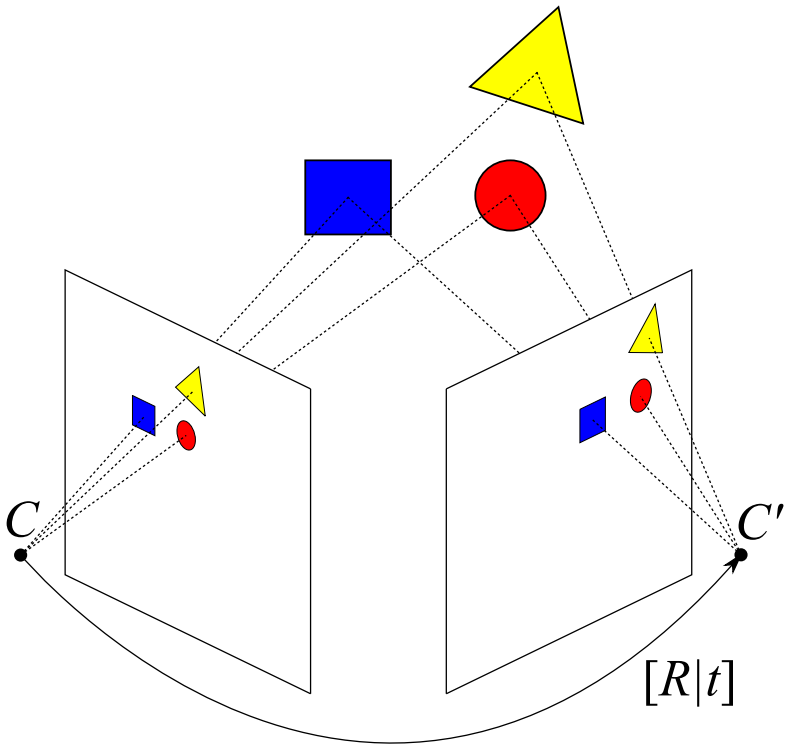


FIGURE 3.13: Stereo Vision. Based on multiple views of a single scene, the 3D position of an object can be determined by triangulation, based on the projection of the object onto both image planes.

Prior to the 3D reconstruction, stereo vision requires a series of preprocessing steps that reduce the complexity of the feature matching or disparity estimation. Among others, these include:

- Distortion correction. In this step distortions due to, for example, lens imperfections are removed. A typical example of this kind of distortions is the barrel distortion, which normally occurs for very wide angle lenses. To achieve this, the image intensities are mappend to the image coordinates that the ideal projection model predicts.
- Epipolar rectification. This step is aimed at aligning epipolar lines with raster lines. If both images from a camera pair are epipolar-rectified so that each corresponding epipolar line is projected onto the same raster line, the search of correspondences becomes a one-dimensional search, thus reducing computational complexity.

Combining restrictions based on the epipolar geometry with the projection and distortion camera models presented, both dense and feature-based stereo schemes can be considered. In a dense stereo setup, a disparity is estimated for every pixel

on an image, while feature-based stereo is based on the prior search of distinctive image point, lines or structures. In the case of point features, these are usually referred to as keypoints or corners [Szeliski, 2006].

In the following sections, epipolar rectification and keypoint-based stereo vision are described in further detail.

3.2.3.2 Epipolar Geometry & Epipolar Rectification

Epipolar geometry refers to the projective properties that arise from the examination of more than one view of a single scene, observed from different perspectives [Hartley and Zisserman, 2000]. Before going into further details, it is worthwhile to define some basic concepts:

- **Baseline**: is the line joining the camera centers of two views of a common scene.
- **Epipole**: is the point of intersection of the line joining the camera centers with the image planes. Equivalently, it is the projection in one view of the camera center of the other view.
- **Epipolar plane**: is every plane that contains the baseline. All epipolar planes form a pencil.
- **Epipolar line**: is the intersection of an epipolar plane with the image plane. All epipolar lines intersect at the epipole. The intersection of a single epipolar plane with both image planes defines two corresponding epipolar lines across both views. In perspective cameras, epipolar lines are straight. This is not true for nonperspective cameras, due to the nonlinearity of their projection fuction.

These elements are described in Figure 3.14.

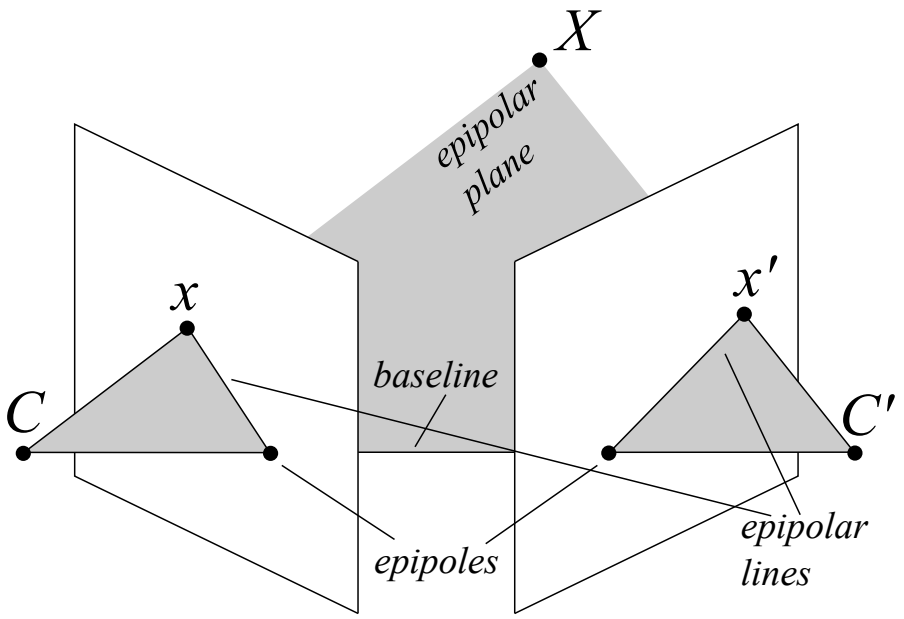


FIGURE 3.14: Epipolar geometry. Any 3D point not contained in the baseline defines a triangle together with both camera centers. The plane to which this triangle belongs is an epipolar plane. All epipolar planes define a pencil which pases through the baseline. The intersection of any epipolar plane with the image planes defines the epipolar lines. In perspective cameras, epipolar lines are imaged as straight lines.

The Fundamental Matrix

The fundamental matrix is a 3×3 matrix of rank two that relates corresponding points in two different views. Considering corresponding homogeneous image coordinates $\boldsymbol{x}$ and $\boldsymbol{x}'$, the fundamental matrix $\mathbf{F}$ satisfies Eq. 3.49.

$$\boldsymbol{x}'^T \mathbf{F} \boldsymbol{x} = 0 \tag{3.49}$$

For correspondence search, the fundamental matrix can be used to obtain the epipolar line l' where the projection $\boldsymbol{x}'$ of $\boldsymbol{x}$ lies, as depicted in Figure 3.15.

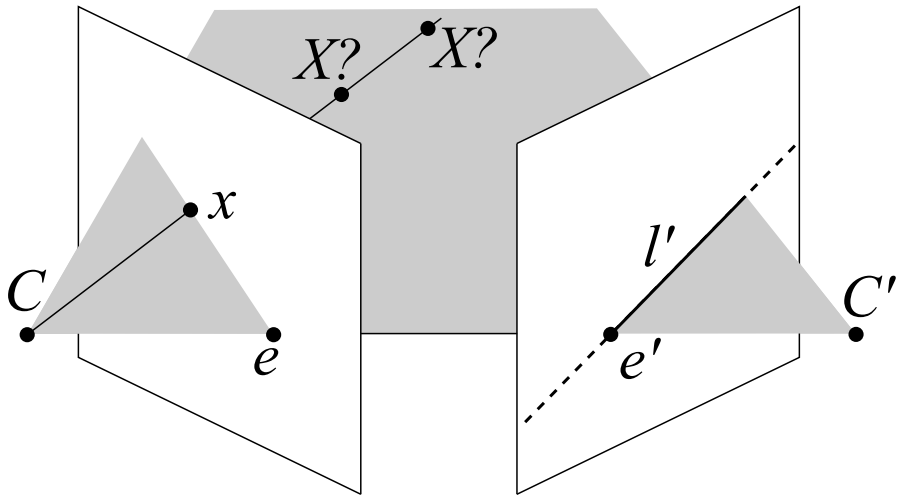


FIGURE 3.15: Fundamental matrix. Given a projection $\boldsymbol{x}$ of $\boldsymbol{X}$, the fundamental matrix defines the epipolar line l' on the other view where the projection $\boldsymbol{x}$' lies.

Epipolar Rectification

Epipolar rectification is a standard step in stereo vision processing, in order to reduce the correspondence search space, thus saving in computation efforts [Hartley and Zisserman, 2000]. It can be regarded as a projection of the world into a virtual camera pair, which fulfills the resctriction that each common point observed by both cameras is imaged on the same image row. A common model used for this purpose is the ideal pinhole model. This model projects the world into a plane which is a fixed distance f away from the projection center.

In conventional perspective cameras, a linear rectifying transformation H exists, as described in Eq. 3.50, where $\mathbf{K}_C, \mathbf{K}_V$ are the projection matrices of the original and virtual cameras respectively, and $\mathbf{R}$ is the rotation that is applied to the original camera to rectify it.

$$\mathbf{H} = \mathbf{K}_V \mathbf{R} \mathbf{K}_C^{-1} \tag{3.50}$$

For nonperspective cameras, a linear epipolar rectification transformation does not exist, thus the previous expression is not valid. In Chapter 5, an extended discussion is presented on existing possibilities for epipolar rectification of omnidirectional images.

Once the epipolar rectification has been carried out, the search of correspondences across the rectified images can be performed. In the following section, an introduction is given into keypoint-based computer vision techniques.

3.2.3.3 Keypoint-based CV

In computer vision, the concept of feature refers to a certain amount of information which can be used to carry out a certain kind of processing [Lowe, 2004], [Oja et al., 1999]. The process of extracting distinctive features is usually split into two more specific steps, namely feature detection and feature description. The nature of the features considered depends largely on the application that is to be conducted. In the literature there exist many examples of possible features: line segments, corner-like keypoints, or even tracks of a primitive feature can be considered as a higher level feature themselves. For the work carried out throughout this thesis, focus has been put into 2D keypoints which fulfill certain local constraints as features. In the following, a review of the keypoint detection and description techniques utilized is presented.

Keypoint Detection

Several possibilities exist for keypoint detection, as in [Harris and Stephens, 1988] and [Shi and Tomasi, 1994]. Depending on the task at hand, different characteristics may be desirable from a feature, thus the detection process shall be based on different tests. Keypoint detectors invariant to scale [Lowe, 2004], [Mikolajczyk and Schmid, 2004], as well as to affine transformations [Morel and Yu, 2009] have been proposed. The detector originally proposed by [Rosten and Drummond, 2005] has been largely used throughout this thesis and is therefore described in the following.

The FAST (Features from Accelerated Segment Test) keypoint detector was originally proposed in [Rosten and Drummond, 2005] with the aim to reach real-time performances on solving the problem of 3D model-based tracking. In order to detect a FAST keypoint at any pixel location $\boldsymbol{p}$, a Bresenham circle of radius 3 surrounding $\boldsymbol{p}$ is considered, which is defined by 16 pixels [Bresenham, 1965]. Pixel $\boldsymbol{p}$ is a corner if there exists a set of n contiguous pixels in the circle which are all brighter than $I + t$, or all darker than I - t, where I is the intensity of the image at $\boldsymbol{p}$ and t is a predefined threshold. An example image from the original paper of [Rosten and Drummond, 2005] is shown on Fig. 3.16.

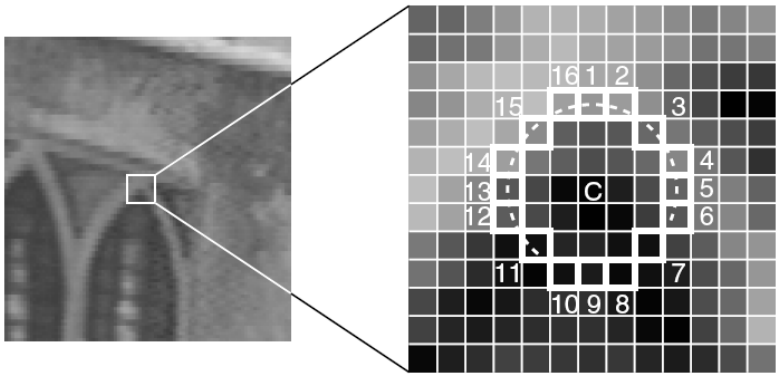


FIGURE 3.16: Description of the FAST (Features from Accelerated Segment Test) keypoint detector. Taken from [Rosten and Drummond, 2005].

Open source implementations exist of the FAST keypoint detector [Bradski, 2000].

Keypoint Description

Up to this point discussion was focused on the detection of image keypoints, which provides no more than a list of 2D coordinates of the representative keypoints. The next step is therefore to provide a description for each of the detected keypoints. A review on local descriptors can be found in [Mikolajczyk and Schmid, 2005]. In the following, the BRIEF (Binary Robust Independent Elementary Features) descriptor is presented, since it is largely used in this thesis.

The BRIEF descriptor was originally proposed in [Calonder et al., 2010] as an efficient keypoint descriptor based on binary strings. It can be computed using simple intensity difference tests as described in the following.

On a first step, a patch $\mathbf{p}$ of size $S \times S$ is defined, centered on the keypoint $\boldsymbol{p}$. A test $\tau(\mathbf{p}; \boldsymbol{x}_1, \boldsymbol{x}_2)$ is defined as in Eq. 3.51 where $I(\boldsymbol{x})$ is the pixel intensity in a smoothed version of the patch $\mathbf{p}$, at pixel location $\boldsymbol{x}$.

$$\tau(\mathbf{p}; \boldsymbol{x}_1, \boldsymbol{x}_2) = \begin{cases} 1 & \text{if } I(\boldsymbol{x}_1) < I(\boldsymbol{x}_2) \\ 0 & \text{otherwise} \end{cases} \tag{3.51}$$

A set of binary tests is then defined by means of unique $(\boldsymbol{x}_1, \boldsymbol{x}_2)$-location pairs that leads to the complete binary description. In the original paper of [Calonder et al., 2010] different spatial arrangements of the binary tests are evaluated. These are shown in Figure 3.17.

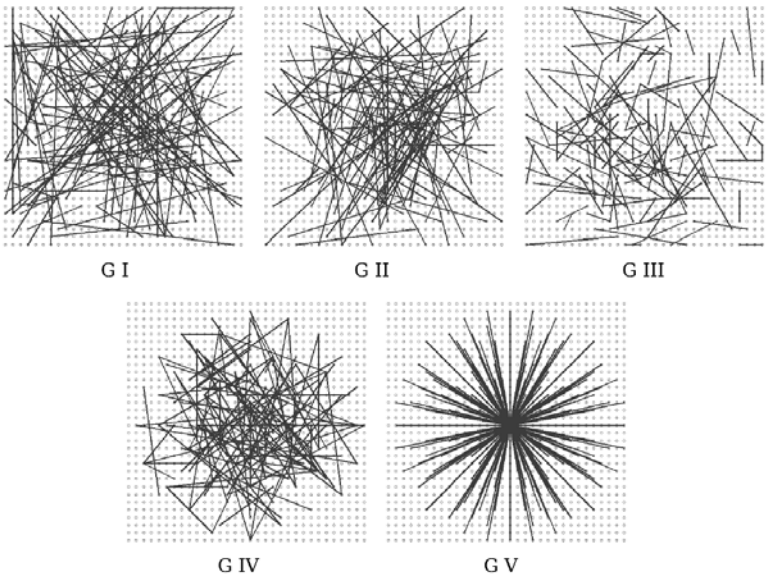


FIGURE 3.17: Spatial arrangements for the binary tests in the BRIEF keypoint descriptor. Taken from [Calonder et al., 2010].
I) (X,Y) $\sim$ i.i.d. U(-$\frac{S}{2}$,$\frac{S}{2}$)
II) (X,Y) $\sim$ i.i.d. G(0,$\frac{1}{25}S^2$)
III) X $\sim$ i.i.d. G(0,$\frac{1}{25}S^2$), Y $\sim$ i.i.d. G(x_i,$\frac{1}{100}S^2$)
IV) (x_i, y_i) are randomly sampled from discrete locations.
V) $\forall i : x_i = (0,0)^T$ and y_i takes all values on a coarse polar grid.

Open source implementations exist of the BRIEF keypoint descriptor [Bradski, 2000].

Descriptor Matching

Once the descriptors are computed for all detected keypoints, these can be compared with a list of descriptors corresponding to keypoints from another image. Usually, a match is given if a minimum similarity score is reached. Depending on the kind of descriptor considered, different similarity measurements can be defined. In the case of binary descriptors like BRIEF, it is common practice to compute the Hamming distance between the binary words, which can be conducted by simply counting how many of the tests described in Eq. 3.51 produce a different output.

Since brute-force matching of keypoint descriptors implies huge computation efforts for large amounts of these (cuadratic with respect to the number of detected keypoints), several approaches have been proposed in literature to perform minimum distance searches on high-dimensional spaces [Nene and Nayar, 1997], [Samet, 1990], [Beis and Lowe, 1997]. In setups where information about the relative camera pose is available and epipolar rectification is conducted (as usually

in stereo vision), the matching can be restricted to the keypoints on a common raster line.

Once the matches are available, 3D reconstruction is possible by means of triangulation. This process is described in the following section.

Triangulation

Assuming known camera poses and calibrated cameras, from two corresponding image points $\boldsymbol{x}$ and $\boldsymbol{x}$', the 3D coordinates of the world point $\boldsymbol{X}$ can be estimated. Figure 3.18 depicts the triangulation scheme for the ideal case where the bearing rays perfectly intersect.

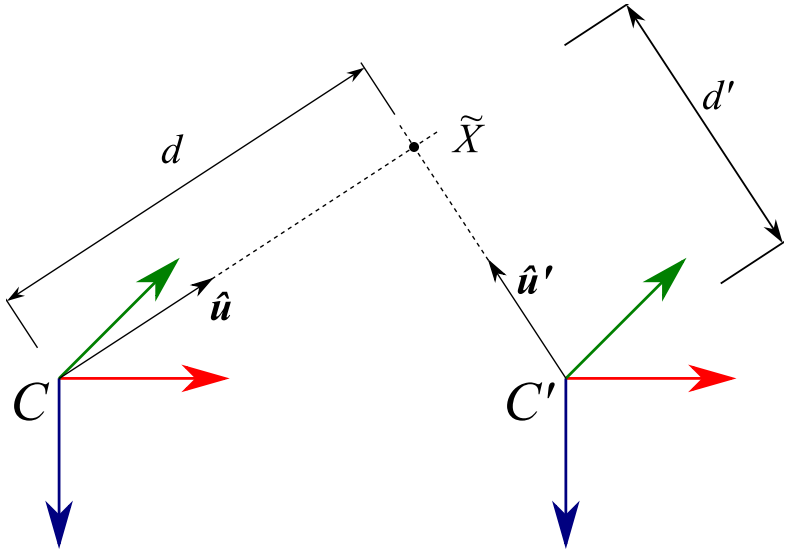


FIGURE 3.18: Ideal triangulation. The 3D rays $\hat{\mathbf{u}}$, $\hat{\mathbf{u}}$', defined by the projections $\boldsymbol{x}$ and $\boldsymbol{x}$', intersect perfectly. The intersection $\widetilde{\boldsymbol{X}}$ is the estimated position of the real 3D point $\boldsymbol{X}$.

In the ideal case, the estimated position of the point $\widetilde{\boldsymbol{X}}$ can be obtained by solving Eq. 3.52, where $\boldsymbol{C}$ and $\boldsymbol{C}'$ represent the position of both camera centers expressed in terms of a common reference frame and $\hat{\mathbf{u}}$, $\hat{\mathbf{u}}'$ are expressed in camera frame, i.e. they already account for the orientation of the cameras with respect to the reference coordinate system.

$$\boldsymbol{C} + d \cdot \hat{\mathbf{u}} = \widetilde{\boldsymbol{X}} = \boldsymbol{C}' + d' \cdot \hat{\mathbf{u}}' \tag{3.52}$$

In most cases, however, the view rays will not perfectly intersect in 3D space, thus this solution for the triangulation problem is not applicable. Different approaches

can be adopted to overcome this. In the following, a simple mid-point intersection approach is described.

Mid-point Algorithm

The mid-point algorithm is a basic workaround technique to allow triangulation of 3D rays that do not perfectly intersect. Its principle is very simple, and it consists on finding the two points along the corresponding view rays with minimum distance between them. The triangulated point $\widetilde{\boldsymbol{X}}$ corresponds to the mid point between these. Figure 3.19 depicts the mid-point algorithm for nonintersecting rays.

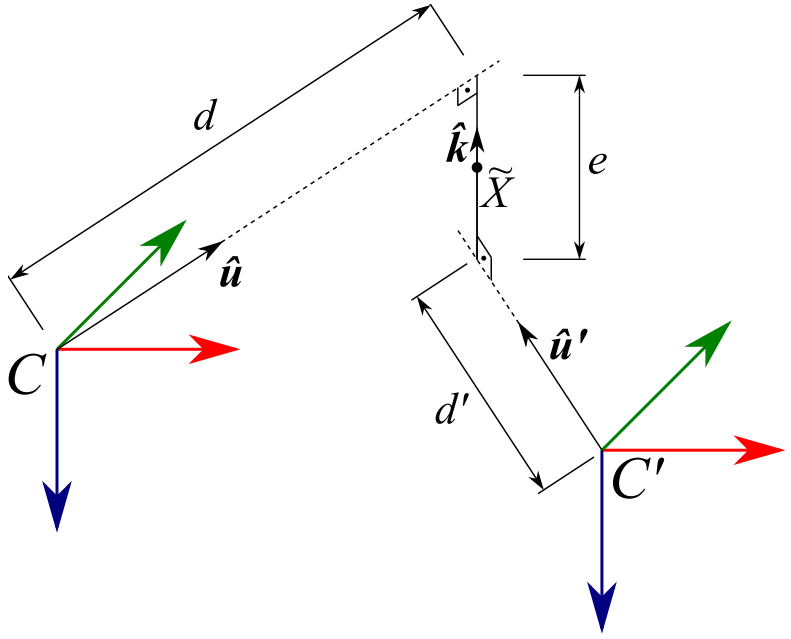


FIGURE 3.19: Mid Point Algorithm. The 3D rays $\hat{\mathbf{u}}$ and $\hat{\mathbf{u}}$' do not intersect. The intersection is approximated by considering the point that has minimum distance to both 3D rays.

The mid-point solution can be obtained by solving Eq. 3.53 for d, d' and e.

$$\widetilde{\boldsymbol{X}} = \boldsymbol{C} + d \cdot \hat{\mathbf{u}} + \frac{1}{2} e \hat{\mathbf{k}} = \boldsymbol{C}' + d' \cdot \hat{\mathbf{u}}' - \frac{1}{2} e \hat{\mathbf{k}} \tag{3.53}$$

The unit vector $\hat{\mathbf{k}}$ can be obtained as the cross product of $\hat{\mathbf{u}}$ and $\hat{\mathbf{u}}'$ as in Eq. 3.54 and e stands for the minimum distance between both 3D rays.

$$\hat{\mathbf{k}} = \hat{\mathbf{u}} \times \hat{\mathbf{u}}' \tag{3.54}$$

As described in Section 3.1, once the 3D measurements are conducted they can be transformed to any other reference coordinate system. In systems with multiple stereo camera pairs, it seems reasonable to define a global reference to which all measurements refer. In the case of study in this thesis, this will be vehicle's origin of coordinates, as defined in Section 3.1.

In Chapter 5 an extended discussion is presented about the accuracy of the 3D measurements depending on the precision of the keypoint detector and descriptors considered for the use case of automotive surround view systems.

3.3 Computer Graphics

The term Computer Graphics refers to the generation of images based on objects models, by means of graphic software and hardware. In a way, it can be seen as the opposite to Computer Vision, where images of the world are taken as an input, and higher level models are generated.

The main topics covered in this section include 3D Graphics and Image Based Rendering. An open source rendering toolkit is also presented.

3.3.1 3D Rendering

The term 3D Rendering refers to the process by means of which a virtually generated 3D scene is "drawn" as a 2D image or frame. There exist different rendering techniques, depending on the type of application requirements. There are, for example, applications that require photo-realistic results (eg. movie making) or real-time rendering (eg. video games). A trade-off between quality and rendering time has to be found. Since driving assistance is a field where large latencies are not permitted, the focus of this thesis will be put into real-time rendering techniques.

3.3.1.1 Real-Time Rendering

A huge amount of dedicated graphic hardware exists that allows for accelerated rendering [Fernando et al., 2004]. Different APIs have been stablished in the last decades to interact with GPUs, being OpenGL [Woo et al., 1999] and DirectX [Bargen and Donnelly, 1998] the dominant ones. For this thesis, only OpenGL is considered.

The APIs make use of geometrical descriptions of objects, together with color and texture properties to create a virtual description of a scene. Once the scene is built, a virtual camera is positioned at a desired location within the scene and the rendering is carried out, ie. a 2D "picture" of the scene is taken.

The geometrical description of objects is based on the so-called wireframe model. The wireframe model allows for description of a solid physical object in terms of lines and points that represents the object's significant edges. The resulting description is a polygonal mesh, of which most basic elements are points, lines and

triangles, as illustrated in Figure 3.20. Despite the term *Point* being very general and *Vertex* rather specific of Computer Graphics, in this section both terms will be used as equivalent.

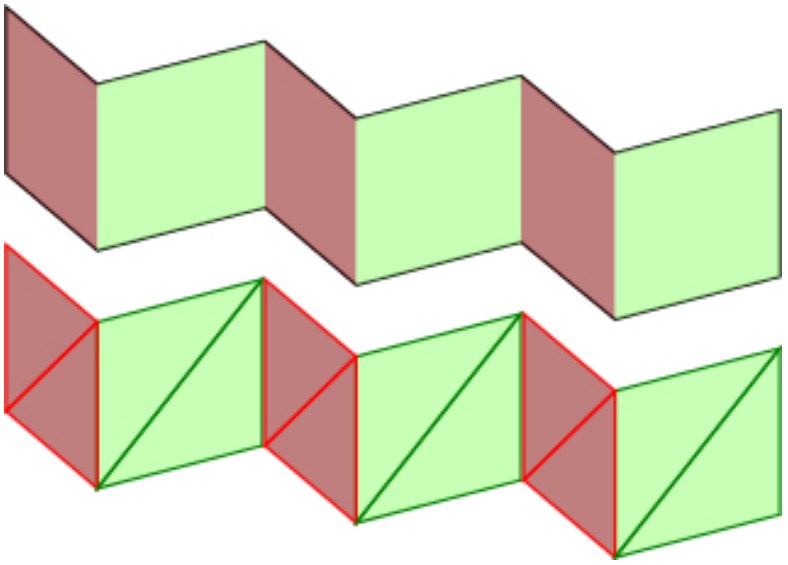


FIGURE 3.20: Wireframe model. The most basic entities in this model are points, lines and triangles. A mesh of triangles is usually defined that approximates a given surface. The size and amount of triangles relates to the wellness of the surface smoothness approximation.

Complementing the geometrical properties of objects, texture and lighting can be used. For this purpose, every vertex can normally be assigned a texture coordinate and a normal. The API takes these attributes from the application and the consecutive vertex operations are usually pipelined. In the following, the OpenGL fixed-pipeline vertex operations are introduced.

3.3.1.2 Vertex Operations on a Fixed Pipeline

In order to define complex scenes with multiple objects, the rendering APIs offer different functionalities to deal with multiple coordinate systems. In particular, transformation matrices can be applied per vertex that allow for independence between scene objects and rendering cameras. In this section, a description of the vertex operation process in OpenGL is described.

In computer graphics, as in computer vision, there is no hard requisite for a global reference coordinate system, since the absolute positions of cameras and objects are not relevant; the relative positions are. In other words, if the situations are considered where a) an object is brought away from the camera and b) the camera

is brought away from the object in the opposite direction, the output generated in the rendering process is exactly the same. This effect is illustrated in Figure 3.21.

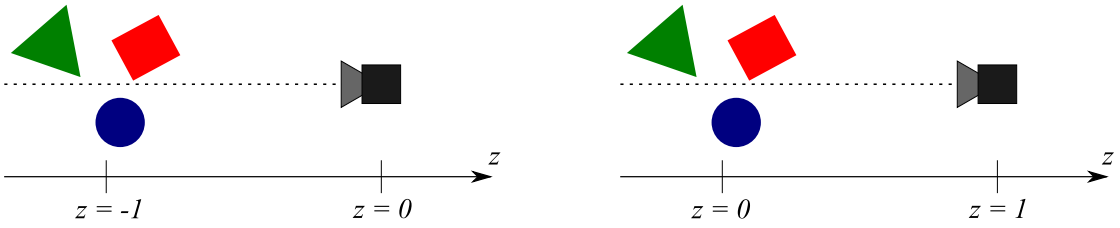


FIGURE 3.21: Relative poses in computer graphics. All transformations are relative between objects and render cameras. Displacing an object in one direction is equivalent to displacing the camera in the oposite direction.

Nevertheless, in complex scenes where many different coordinate systems exist, a global reference is helpful. Since this will be the case in this thesis, the first step is to define a global "world reference" coordinate system. This system can be considered as the reference for describing all the objects present in the scene, as well as the camera view that will be rendered on every frame.

The vertices that describe each object can be defined in terms of an object-specific local coordinate sytem. A 4×4 model matrix $\mathbf{M}_m$ defines the transformation of a vertex in local model coordinates $\boldsymbol{V}^m$ to global world coordinates $\boldsymbol{V}^w$, both expressed in homogeneous coordinates, as in Eq. 3.55. The matrix $\mathbf{M}_m$ corresponds to a Local-To-Global transformation as presented in Section 3.1.2.2.

$$\boldsymbol{V}^w = \mathbf{M}_m \cdot \boldsymbol{V}^m \tag{3.55}$$

A view matrix $\mathbf{M}_v$ also exists that defines the vertex transformation from world coordinates to the so-called camera "eye coordinates", as in Eq. 3.56. The eye coordinates are referred to the center of projection of the render camera. The view matrix corresponds to a Global-To-Local transformation as presented in Section 3.1.2.2.

$$\boldsymbol{V}^e = \mathbf{M}_v \cdot \boldsymbol{V}^w \tag{3.56}$$

Both model and view matrices are graphically described in Figure 3.22. The composite transformation of the model matrix and the view matrix usually receives the name *modelview* matrix. The modelview matrix is independent from the world reference, and only encapsulates the relative pose between camera and model.

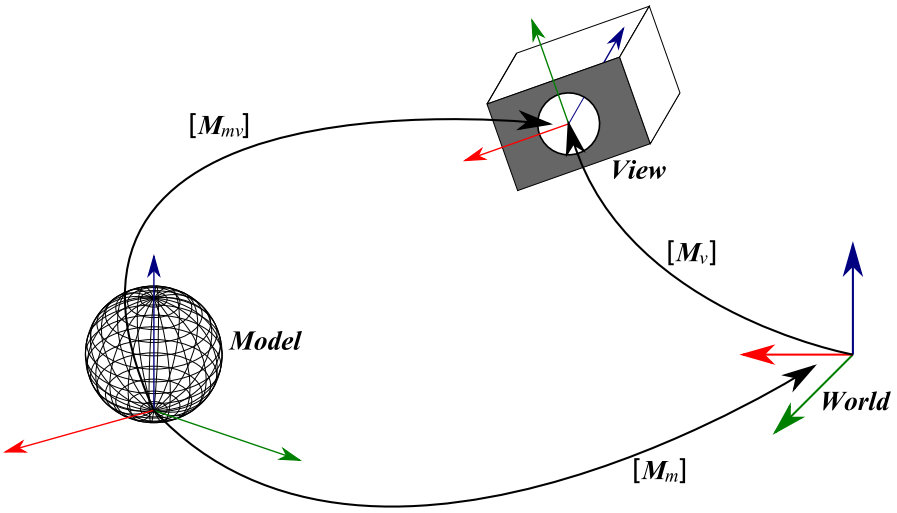


FIGURE 3.22: Modelview matrix. This matrix describes the relative transformation that exists between any object (model) and the render camera. It is independent from any global reference coordinate system.

The virtual camera used for the 3D rendering normally makes use of an ideal projection model to map eye coordinates to screen coordinates. In order to decide which vertices belong to the visible field of view, a view frustum is defined and a transformation $\mathbf{M}_p$ to clip coordinates is applied, as in Eq. 3.57.

$$\boldsymbol{V}^{\mathrm{c}} = \mathbf{M}_p \cdot \boldsymbol{V}^{\mathrm{e}} \tag{3.57}$$

The frustum does not only describe the angular field of view of the camera, but also a minimum and maximum depth.

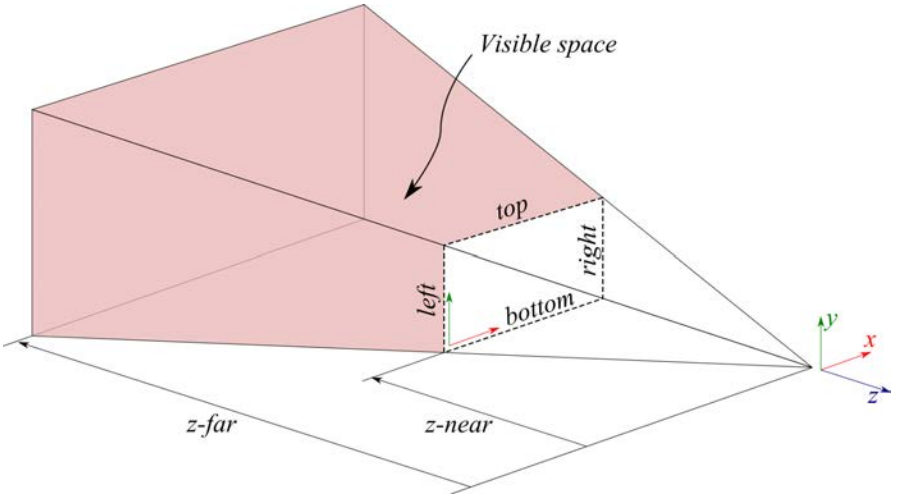


FIGURE 3.23: Perspective camera frustum. The frustum defines the volume in front of the render camera which will be rendered. Any object lying outside of it will be clipped and will not be visible on the rendered view. As an addition to the lateral field of view limits, the frustum considers a minimum and maximum depth.

In particular, for a perspective camera, OpenGL defines $\mathbf{M}_p$ as in 3.58, where l, r, t, b, n, f stand for *left, right, top, bottom, z-near, z-far* as described in Figure 3.23, respectively.

$$\mathbf{M}_p = \begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & -\frac{f+n}{f-n} & -\frac{2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{pmatrix} \tag{3.58}$$

From clip coordinates, a transformation into normalized device coordinates (NDC) is performed by means of Eq. 3.59.

$$\begin{pmatrix} x_{ndc} \\ y_{ndc} \\ z_{ndc} \end{pmatrix} = \begin{pmatrix} x_c/w_c \\ y_c/w_c \\ z_c/w_c \end{pmatrix} \tag{3.59}$$

Once the vertices are expressed in terms of NDC, they can be mapped onto the rendering viewport (in window coordinates), by means of Eq. 3.60. The viewport is expressed in pixel units and defined by its height h, width w and origin (x_0, y_0).

$$\begin{pmatrix} x_{wi} \\ y_{wi} \\ z_{wi} \end{pmatrix} = \begin{pmatrix} \frac{w}{2}x_{ndc} + (x_0 + \frac{w}{2}) \\ \frac{h}{2}y_{ndc} + (y_0 + \frac{h}{2}) \\ \frac{f-n}{2}z_{ndc} + \frac{f+n}{2} \end{pmatrix} \tag{3.60}$$

The complete OpenGL vertex operations pipeline is shown in Figure 3.24.

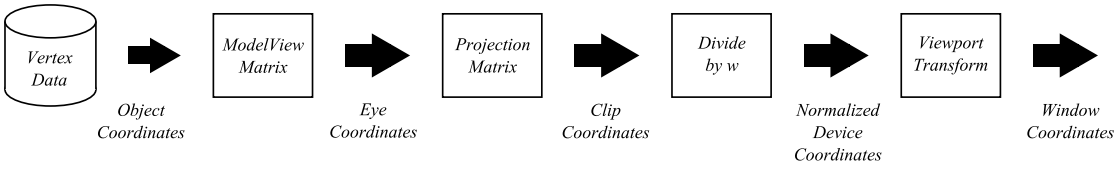


FIGURE 3.24: GL vertex operations. This diagram shows the complete pipeline through which all vertex described in the graphics application go until the final view is rendered.

In order to bring larger flexibility to the render pipeline, new OpenGL API versions allow the use of nonfixed pipelines by means of vertex and pixel shaders. This is, however, out of the scope of this thesis. For further reading on this topic, the author recommends [Rost et al., 2009].

3.3.2 Image Based Rendering

Within the large range of topics covered by the computer graphics field, the so-called Image Based Rendering (IBR) is of special relevance for this thesis. IBR makes reference to the techniques employed for generating novel views of a scene, based on real images of it. The main idea behind IBR is to combine images of a given scene with geometrical knowledge of the same scene in a way that a virtual 3D reconstruction is generated thus allowing rendering from any view point.

In order to achieve this, the real images are used as textures and the corresponding texture coordinates are calculated based on: a) the geometry, b) the position of the real camera with respect to the geometry, and c) the intrinsic parameters of the real camera itself. The vehicle's origin of coordinates is defined as the joint between the rendered and real worlds. The D70K reference is used as the top level on the hierarchy of coordinate systems for both real and virtual cameras.

The position of a real camera within a scene is considered first. In the use case relevant for this thesis, the camera pose is assumed to be known with respect to the D70K origin. A 4×4 matrix $\mathbf{M}_{c,D70K}$ exists that transform any vertex $\boldsymbol{V}^{D70K}$ expressed given in the vehicle's origin to the camera coordinate system.

$$\boldsymbol{V}^{c} = \mathbf{M}_{c,D70K}\, \boldsymbol{V}^{D70K} \tag{3.61}$$

Since the virtual world reference is defined coincident with the vehicle's origin of coordinates, every geometry vertex $\boldsymbol{V}^{m}$ that belongs to an object m can be transformed to vehicle coordinates by means of Eq. 3.62, through its model matrix $\mathbf{M}_{m}$.

$$\boldsymbol{V}^{D70K} = \mathbf{M}_{m}\, \boldsymbol{V}^{m} \tag{3.62}$$

With the previous expressions, a virtual vertex $\boldsymbol{V}^{m}$ can be transformed to the coordinate system of a real camera in the present setup, as in Eq. 3.63.

$$\boldsymbol{V}^{c} = \mathbf{M}_{c,D70K}\mathbf{M}_{m}\, \boldsymbol{V}^{m} \tag{3.63}$$

A representation of all the coordinate systems considered is shown on Figure 3.25.

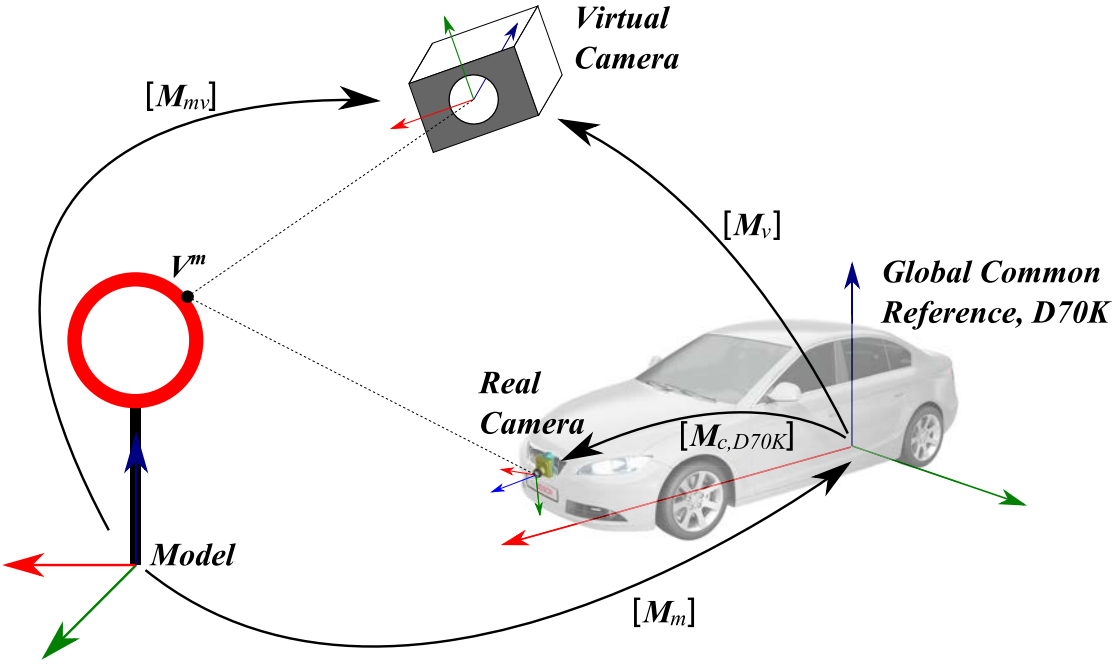


FIGURE 3.25: IBR Coordinate systems. Real coordinates are combined with virtual ones, in order to use images as perspective-correct rendering texture.

Once a vertex is expressed in the local coordines of a real camera, its projection model can be applied, obtaining its projection on the image plane. This step has been described in detail in Section 3.2.1.2. The obtained image coordinates correspond to the texture coordinates that can be assigned to the vertex prior to the rendering.

3.3.2.1 Geometry Support

In order to perform the IBR, a geometry has to be provided as projection surface. On this geometry, each vertex can be assigned texture coordinates, as has been described in the previous section. Different possibilities have been presented in Chapter 2 as an input source for the geometry information. The work presented in the following chapters of this thesis is focused on the application of computer vision techniques in order to generate the 3D information that can be used as support for the IBR.

3.3.3 A 3D Rendering Toolkit: OpenSceneGraph

For the aspects of this thesis related to computer graphics, the OpenGL API has not been considered directly. Instead, the OpenSceneGraph 3D rendering toolkit

has been used.

The OpenSceneGraph is an open source high performance 3D graphics toolkit, used by application developers in fields such as visual simulation, games, virtual reality, scientific isualization and modelling [Osfield et al., 2004]. OpenSceneGraph is written entirely in Standard C++ and OpenGL and it runs on today's most common operating systems. It offers support for animation, camera manipulation and other special effects providing large degree of scalability and portability.

Implementation details remain out of the scope of this thesis, but a very complete set of examples is available within the source code distribution. Further reading can be found in [Wang and Qian, 2010] and [Wang and Qian, 2012].

Chapter 4

Reference Sensor: 3D Laser Range Finder

In this chapter the 3D Laser Range Finder sensor is presented as an enabler for ground truth generation in automotive computer vision applications. The content of this chapter is mainly extracted from the papers [Esparza et al., 2014c], [Esparza et al., 2014b], which have been published within the framework of this thesis.

In order to bring vision-based driver assistance systems to the market, the developed image processing algorithms have to fulfill high requirements regarding functional safety and legal constraints [Stein, 2012] and need to be evaluated before putting them into a real product. Lately, 3D Laser Range Finders (LRF) were introduced as a possibility to obtain ground truth measurements in real world scenarios for testing and evaluating computer vision algorithms [Morales and Klette, 2011], [Geiger et al., 2012a]. LRFs enable to monitor the vehicle surrounding with high precision 3D measurements. Distances to reflecting surfaces are measured and these measurements are usually converted to 3D coordinates, as seen from the virtual center of the LRF. In order to use the LRF as a reference for these algorithms applied to multi-camera systems, a registration of the LRF with all the cameras in the system is needed. A description of the problem is depicted in Figure 4.1.

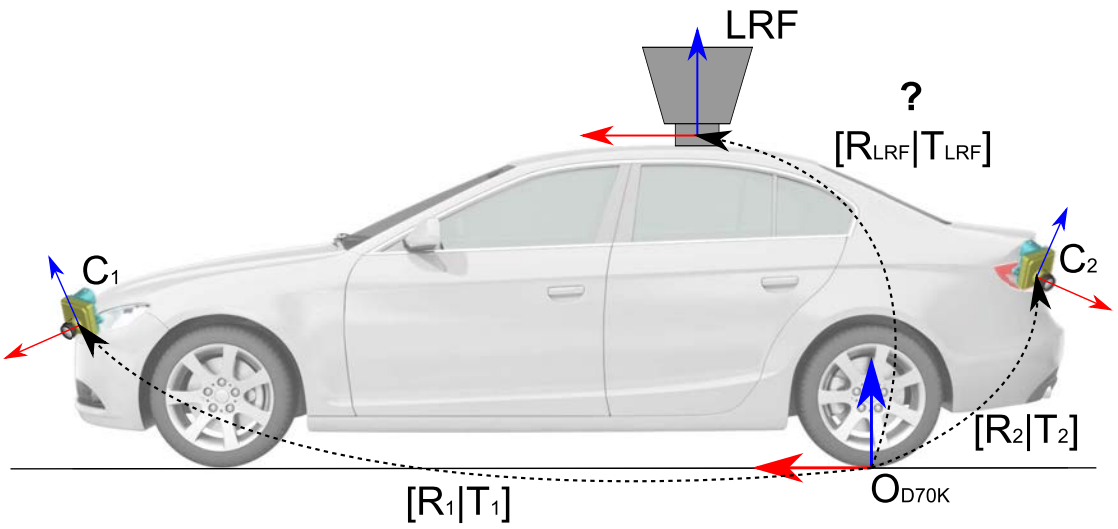


FIGURE 4.1: Registration of a 3D LRF to a multi-camera system. Assuming calibrated cameras and known extrinsic poses $[Rc \mid Tc]$ with respect to a common reference system O_{D70K}, the pose of the LRF $[R_{LRF} \mid T_{LRF}]$ can be estimated by means of external targets.

Previous work in this field coped with the registration of 2D LRFs using planar calibration patterns to perspective [Zhang and Pless, 2004] or catadiotric cameras [Mei and Rives, 2006]. These methods were extended to work with 3D LRFs [Unnikrishnan and Hebert, 2005], [Scaramuzza et al., 2007], [Pandey et al., 2010]. In [Haselich et al., 2012], multiple cameras were registered to a 3D LRF by estimating each camera position individually. In the work of [Geiger et al., 2012b], a calibration toolbox was presented that provides registration of a 3D LRF with a stereo camera system in the 3D domain by matching automatically detected calibration target planes.

The presented methods provide only a locally best solution since they are limited to determining the pose of an LRF relative to a single camera or a stereo system. Hence these approaches show a visible mismatch when evaluated on the other cameras of the system (see Figure 4.8c).

In contrast to this, a new registration method of a 3D LRF with a multi-camera system is introduced, as in the original paper [Esparza et al., 2014c]. Instead of using a single camera, a set of cameras is considered which are previously registered and referenced with respect to a global coordinate system. The LRF is then registered by minimizing the reprojection error of the LRF measurements over all cameras of the system. It is demonstrated that the global method provides a more stable global pose estimation of the LRF.

4.1 Selection of Corresponding Measurements

The proposed method relies on manual selection of corresponding points visible across different sensors, which is a critical part during the registration process. In order to ease the correspondence search, a calibration target has been designed. A picture of the custom target is shown in Figure 4.2.

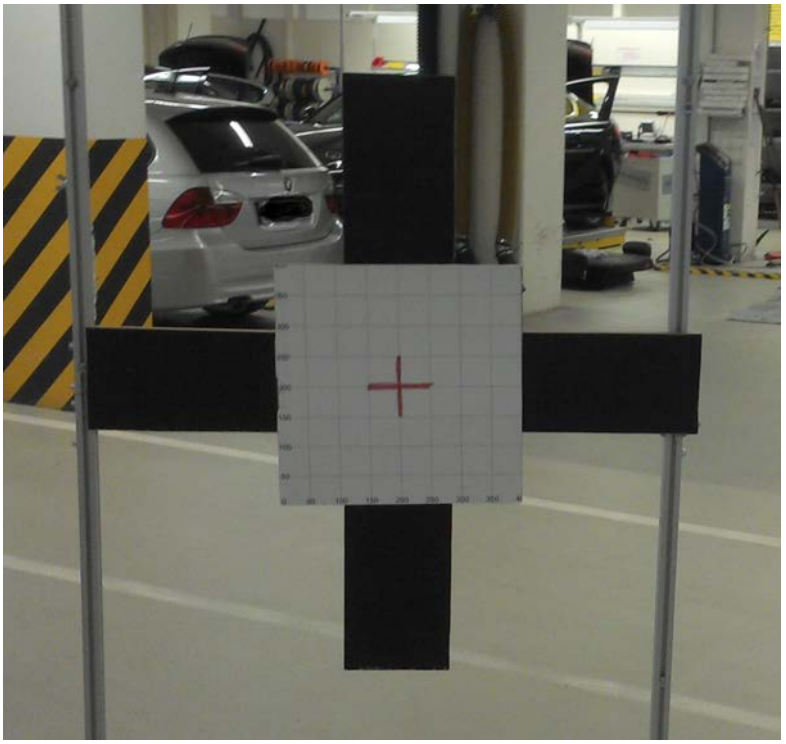


FIGURE 4.2: Utilized target. It is designed such that its color properties make it distinguishable on camera images and, simultaneously, its geometry is clearly recognizable in the range data from the LRF.

This target is constructed such that its color properties make it distinguishable on camera images and, simultaneously, its geometry is clearly recognizable in the range data from the LRF, as highlighted in Figure 4.3.

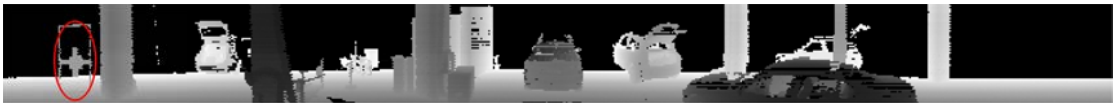


FIGURE 4.3: Example of rasterized depth map, as seen from the 3D LRF virtual center. In red: Target used for manual selection of keypoints. Taken from [Esparza et al., 2014c].

Before registration, keypoints are manually labelled, each label consisting of the pixel position of the target's center in the camera image and the corresponding 3D coordinate measured by the LRF.

Since manual inspection of 3D data is complex and tedious, a 2D depth map based on range information is generated in order to ease the task. For rotating LRFs, a set of key directions can be defined, and the measurements closest to these are taken. This generates a full 2D matrix that encodes depth which can be rastered as an image. This process resembles that of a cylindric warping of an image sequence acquired by a camera panning 360°. An example of the resulting view is shown in Figure 4.3. Although this preparation step is used for simplicity of data inspection, the original 3D measurements are to be considered during the whole registration process.

4.2 Registration of 3D LRF

The extrinsic calibration is based on the minimization of the global reprojection error of key geometrical points onto the different camera images. Keypoints are selected by means of the proposed target in a way that the whole surrounding of the vehicle is covered. Each keypoint shall be selected such that it is present on the LRF data and on, at least, one camera image. Also all cameras should contribute to keypoints, since the method is aimed at a global solution that minimizes the reprojection error over all cameras. For each keypoint, a measurement $\mathbf{S}$ is considered, as in Eq. 4.1.

$$\mathbf{S} = \left\{ \mathbf{U}^k, \boldsymbol{P} \right\} \tag{4.1}$$

In Eq. 4.1, k represents the camera index, $\boldsymbol{P} = (X, Y, Z)^T$ the measured 3D position of the keypoint with respect to the LRF centre and $\mathbf{U}^k$ a 3D-vector defined as the direction on which the keypoint lies, as observed from the virtual center of camera k. $\mathbf{U}^k$ can be calculated from the pixel coordinates $\boldsymbol{u}^k = (u, v)^k$ corresponding to the keypoint in camera k, given that the cameras are intrinsically calibrated. This is done according to the model proposed in [Mei and Rives, 2007], as described in Section 3.2.1.2. The generation process for measurements $\mathbf{S}$ is shown in Figure 4.4.

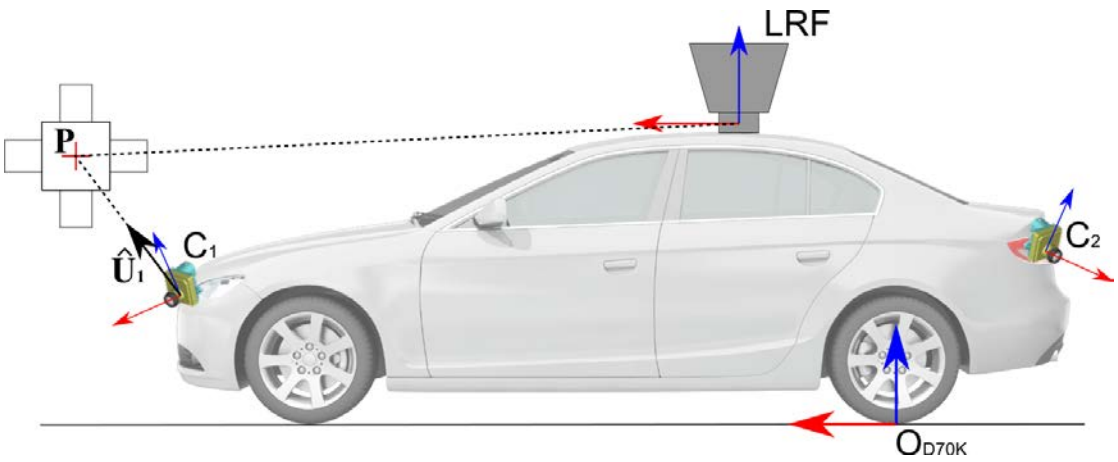


FIGURE 4.4: Generation of reference measurements **S** for the registration process. A target is considered and its position with respect to each sensor is used. For the LRF, the scale-correct 3D coordinates are used. For the cameras, the view rays are used.

A set of parameters $\theta = \{X_L, Y_L, Z_L, \alpha_L, \gamma_L, \beta_L\}$ is assumed that describes the pose of the LRF with respect to the vehicle's coordinate system. Based on this set of parameters, a 4×4 translation and rotation matrix $\mathbf{M}_{D70K,LRF}(\theta)$ can be defined, such that a point $\boldsymbol{P}$ observed from the LRF origin, can be described as another point $\boldsymbol{Q}(\theta)$, with respect to the vehicle's origin, by means of Eq. 4.2.

$$\begin{pmatrix} \boldsymbol{Q}(\theta) \\ 1 \end{pmatrix} = \mathbf{M}_{D70K,LRF}(\theta) \cdot \begin{pmatrix} \boldsymbol{P} \\ 1 \end{pmatrix} \tag{4.2}$$

Similarly, for any given point $\boldsymbol{Q}$ represented in vehicle coordinates, a transformation $\mathbf{M}_{k,D70K}$ to local camera frame coordinates, namely $\mathbf{V}^k$, can be applied by means of Eq. 4.3. For every camera k, the transformation matrix $\mathbf{M}_{k,D70K}$ is known, given the previous registration of the multi-camera system with respect to the global origin of the vehicle.

$$\begin{pmatrix} \mathbf{V}^k \\ 1 \end{pmatrix} = \mathbf{M}_{k,D70K} \cdot \begin{pmatrix} \boldsymbol{Q} \\ 1 \end{pmatrix} \tag{4.3}$$

By combining Eqs. 4.2 and 4.3, it is now possible to describe any point $\mathbf{P}$ observed by the LRF with respect to the coordinate system of any camera k as function of the parameter set θ, as in Eq. 4.4.

$$\begin{pmatrix} \mathbf{V}^k(\theta) \\ 1 \end{pmatrix} = \mathbf{M}_{k,D70K} \cdot \mathbf{M}_{D70K,LRF}(\theta) \cdot \begin{pmatrix} \boldsymbol{P} \\ 1 \end{pmatrix} \tag{4.4}$$

For each keypoint $\mathbf{S}_i$ and for a given LRF pose estimate θ, an error measure $e_i(\theta)$

can be defined, as the angular distance between $\mathbf{V}_i^k$, and $\mathbf{U}_i$ seen from camera k's center. Considering $\hat{\mathbf{V}}_i^k$ and $\hat{\mathbf{U}}_i$ as the normalized $\mathbf{V}_i^k$ and $\mathbf{U}_i$ respectively, $e_i(\theta)$ can be obtained by means of Eq. 4.5.

$$e_i(\theta) = \arccos\left(\hat{\mathbf{V}}_i^k(\theta)^T \cdot \hat{\mathbf{U}}_i\right) \tag{4.5}$$

In the described setup, as in the work of [Scaramuzza et al., 2007], the angular error is more suitable for characterising the quality of the pose estimation than the pixel error. This is due to the non uniform pixel density over the whole FoV on cameras with fish-eye optics, as was shown in Section 3.2.1.2.

Using non-linear search algorithms, a set of parameters $\tilde{\theta}$ can be obtained, by minimizing the cost function over all reference keypoints, as proposed in Eq. 4.6

$$\tilde{\theta} = \underset{\theta}{\operatorname{argmin}} \frac{1}{2} \sum_i e_i^2(\theta) = \underset{\theta}{\operatorname{argmin}} \frac{1}{2} \sum_i \arccos^2\left(\hat{\mathbf{V}}_i^k(\theta)^T \cdot \hat{\mathbf{U}}_i\right) \tag{4.6}$$

such that the overall reprojection error is minimum over all cameras on the system.

4.3 Results of the 3D LRF Registration

For evaluating the proposed method, a vehicle equipped with a Velodyne HDL-64E S2 LIDAR and a multi-camera system in a common surround-view configuration (as in Section 2.1.1) was used. The Velodyne LIDAR provides a full 360° field of view on the horizontal direction and a field of view of 26.8° on the vertical direction, distributed over 64 independent laser beams. This accounts for a vertical angular resolution equivalent to approximately 0.43° per laser beam. As described in Section 4.1, a rasterization of the depth map is generated for the sake of data inspection. For this, horizontal angular steps of 0.5° are considered.

The multi-camera system consists of four fish-eye cameras with a horizontal FoV of 185° and a resolution of 1280 × 960 pixels. The cameras were mounted on the front and rear ends of the vehicle, as well as on the left and right external mirrors, as on the configuration presented in Section 2.1.1. A set of 240 measurements (60 measurements per camera) was manually labelled, each comprising the camera index, the 3D position of the target measured from the LRF center and its corresponding 2D pixel coordinates in the camera image, as defined in Eq. 4.1. For the

measurements, the positions of the target were chosen such that different ranges are covered, while posing a uniform distribution over the FoVs of all cameras.

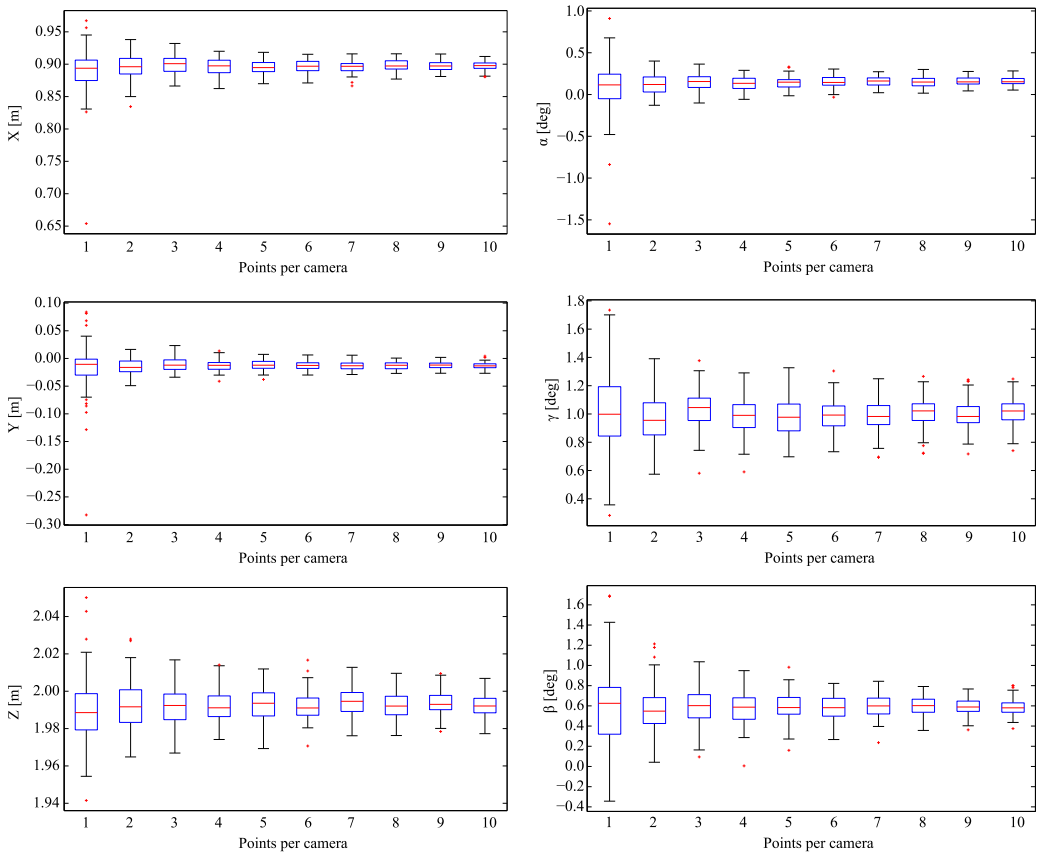
In order to evaluate the proposed method, focus has been set on two aspects. First, evaluation of the robustness of the presented approach over multiple runs with different measurements. Second, evaluation of the reprojection error of the 3D LRF measurements onto the image plane of the cameras and comparison the results to a single-camera registration method, as proposed in [Scaramuzza et al., 2007].

4.3.1 Evaluation of the Robustness of the Pose Estimation

The robustness of the presented approach is evaluated by examining the variation of the estimated pose parameters over multiple repetitions. In particular, this is aimed at evaluating the stability of the parameters, depending on the amount of measurements used for the pose estimation. Hence the number of keypoints used for the registration of the LRF was varied between 1 and 10 per camera (between 4 and 40 in total). This process was repeated 100 times. Figure 4.5 shows the distribution of the six pose parameters over different iterations. Careful analysis of these results shows that the estimation of the parameters becomes more stable when the amount of reference keypoints is increased. However, the variance of all parameters does not decrease significantly after a certain point. It can be concluded that with approximately 5 measurements per camera (20 measurements for the whole system) a stable pose can be estimated. A good selection of keypoints may further reduce the number of measurements needed. Table 4.1 shows the average estimated position and orientation of the LRF considering 5 reference points per camera.

$\bar{X}$ [m]	σ_X [m]	$\bar{Y}$ [m]	σ_Y [m]	$\bar{Z}$ [m]	σ_Z [m]
0.894	0.010	-0.012	0.009	1.993	0.009
$\bar{\alpha}$ [deg]	σ_α [deg]	$\bar{\gamma}$ [deg]	σ_γ [deg]	$\bar{\beta}$ [deg]	σ_β [deg]
0.141	0.069	0.981	0.135	0.592	0.129

TABLE 4.1: Averaged parameter estimation of the 3D LRF pose over 100 repetitions, considering 5 reference points per camera. Data extracted from [Esparza et al., 2014c].



(A) Variance of position parameters (B) Variance of orientation parameters

FIGURE 4.5: Variance of position (a) and orientation (b) parameters of the 3D LRF registration based on the number of measurements per camera. Taken from [Esparza et al., 2014c].

4.3.2 Evaluation of the Reprojection Error

The reprojection error represents a good quality measure to describe the accuracy of the estimated parameters that define the pose of the LRF with respect to the global coordinate system [Hartley and Zisserman, 2000]. It can be expressed in terms of pixel error or angular error. Although the angular error may be a more correct metric given the non-uniform distribution of pixels over the whole field of view of the cameras, the pixel error is also analyzed since it is an important figure for data association and in data fusion systems [Scaramuzza et al., 2007]. A test set was defined, consisting of 40 measurements uniformly distributed over all cameras in the system. The evaluation was performed by reprojecting the 3D LRF measurements of the test set into the associated camera image using the estimated pose as in Eq. 4.4. The results were compared with manually labelled reference data.

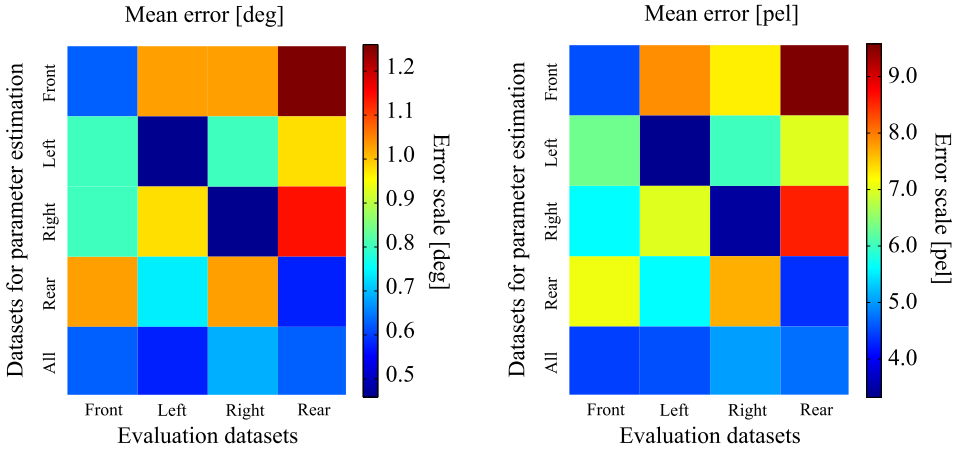
Based on the results presented in Sec. 4.3.1, five measurements for each of the four cameras (20 measurements in total) were considered for the registration of the LRF to the multi-camera system. Table 4.2 shows the average errors obtained.

$\bar{e}$ [deg]	σ_e [deg]	$\bar{e}$ [pel]	σ_e [pel]
0.624	0.285	4.649	2.308

TABLE 4.2: Reprojection error after registration of the 3D LRF with the multi-camera system. Data extracted from [Esparza et al., 2014c].

Given the angular resolutions of the LRF data which was considered for the experiments (namely 0.43° and 0.5° for the vertical and horizontal directions) the resulting 0.624° error can be considered a good result.

As reference, the results are compared with a single-camera registration as proposed in [Scaramuzza et al., 2007]. The LRF was registered to every camera in the system individually using 20 measurements for the pose estimation. Evaluation is in every case performed over all cameras of the system. For all the experiments, the registration process was repeated 100 times. In Figure 4.6, the mean of the reprojection error is shown for each combination of datasets used for pose estimation and evaluation.



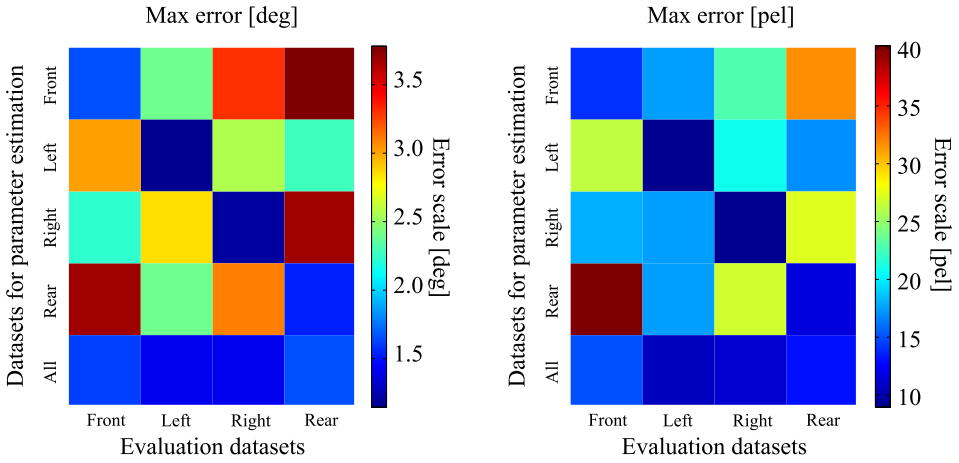
(A) Mean reprojection error in degrees (B) Mean reprojection error in pixels

FIGURE 4.6: Comparison of the mean reprojection error in degrees (a) and in pixels (b) of different single and a multi-camera LRF registration. The error is depicted for every camera individually. Taken from [Esparza et al., 2014c].

As expected, it can be observed that the cases where registration and evaluation are conducted with data from the same single camera produce the smallest error. One

can also notice that the error increases strongly if the estimated pose is evaluated on any other cameras of the system. Evaluation on the cases where all cameras are used for the registration, on the other hand, produces a much smaller reprojection error over all cameras in the system. This global best is only slightly outperformed by the cases where both pose estimation and evaluation are performed with data of the same camera individually.

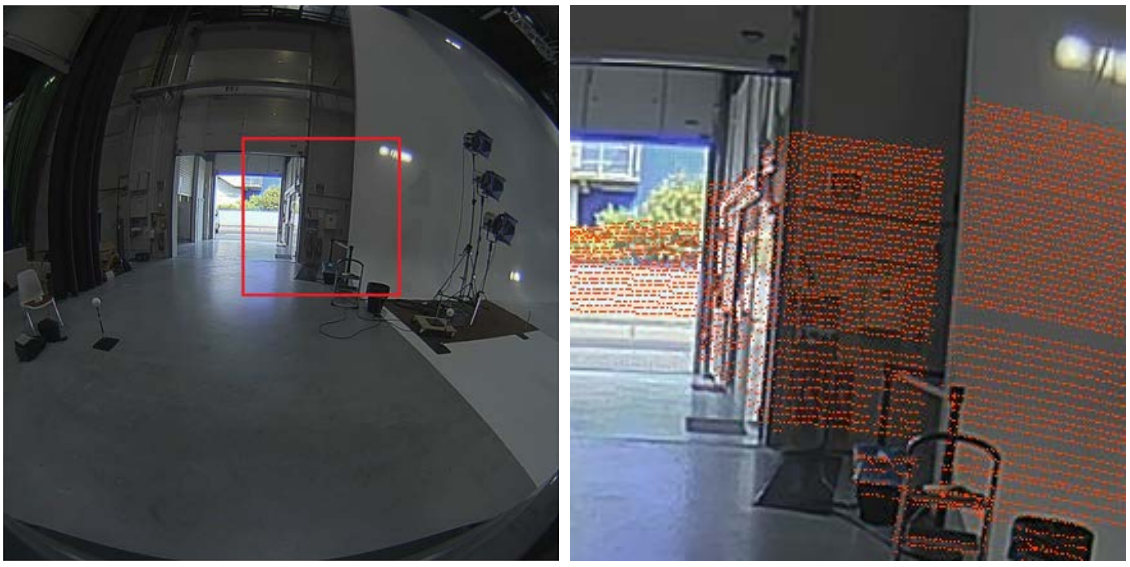
Although the average reprojection error is a good indicator of for the accuracy of the registration, it is also interesting to look into the maximum reprojection error found for each case. This approximates the worst-case scenario and defines an upper bound limit to the magnitude of the errors which are to be expected. Results are shown in Figure 4.7.



(A) Max. reprojection error in degrees (B) Max. reprojection error in pixels

FIGURE 4.7: Comparison of the maximal reprojection error in degrees (a) and in pixels (b) of different single and a multi-camera LRF registration. The error is depicted for every camera individually. Taken from [Esparza et al., 2014c].

A graphical comparison of the results obtained after registration of the LRF globally with the multi-camera system, and with a single camera is presented in Figure 4.8. The improvement of the proposed global multi-camera approach with respect to a single-camera registration is clearly visible by comparison of Figure 4.8c and Figure 4.8d.



(A) (B)

(C) (D)

FIGURE 4.8: Reprojection of LRF points using different input for registration. The quality of the registration can be clearly observed by comparing the edges of the walls to the reprojected points. a) Original image. In red: area selected for visualization. b) Registration using rear camera only - LRF points projected onto rear camera image. c) Registration using front camera only - LRF points projected onto rear camera image. d) Registration using all cameras - LRF points projected onto rear camera image. Taken from [Esparza et al., 2014c].

4.4 Ground Truth Generation for 3D Measurements

In order to use the 3D measurements from the LRF as ground truth, an error metric has to be defined that can be applied to the depths obtained by means of

computer vision algorithms.

In the following it is proposed to use the image plane as a common domain, where visual comparison is possible and an error metric can be defined for the accuracy of the measured distances. The 3D measurements from the lidar and obtained by means of CV algorithms are projected onto the image planes and compared. Figure 4.9 shows an overview of the complete setup utilized.

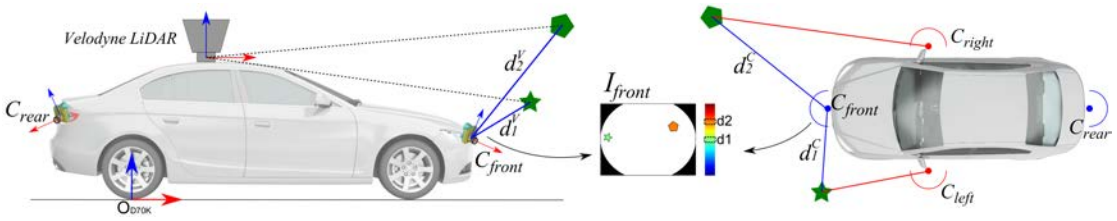


FIGURE 4.9: Setup overview. Left: Configuration for reference measurements. The lidar was registered to the surround-view system, so that 3D measurements can be re-projected onto any image plane and used as reference. Right: CV-based measurements to be evaluated. Taken from [Esparza et al., 2014b].

Since the measurements given by the lidar are very sparse in the vertical direction - it has a vertical resolution of 64 lasers, compared to the 960 pixels of the camera imager - each measurement is thickened after reprojection, in order to become a denser depth reference. In particular, a 20-pixel high mask is used to achieve this.

As an error metric, the differences between the pixel depth values obtained by CV algorithms and the nearest lidar measurement projected on the proposed common domain are considered. A set C is considered of 3D measurements obtained by means of any image-based 3D reconstruction approach to be evaluated, and a set V of 3D points given as a result of the lidar measurements. The distance d^{V} of each point in V to the camera center can be computed since the lidar has been registered to the vehicle's coordinate system, as in Section 4.2. An L1-norm error metric can be now defined as in Eq. 4.7.

$$e_d = \frac{1}{\dim(C \cap V)} \sum_{i \in (C \cap V)} \left| d_i^{\mathrm{V}} - d_i^{\mathrm{C}} \right| \tag{4.7}$$

Figure 4.10 shows an example of the ground truth generated for the front camera measurements, where color encodes distance to the camera center of each lidar measurement.

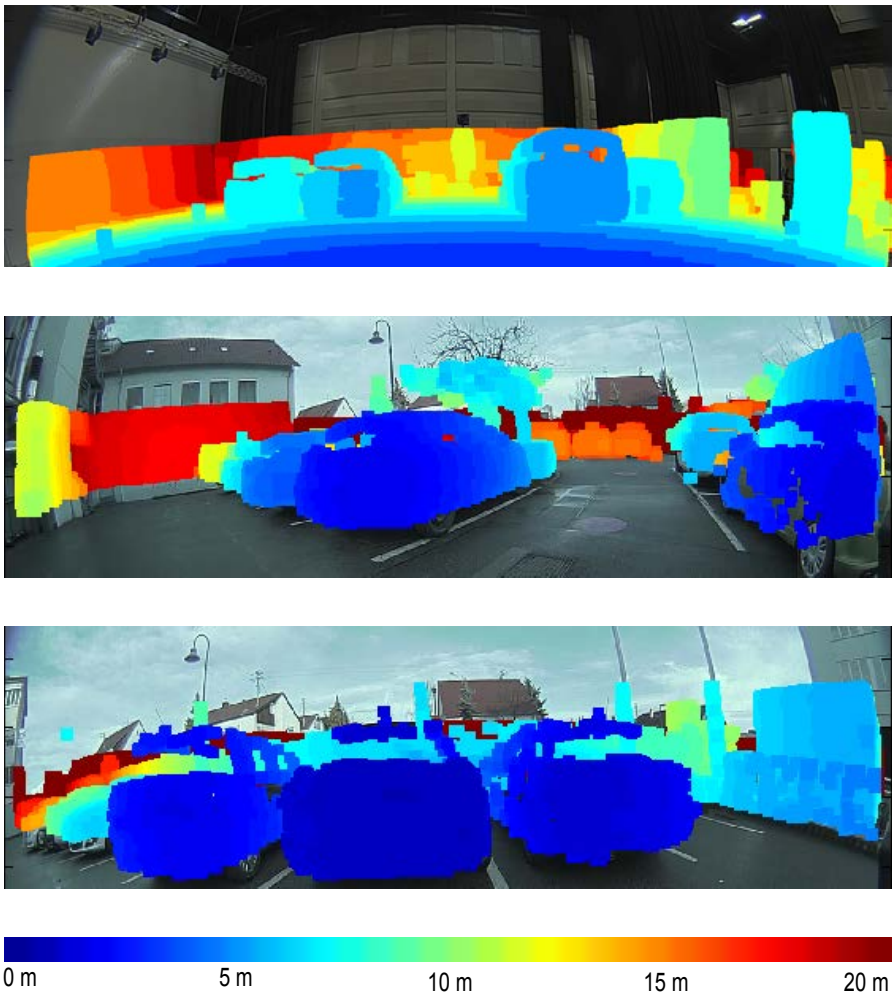


FIGURE 4.10: Ground truth generation based on lidar. The reference measurements backprojected onto front camera image. Color encodes distances to camera center. Since the floor is considered as reference for the multi-camera registration, it can be considered for the ground truth or filtered out, as shown in the center and bottom images.

4.5 Conclusions

A new approach for global registration of a 3D LRF to a multi-camera system has been proposed. The proposed approach shows improved results compared to a state of the art LRF registration with respect to a single camera, if evaluated for the multicamera system.

The robustness and stability of the developed method have been demonstrated and an L1-norm error metric has been proposed for evaluation of CV-based measurements, considering the reprojection of reference measurements to the camera image planes.

The proposed error metric will not cover the entire measurement space of an automotive CV system, but it covers the most relevant fraction of it for driving assistance applications.

Chapter 5

3D Measurements with Automotive Surround View Systems

This chapter is aimed at describing the proposed 3D reconstruction scheme and evaluating the accuracy of the 3D measurements achieved by means of stereo vision with fisheye surround view cameras. Benchmarking of the measurements is done with respect to a 3D LRF, which data requires registration to the considered multicamera system. This process has been described in detail in Chapter 4.

The chapter is organized in three main sections, which correspond to three different groups of experiments carried out within the framework of this thesis.

5.1 3D Reconstruction with Fisheye Optics

This section addresses the main considerations with respect to performing 3D stereo reconstruction based on automotive surround view camera systems. These setups put very hard restrictions on the stereo processing, namely, very large stereo bases between adjacent cameras, extreme misalignments of the optical centers and very severe distortions due to the fisheye optics.

The 3D stereo reconstruction process is analyzed with respect to different factors. Firstly, the precision of the feature detection and matching together with the accuracy of the camera synchronization are discussed for large stereo bases. Analysis of the existing overlaps in the camera fields of view is conducted and epipolar

rectification process is described accordingly. Finally, experimental setup and results are presented. The content of this section has been mainly extracted from [Esparza et al., 2014b], which was published within the framework of this thesis.

5.1.1 Precision of Keypoint Detection

There exist large amounts of computer vision applications for which high accuracy 3D reconstruction is a very important requirement. For such applications, the accuracy of the keypoint detectors plays a big role, with state of the art keypoint detectors reaching sub-pixel accuracies [Ke and Sukthankar, 2004]. For the proposed application, a certain tolerance on the accuracy of the 3D measurements is acceptable, and therefore in the following it is studied what the effects are, of considering a keypoint detector with an accuracy of 1 pixel at best.

For the following discussion, the stereo camera pair depicted in Figure 5.1 is considered, with cameras covering a FoV of 180° horizontally, and a resolution of 1280×960 pixels.

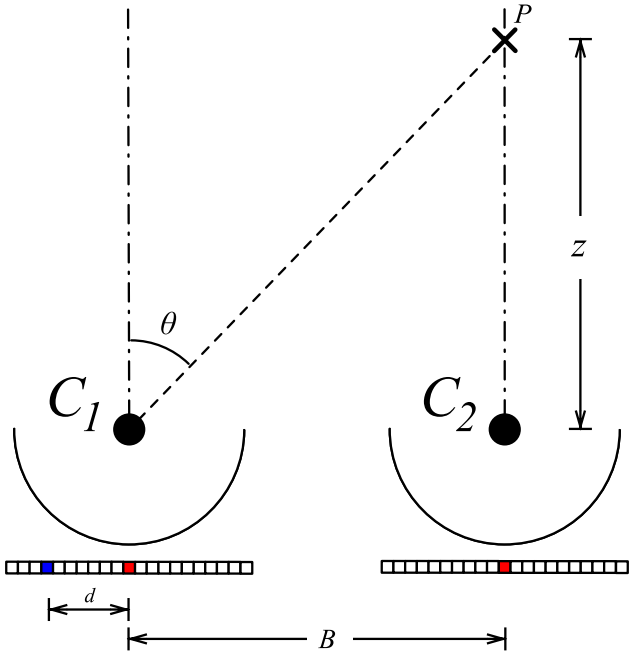


FIGURE 5.1: Typical stereo setup with fisheye optics. B represents the stereo base, and z the distance to the projection center of camera C_2. d stands for the disparity of the projection of point $\boldsymbol{P}$ on both images. θ is the angle defined by the optical axis of camera C_1 and the projection of $\boldsymbol{P}$.

Under this setup, the relation in Eq. 5.1 exists.

$$\tan\theta = \frac{B}{z} \tag{5.1}$$

In Eq. 5.1, B represents the base between both camera centers, z is the distance to point $\boldsymbol{P}$, assuming it lies on the optical axis of camera C_2, and θ is the angle formed by the projection of $\boldsymbol{P}$ into the center of camera C_1 and its optical axis. At this point, it is important to note that the standard pinhole expression in Eq. 5.2 that relates disparity d and focal length f does not hold for this setup, since cameras with fisheye optics are being considered.

$$\tan\theta = \frac{d}{f} \tag{5.2}$$

For the considered optics an equidistant mapping function is more suitable, as in Eq. 5.3, where $K\,[^{pel}/_{rad}]$ depends on the imager and optics characteristics. Other projection models for fisheye optics have been presented in more detail in Section 3.2.1.2.

$$d = K \cdot \theta \tag{5.3}$$

For a camera with a horizontal field of view of 180 degrees and an image width of 1280 pixels, K has a value of $K = {}^{1280}/_{\pi}\,[^{pel}/_{rad}]$.

Bringing Eqs. 5.1 and 5.3 together leads to Eq. 5.4 that relates the estimated depth to the pixel disparity, for given $B = B_0$ and $K = K_0$.

$$z\,(d) = B_0 \cdot \frac{1}{\tan\left[{}^{d}/_{K_0}\right]} \tag{5.4}$$

In Figure 5.2 it is shown that, for a given target depth, a wider stereo base provides a finer sampling. In other words, the wider the stereo base, the smaller becomes the distance step represented for each *pel* of disparity. This was described in [Okutomi and Kanade, 1993] as a magnification effect on the disparity, due to the stereo base. The authors also pointed out the large disparities as a new source of error, since incorrect matches are more prone to happen, the bigger the search range is. This is, however, out of the scope of this discussion, since the calculations assume correctness in the keypoint matching.

In the following a generic keypoint detector is considered with an accuracy of 1 pixel at best. This detector introduces a maximum quantization error of $1/2$ pixels. For such a maximum error, Figure 5.2 shows the maximum uncertainty on the depth estimation to be expected for different stereo bases.

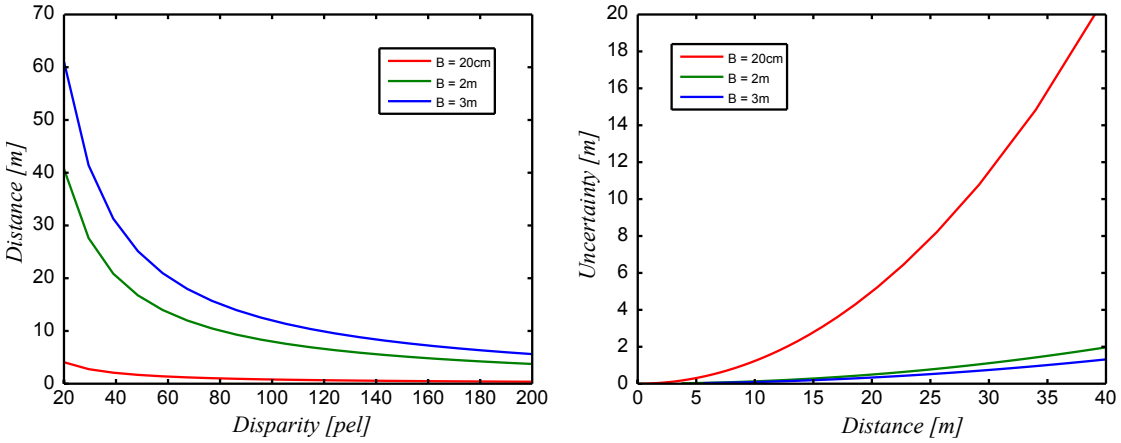


FIGURE 5.2: Left: Distance estimates, based on image disparities for different stereo bases. Right: Uncertainty due to one pixel accuracy at best, with respect to target depths. Expected error is shown for different stereo base lengths. Note that the calculations are done with fisheye optics, therefore the large uncertainties for narrow stereo bases.

This relationship can also be evaluated with respect to the distance. If a target distance is set, the maximum error for such a distance can be estimated, considering the quantization error as the only source of errors, and no mismatches. The behaviour of this error with respect to the target distance is characterized in Figure 5.2.

Based on this discussion, it is observed that for the proposed setup, at a distance of up to 30 meters the expected error is within 1 meter, due to the coarse precision of the keypoint detection. This seems acceptable for the accuracies required by driving assistance functions. Therefore, it seems reasonable not to opt for high accuracies on keypoint detection, but rather to work on a coarser level, for the proposed wide stereo base setup. It can also be observed, that for the large stereo bases considered, a disparity of 200 pixels corresponds to distances of approximately 6 meters.

5.1.2 Impact of Temporal Jitters with Large Stereo Bases

It is common understanding that traditional stereo vision with small stereo bases requires high accurate synchronization in order to keep errors on depth estimation low. This is a restriction which may considerably increase costs and complexity of the acquisition system, since a camera triggering system is required. In the following it is argued in favour of systems with a wide stereo base, where synchronization can be relaxed since the impact on accuracy is largely reduced.

A maximum temporal jitter of half a frame period is assumed. In the extreme case, the motion of the vehicle would be such, that one camera center is moving over the line joining both epipoles.

Under such a motion pattern, the lack in synchronization would generate, at most, a maximum camera displacement equal to $\Delta B = v \cdot \Delta t$, where v represents the magnitude of the velocity and Δt is the maximum time offset between image pairs. This is illustrated on Figure 5.3.

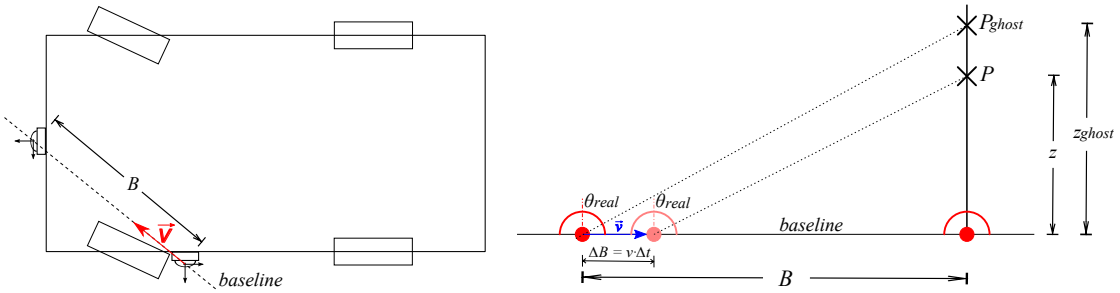


FIGURE 5.3: Left: The maximal relative motion between cameras of a stereo pair, is such that the movement is along the line joining both camera centers. Right: Maximal displacement. A ghost point is generated, since the assumed distance B between cameras is not valid, due to the relative motion.

The effect of the time shift of images of one camera with respect to the other can result in the effective stereo base being different to the geometric stereo base. This, as a side effect, creates a ghost measurement z_{ghost}, as depicted in Figure 5.3. The reason why z_{ghost} exists, is due to the fact that the calibrated stereo base B is assumed to be valid, although the real effective base in this situation is equal to $(B - \Delta B)$. In this case, the error being made can be estimated by means of Eq. 5.5.

$$e_z = z_{ghost} - z_{real} = \frac{B}{\tan\theta_{real}} - \frac{(B - \Delta B)}{\tan\theta_{real}} = \frac{\Delta B}{\tan\theta_{real}} = z \cdot \frac{\Delta B}{(B - \Delta B)} \tag{5.5}$$

Based on Eq. 5.5, it is possible to visualize the expected maximum depth error due to lack of synchronization for different depth ranges and stereo bases. These can be seen in Figure 5.4.

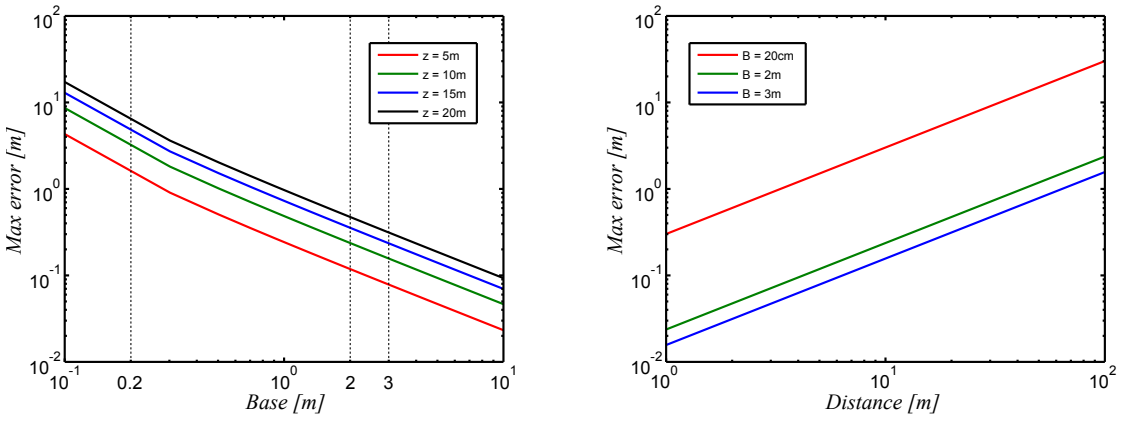


FIGURE 5.4: Error analysis due to loose synchronization. The magnitude of the velocity considered is $v = 10^{km}/_h$ and the imager frame rate equal to $30fps$, which translates to a maximum $\Delta B = 46mm$, assuming a maximum synchronisation shift of $^1/_2$ frame

These results estimate, for a stereo base equal to $3m$, an error below $0.5m$ at a distance of $30m$. This error is comparable to that expected due to the quantization error of the keypoint detector. Therefore, it can be concluded that the use of wide stereo bases introduces certain levels of robustness for the 3D reconstruction against loose synchronization of the camera pairs.

5.1.3 Description of the Overlapping Fields of View

On the current system configuration, given the strong deviation of the optical axes, the field of view of each camera of a camera pair overlap only partially. Therefore stereo processing is only meaningful on the fraction of the images that correspond to this area.

In the following, a simple two dimensional model of the camera setup is proposed, in order to describe the effective field of view of each camera pair. All camera centers are assumed to be contained on a single plane parallel to the ground plane, and the maximum field of view of the cameras to be contained on this plane. This is a reasonable assumption since the considered camera setup fulfills the condition that $\|C_{z,L} - C_{z,R}\| \ll \|C_{xy,L} - C_{xy,R}\|$, where C_L and C_R represent the positions of the Left and Right cameras on each stereo pair.

Based on this assumption, and with known intrinsic and extrinsic calibration, Eq. 5.6 can be defined that expresses the effective field of view of the left camera; i.e. the fraction of its own field of view that may overlap with the field of view of

its adjacent right camera.

$$\left[\psi^{-}, \psi^{+}\right]_{L} \approx \left[-\arccos\left(\hat{\mathbf{O}}_{xy,L}^{T} R_{F/2} \hat{\mathbf{O}}_{xy,R}\right), {}^{F}/_{2}\right] \tag{5.6}$$

In this expression, $R_{F/2}$ represents a 2D rotation matrix over the normal $\hat{\mathbf{e}}_z$ of the ground plane, of magnitude equal to half of the field of view of the cameras. $\hat{\mathbf{O}}_{xy}$ stands for the normalized projection of the optical axis of each camera over the XY plane. A similar expression can be obtained for the right camera.

This model provides an estimate of the amount of overlap existing for each pair of adjacent cameras. This approximation is only valid as long as the 2D optical axes of the cameras do not intersect in front of both image planes. This situation is not possible on the discussed setup, since the cameras are mounted on each side of the vehicle, looking outwards. From the previous expresion, it can be inferred that the field of view of the resulting virtual cameras will be asymmetric with respect to their principal points. In the following this effect is analyzed.

The optical axis $\hat{\mathbf{O}}_V$ of the virtual camera is defined by the principal point. It is possible to describe $\hat{\mathbf{O}}_V$ by means of Eq. 5.7, where $\hat{\mathbf{t}}_{C_R C_L}$ is the normalized vector joining both camera centers.

$$\hat{\mathbf{O}}_V = \hat{\mathbf{t}}_{C_R C_L} \times \hat{\mathbf{e}}_z \tag{5.7}$$

Based on the previous definitions, the asymetric field of view $[\psi^{-}, \psi^{+}]_V$ of the rectified virtual cameras can be described by means of Eq. 5.8.

$$\left[\psi^{-}, \psi^{+}\right]_{V} = \left[-\arccos\left(\hat{\mathbf{O}}_{xy,V}^{T} R_{F/2} \hat{\mathbf{O}}_{xy,R}\right), \arccos\left(\hat{\mathbf{O}}_{xy,V}^{T} R_{F/2}^{-1} \hat{\mathbf{O}}_{xy,L}\right)\right] \tag{5.8}$$

In this expression, $\hat{\mathbf{O}}_{xy,V}$ represents the normalized projection of $\hat{\mathbf{O}}_V$ over the XY plane. Figure 5.5 depicts how the principal point is largely displaced with respect to the center of the field of view shared by a camera pair.

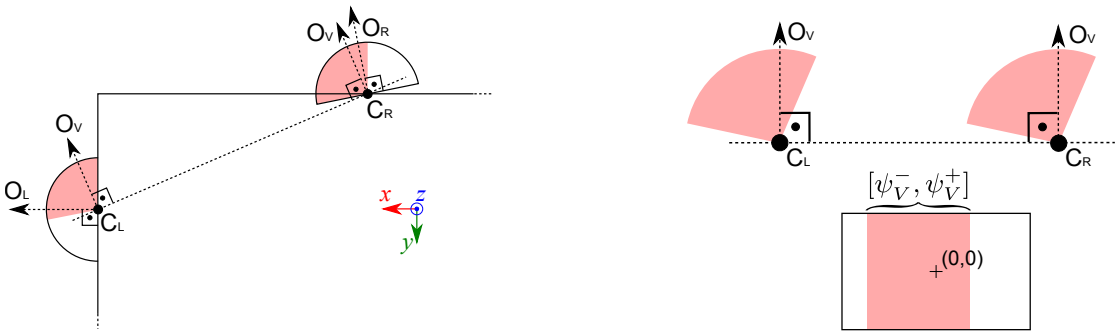
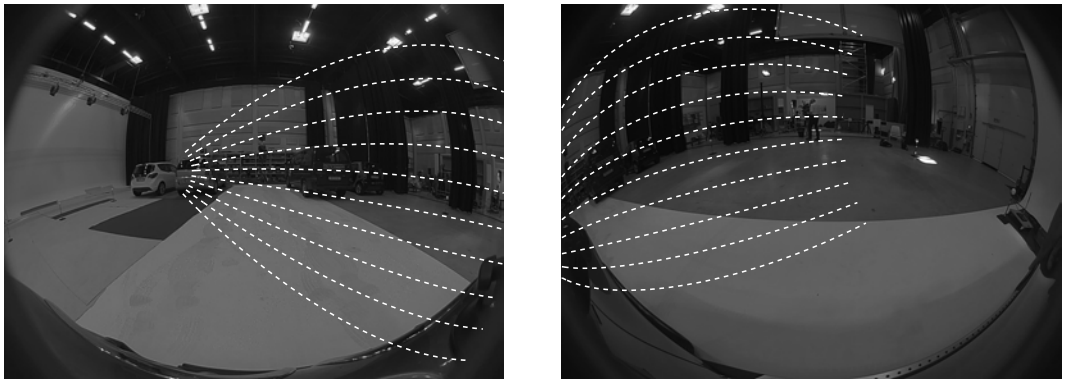


FIGURE 5.5: Overview on the stereo pair overlaps. Left: 2D representation of overlapping fields of view for 2 adjacent cameras. Right: Nonsymmetric rectified fields of view with respect to the principal point. Taken from [Esparza et al., 2014b].

5.1.4 Epipolar Rectification with Fisheye Optics

Once the shared field of view of adjacent cameras is defined, the epipolar rectification model can be introduced. It is well understood that fisheye optics introduce distortions such that epipolar planes do not project into the image planes as straight lines [Abraham and Förstner, 2005], [Herrera et al., 2009]. This effect can be seen in figures 5.6a and 5.6b.



(A) Epipolar lines on front camera image

(B) Epipolar lines on right camera image

FIGURE 5.6: Example of epipolar lines with fisheye optics corresponding to a front - right camera pair. Taken from [Esparza et al., 2014b].

In particular, considering the use of fisheye cameras, the linear transformation presented in Eq. 3.50 has to be replaced by a nonlinear transformation.

As discussed earlier in Section 3.2.1.2, the camera projection model proposed in [Mei and Rives, 2007] is considered to represent $\mathbf{T}_C$, which accounts for both the projection and distortions of the fisheye optics.

For the rectification model $\mathbf{T}_\mathrm{V}$, an epipolar-equidistance rectification model was introduced in [Abraham and Förstner, 2005] that allows for epipolar rectification of very large fields of view in both vertical and horizontal directions. This will be introduced in the following.

5.1.4.1 Epipolar-Equidistance Rectification Model

A rectification model for non-perspective omni-directional cameras requires a non-linear function $(u, v)_\mathrm{V} = \mathbf{T}_\mathrm{V}(\boldsymbol{X}_\mathrm{V})$ to be defined, which maps a 3D point represented in virtual camera coordinate system $\boldsymbol{X}_\mathrm{V} = (x, y, z)$ into virtual pixel coordinates $(u, v)_\mathrm{V}$. In order to cover a large field of view, this function has to fulfill some special properties. In particular, it should be such that distances to the principal point are proportional to the angle with respect to the optical axis, rather than to their tangents, as it happens in the pinhole model. The epipolar-equidistance model proposed by [Abraham and Förstner, 2005] considers a function $\mathbf{T}_\mathrm{V}$ such that $(u, v)_\mathrm{V} \sim (\psi, \beta)_\mathrm{V}$, where ψ and β are defined as in Eq. 5.9.

$$\psi = \arctan \frac{x}{\sqrt{y^2+z^2}} \quad \beta = \arctan \frac{y}{z} \tag{5.9}$$

In particular, its inverse transformation is given by Eq. 5.10, where (u', v') represent normalized coordinates in the virtual camera coordinate system.

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \sin u' \\ \cos u' \sin v' \\ \cos u' \cos v' \end{pmatrix} \tag{5.10}$$

Therefore, a rectification model like this causes that the coordinate v depends exclusively on the angle β, while the coordinate u depends on the position of the point on the epipolar plane.

Considering that the camera centers of the virtual cameras are the same as those from the real ones, the relation between virtual camera coordinates and original camera coordinates is given by Eq. 5.11, where the subindex C refers to real camera, and $\mathbf{R}_{\mathrm{C,V}}$ is a 3×3 matrix that represents the relative rotation between virtual a real camera.

$$(u, v)_\mathrm{C} = \mathbf{T}_\mathrm{C}\left(\mathbf{R}_{\mathrm{C,V}} \mathbf{T}_\mathrm{V}^{-1}\left((u, v)_\mathrm{V}\right)\right) \tag{5.11}$$

In the following a new formulation for the inverse rectification model introduced in [Abraham and Förstner, 2005] is given, where the offset of the center of symmetries with respect to the virtual image center is accounted for.

A reference epipolar plane is defined that contains the principal point, as introduced in Eq. 5.7. Over the reference epipolar plane a horizontal field of view of $\psi_V = \left[\psi_V^+ - \psi_V^-\right]$ is covered, which can be computed by means of Eq. 5.8. On the reference epipolar plane the view ray that projects onto each pixel position (u, v) can be computed by means of Eq. 5.12.

$$\hat{\mathbf{d}}_0(u) = Rot([\mathbf{t}_{C_R C_L} \times \hat{\mathbf{e}}_z] \times \hat{\mathbf{e}}_z, (\psi_V^- + u \cdot \Delta\psi_V))\hat{\mathbf{O}}_V \tag{5.12}$$

In this expression, $Rot(\mathbf{e}, \alpha)$ is a 3×3 matrix that defines a rotation around axis $\mathbf{e}$ by an angle α, and $\Delta\psi$ is the angular distance between two consecutive pixels on the same row of the virtual image.

The rest of the epipolar planes can be described as a revolution of the reference one, over the baseline joining both camera centers. According to this, the inverse projection function $\mathbf{T}_{\mathrm{V}}^{-1}(u, v)$ of the virtual cameras can be defined as in Eq. 5.13.

$$\mathbf{T}_{\mathrm{V}}^{-1}(u, v) = Rot(\mathbf{t}_{C_R C_L}, (\beta_V^- + v \cdot \Delta\beta_V))\hat{\mathbf{d}}_0(u) \tag{5.13}$$

This inverse function describes the viewing direction for each pixel coordinate $(u, v)_V$ and maps it onto the unit sphere. The angular step $\Delta\beta$ corresponds to the distance between consecutive epipolar planes and $\beta_V = \left[\beta_V^+ - \beta_V^-\right]$ can be defined based on ψ_V and on the desired aspect ratio.

5.1.5 Change of Pixel Sizes

So far it has been described how the virtual views can be defined so that constraints for epipolar rectification are met. At this point, the next step is to resample the original images at the desired locations, which implies a change in pixel sizes. This effect is pronounced when considering fisheye optics and disaligned optical axes, as depicted in Fig. 5.6 by means of the epipolar lines. The problem of image interpolation has been largely discussed in literature and was introduced in Chapter 3. In the experiments here conducted a Lanczos filter [Turkowski, 1990] with a size parameter $a = 4$ has been applied for image interpolation. No significant change has been observed by applying different size parameters.

After resampling, rastering of the virtual images is possible. In Fig. 5.7 an example of the results after all the described steps is shown. As can be observed, epipolar rectification is achieved for a large field of view, while avoiding the mentioned image artefacts.

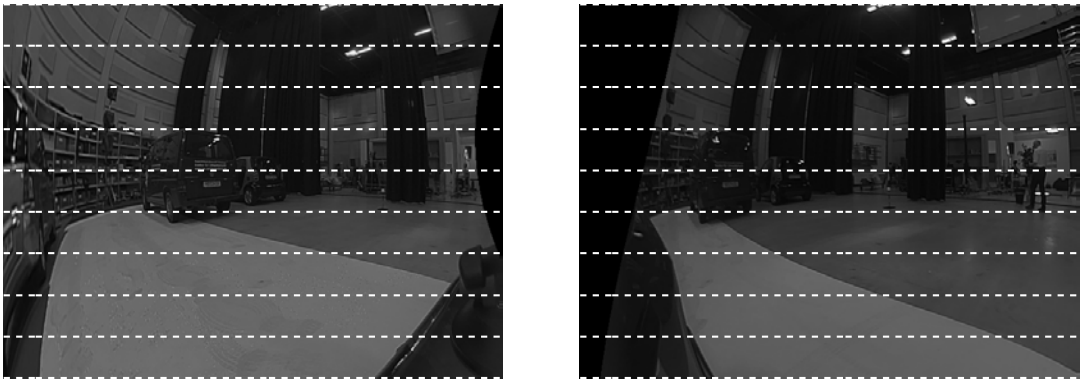


FIGURE 5.7: Result of the rectification process. The image pair corresponds to the original images in Fig. 5.6. It can be observed how now the epipolar planes project into straight lines on the virtual images. Taken from [Esparza et al., 2014b].

5.1.6 Feature-based Disparity Estimation

After conducting the epipolar rectification step, detection and matching of features is to take place. In standard stereo vision setups, the vertical disparity of common features after epipolar rectification is expected to be below one pixel. Nonaccurate calibration may lead to failures in fulfilling this requirement. In the considered surround view setup, due to the large stereo bases, factors like temperature changes, vibrations, etc, introduce a high variance on the relative camera calibration. For this reason, the assumption of a static extrinsic calibration is not strictly valid. In order to deal with this issue, it is proposed to relax the assumption that the vertical offset between different views of a common feature is below one pixel, and accept a larger tolerance on the vertical direction. In this way, vertical offsets larger than a predefined maximum can be discarded. How large this tolerance should be, depends largely on the quality of the camera intrinsic and extrinsic calibration, as well as on the chosen resolution for the epipolar-rectified images. For the experiments in this thesis, the values utilized are described in following sections. As keypoint detector and descriptor those proposed by [Rosten and Drummond, 2005] and [Calonder et al., 2010] are considered, as described in Chapter 3.

After the matching process, triangulation of correspondences is carried out. For rays not perfectly intersecting, the mid-point of the segment covering the shortest distance between both is considered.

An overview of the proposed system is presented in Figure 5.8, where every step can be related to the relevant concepts that were introduced in Chapter 3.

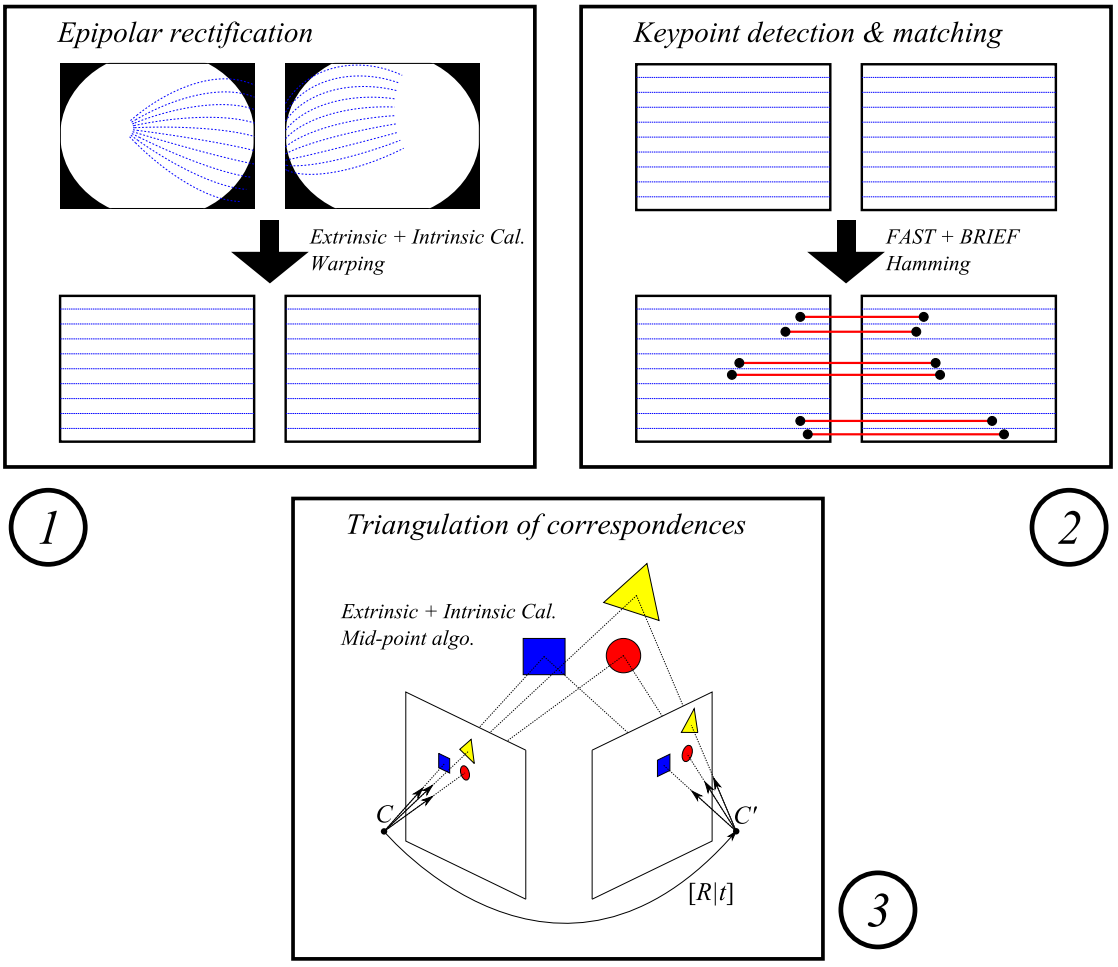


FIGURE 5.8: Overview of proposed system, showing the 3 subsequent processing steps.
Top left: epipolar rectification is conducted based on the intrinsic and extrinsic calibration information.
Top right: Keypoint detection and matching, based on FAST detector and BRIEF descriptor. Matching is performed based on Hamming distance.
Bottom: Triangulation of correspondences. Since the extrinsic and intrinsic parameters of the cameras are known, the correspondences can be triangulated. For nonintersecting rays, the midpoint algorithm is used.

5.1.7 Experimental Setup

For the initial experiments four cameras are considered, which offer a resolution of 1280×960 pixels covering a horizontal field of view of approximately 180 degrees per camera. In Table 5.1, the effective field of view is shown for each adjacent camera pair on the configuration used.

TABLE 5.1: Effective fields of view on current setup, calculated by means of Eqs. 5.6 and 5.8. Data from [Esparza et al., 2014b].

Camera Pair	$[\psi_L^-, \psi_L^+]$ $[deg]$	$[\psi_R^-, \psi_R^+]$ $[deg]$	$[\psi_V^-, \psi_V^+]$ $[deg]$
Front-Right	$[-4.60, 90.00]$	$[-90.00, 4.60]$	$[-60.76, 33.84]$
Right-Rear	$[-11.20, 90.00]$	$[-90.00, 11.20]$	$[-11.56, 67.23]$
Rear-Left	$[-7.03, 90.00]$	$[-90.00, 7.03]$	$[-70.88, 26.15]$
Left-Front	$[-0.43, 90.00]$	$[-90.00, 0.43]$	$[-29.69, 59.88]$

The operating frequency is set to 30 frames per second and the cameras are not synchronized. A common time-stamping system is used by the frame logger that guarantees a maximum temporal jitter of half a frame period. In previous sections it has been discussed why this is acceptable at low speed maneuvering with large stereo bases. The cameras use a CMOS technology with rolling shutter, and frames are compressed previous to storage using JPEG compression. The epipolar rectification is done as described in Section 5.1.4 and an output resolution of 640×480 pixels is used. A search window for corresponding features is set equal to $\pm$ 3 pixels on the vertical direction and 200 pixels on the horizontal direction. In Section 5.1.1 it has been demonstrated that 200 pixels are sufficient, for the considered wide stereo base setup, to detect objects which stand a minimum distance of 6 meters away from the cameras. The implementations for the keypoint detector [Rosten and Drummond, 2005] and descriptor [Calonder et al., 2010] are those available within the OpenCV Library [Bradski, 2000] and matching is performed based on Hamming distance.

5.1.8 Sample Images

In the following, example images obtained before and after conducting the epipolar rectification are shown for different camera pairs. Detected corresponding features are also displayed.

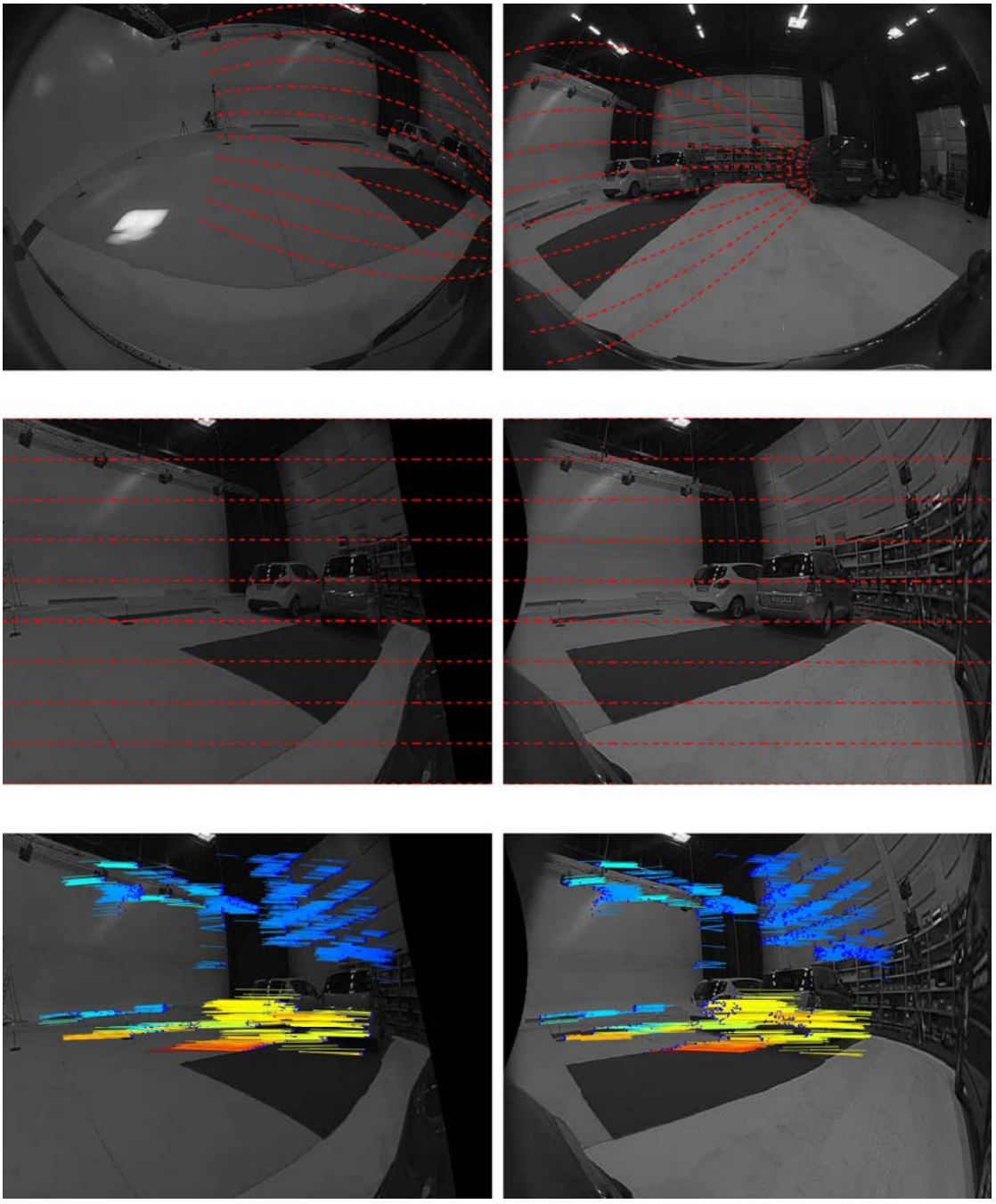


FIGURE 5.9: Result of the rectification and feature matching processes. Upper row: Original images. Middle row: Epipolar rectified images. Lower row: Found correspondences. The image pair corresponds to the Left-Front camera pair

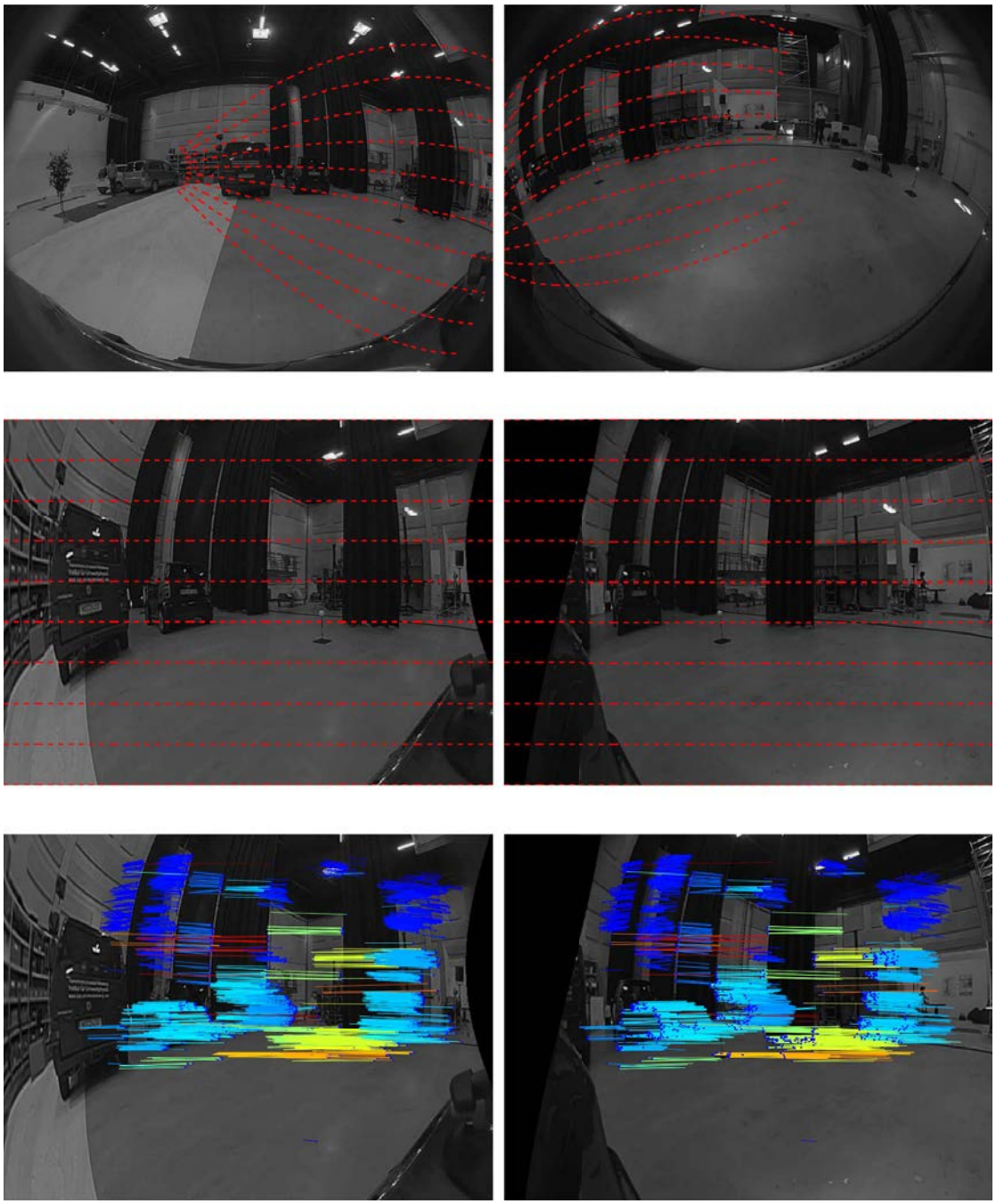


FIGURE 5.10: Result of the rectification and feature matching processes. Upper row: Original images. Middle row: Epipolar rectified images. Lower row: Found correspondences. The image pair corresponds to the Front-Right camera pair

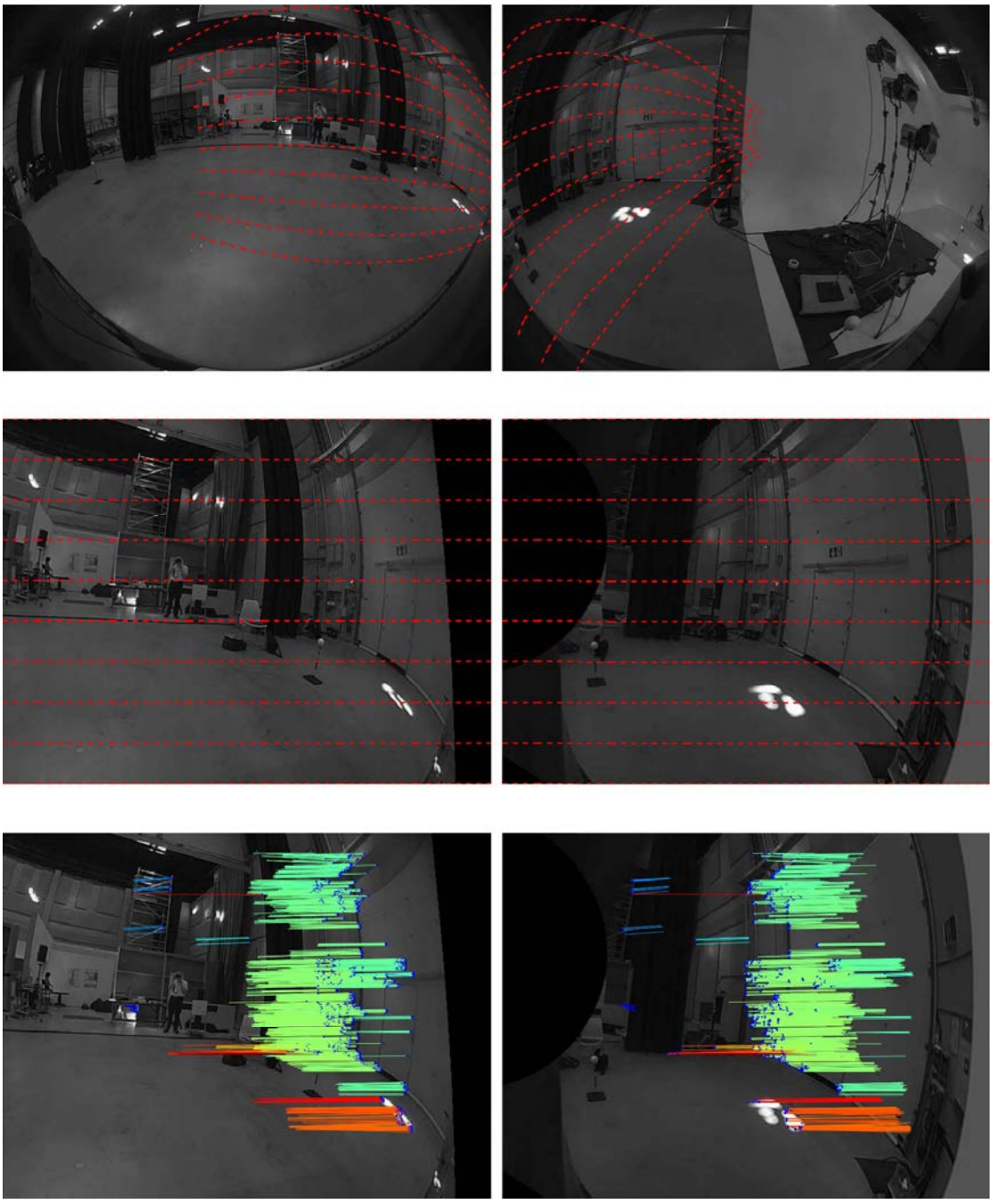


FIGURE 5.11: Result of the rectification and feature matching processes. Upper row: Original images. Middle row: Epipolar rectified images. Lower row: Found correspondences. The image pair corresponds to the Right-Rear camera pair

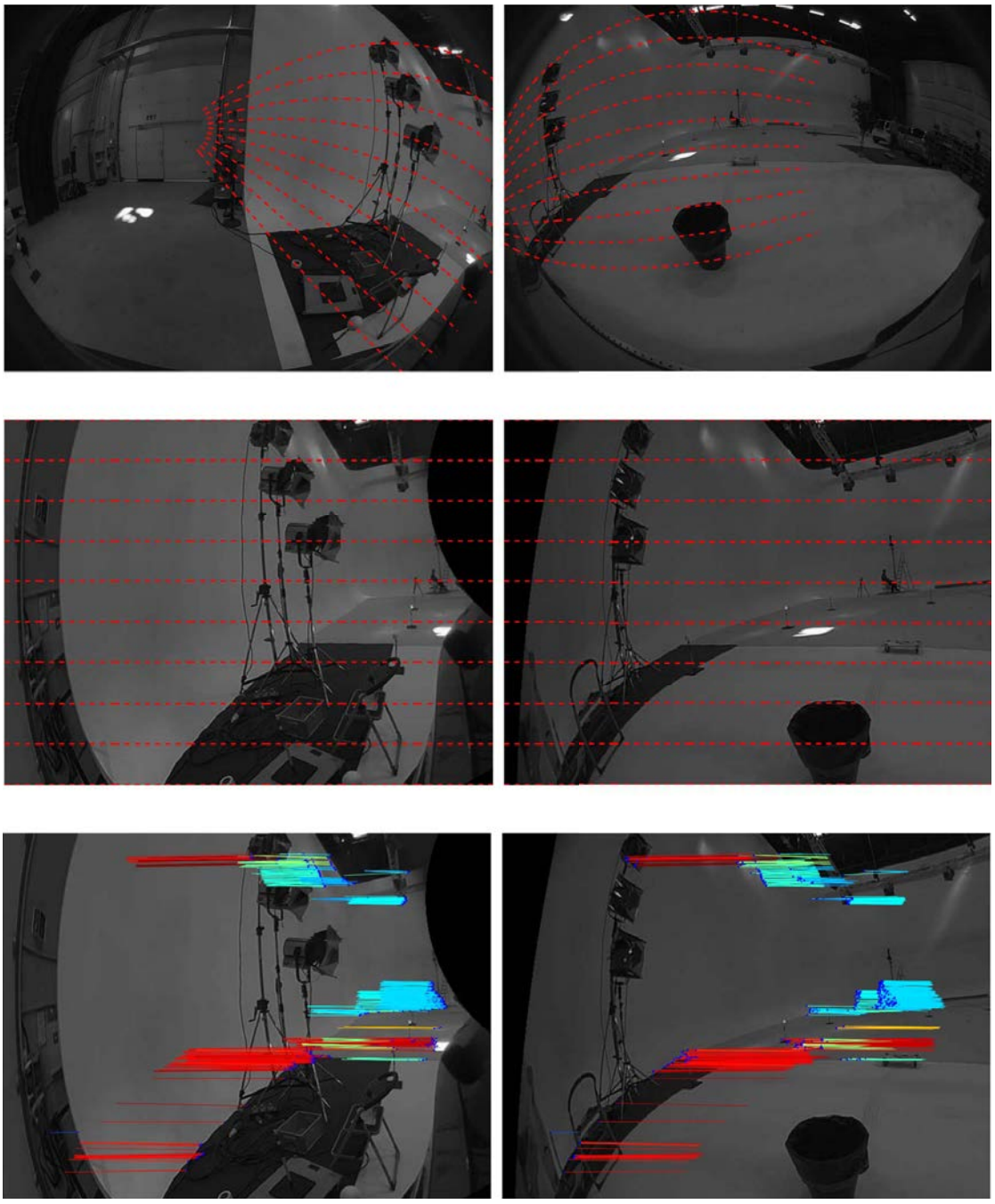


FIGURE 5.12: Result of the rectification and feature matching processes. Upper row: Original images. Middle row: Epipolar rectified images. Lower row: Found correspondences. The image pair corresponds to the Rear-Left camera pair

5.1.9 Evaluation

For evaluating this approach, 10 static sequences were considered. Results correspond to single frames, without time accumulation. The error is evaluated by reprojecting all measurements to the front and rear cameras only and results are shown in Table 5.2.

TABLE 5.2: Absolute and relative error analysis, as presented in Eq. 4.7. Quartile information is included since it is representative for discussion of results. Data from [Esparza et al., 2014b].

Seq. ID	Measurements	Mean [m]	σ [m]	Q50 [m]	Q75 [m]	Q50 [%]	Q75 [%]
1	5670	1.14	1.69	0.49	1.38	5.16	11.43
2	4280	0.97	1.76	0.38	1.04	4.17	9.23
3	1515	0.98	1.61	0.34	0.96	4.00	10.01
4	5115	1.31	1.85	0.52	1.84	6.51	17.51
5	6205	1.48	2.10	0.58	1.75	7.17	17.35
6	11084	1.10	1.69	0.42	1.31	4.96	12.07
7	4542	0.88	3.04	0.19	0.77	4.43	12.53
8	6487	2.54	6.58	0.25	1.28	4.27	16.54
9	6238	1.93	6.69	0.36	1.32	6.09	17.92
10	22531	0.62	1.36	0.21	0.52	3.34	7.68

The average error achieved is approximately between ± 1 and ± 2 meters on a range which covers distances of up to 20 meters, which is reasonable for park & maneuver systems. In most sequences, quartile information shows relative errors lower than 6% and 20% for 50% and 75% of all measurements, respectively. Although all sequences were recorded on similar conditions, larger level of error is observed in some of them. A deeper look into the data shows a high level of confusion on the feature matching process, due to certain repetitive patterns. This effect has been highlighted on Fig. 5.13.

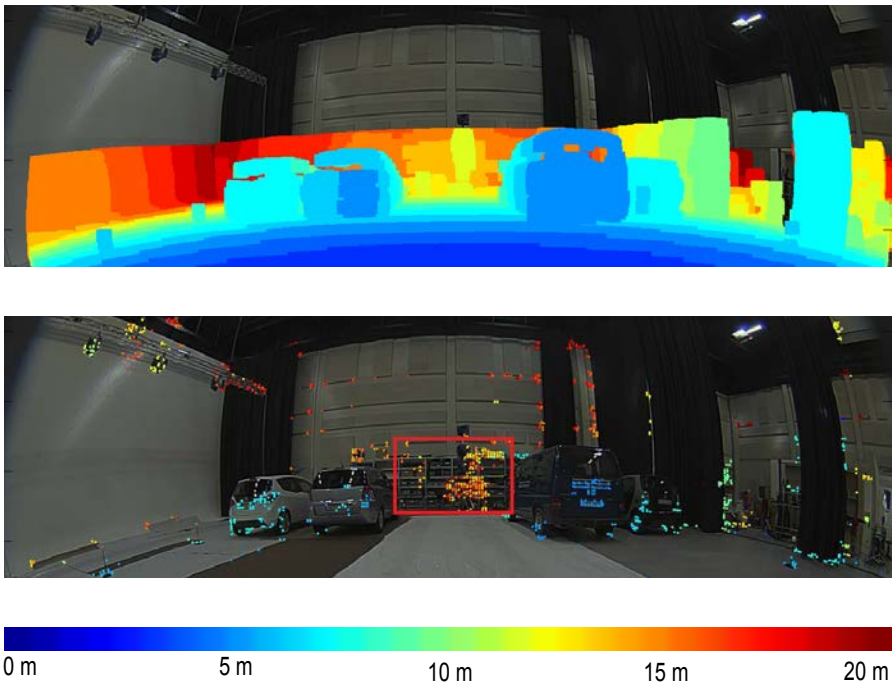


FIGURE 5.13: Comparison of estimated depth values with reference. Above: Reference measurements backprojected onto front image. Below: Measurements of the camera pairs left-front and front-right backprojected onto front image. Color encodes distances to camera center, and is on the same scale for both images. Red: High confusion area due to repetitive pattern. Taken from [Esparza et al., 2014b].

In the present system configuration, the considerable overlap of the field of view of any pair of adjacent cameras is limited to approximately 90 degrees. Therefore only in these regions 3D information could be recovered. Furthermore, objects in the very close vicinity of the ego vehicle show a very large disparity on the rectified images. The feature search was restricted to 200 pixels on the horizontal direction, which allows detection of objects which are, at least, in the order of 6 meters away from the vehicle. In the areas where 3D information could be recovered, the distances compare well with the distances obtained by the lidar, as can been seen in Figue 5.13. No time accumulation is required, thus being the 3D information recovered from single pairs of images. Furthermore, considering that these results are based on a general purpose feature detector and descriptor, performance is expected to benefit substantially from denser state-of-the-art disparity estimators which would allow for more 3D measurements and a reduced count of outliers.

There is a very wide field of potential applications for the proposed system within driving assistance, especially in low speed parking and maneuvering. In the next sections, focus will be put into the optimization of the current camera mounting as well as on performing measurements on a nearer area around the vehicle.

5.1.10 Conclusions

The experiments carried out in this section show that 3D spatial reconstruction is feasible based on stereo measurements with automotive surround view cameras. The benefits of such a system have been discussed in the context of field of view coverage and accuracy of the measurements. In the areas where 3D information could be recovered, the distances compare well with the distances obtained by the lidar, as can been seen in Figure 5.13. No time accumulation is required, thus being the 3D information recovered from single pairs of images.

In the present system configuration, the considerable overlap of the field of view of any pair of adjacent cameras is limited to approximately 90 degrees. Therefore only in these regions 3D information could be recovered. Furthermore, objects in the very close vicinity of the ego vehicle show a very large disparity on the rectified images. The feature search was restricted to 200 pixels on the horizontal direction, which allows detection of objects which are, at least, in the order of 6 meters away from the vehicle.

As discussed in Chapter 4, limitations exist by using a lidar for benchmarking since the fields of view of both sensors are not completely coincident. In particular, objects too near to the ego vehicle, or too high, remain outside the visibility range of the lidar. The latter is, however, not crucial for these use cases, since for driving assistance such heights usually lack interest. There is a very wide field of potential applications for the proposed system within driving assistance, especially in low speed parking and maneuvering.

The focus of this work was to demonstrate feasibility of the approach using standard open source disparity estimators, although proprietary algorithms with superior performance exist. The next step presented in this thesis will focus on optimization of the current camera mounting as well as on performing measurements on a nearer area around the vehicle.

5.2 Towards Surround 3D Measurements

In the previous section it has been shown that stereo reconstruction is possible under the described four-camera surround view setup. Nevertheless, it has also been shown that the areas around the vehicle where this is possible are largely restricted. This section presents an extension of the current standard surround view system where eight fisheye cameras are used. It is based on the work of [Esparza et al., 2014a], which was done within the framework of this thesis.

In the field of park and maneuver assistance, different configurations have been proposed for surround-view systems. The most common criteria used to define the mounting of cameras is the amount of space in the vicinity of the vehicle that is imaged by at least one camera. In the work of [Ehlgen and Pajdla, 2007] the position of catadioptric cameras mounted on a truck was optimized in order to see every point on the ground plane in the truck surrounding by at least one camera.

Surround view configurations with more than four cameras have been previously proposed in literature. In [Liu et al., 2008] a system with six cameras that individually cover small fields of view is proposed. This system, however, does not consider 3D measurements. This section proposes an extension to the conventional surround view configuration with additional cameras in order to increase the area around the ego vehicle where 3D measurements are possible. In particular 8 fisheye cameras are considered with a horizontal field of view of approximately 180 degrees. In addition to the standard four positions four cameras were added on the four corners of the vehicle so that the angular distance between the optical axes of adjacent cameras is reduced from 90 to 45 degrees approximately. Distances between camera centers were also reduced as a result. This involves that 3D measurements are possible on closer distances to the ego vehicle.

Areas lying within the possible driving path of the ego vehicle were prioritized, thus seeking better overlaps on the front and rear ends of the vehicle. Such a configuration allows for a surround stereo system, where 3D measurements can be effectively conducted on every direction, except for the very near vicinity of the ego vehicle, where other sensors like ultrasound perform well on measuring distances to objects.

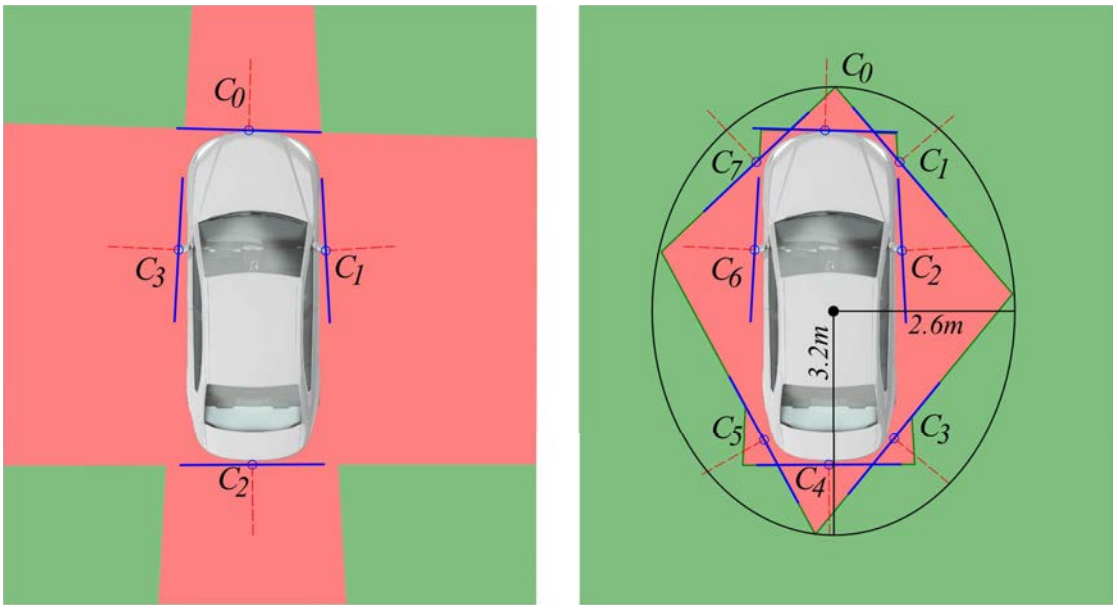


FIGURE 5.14: Overview on the stereo pair overlaps. Areas where stereo measurements are possible are displayed in green, and those where not, in red. Left: Traditional configuration. Right: Proposed configuration. For the new configuration, a complete coverage for stereo estimation is possible outside the area limited by an ellipse of half axes equal to 2.6m and 3.2m. Taken from [Esparza et al., 2014a].

In Figure 5.14, a schematic representation of the existing fields of view is depicted, both for a traditional surround view configuration and for the proposed extended setup. Real calibration data from the setup has been considered to generate this view. As can be seen, a close-to-full 360 degree surround stereo system can be achieved considering a minimum distance from the vehicle.

5.2.1 Experiments

A setup with 8 cameras has been considered for these experiments, which offer a resolution of 1280x960 pixels covering a horizontal field of view of approximately 180 degrees per camera. The cameras used are of the kind described in Section 5.1 and a similar operating mode has been considered.

The positions were manually adjusted to fit the exterior design of the vehicle and to lay on salient areas where their fitting would be feasible in practice. The cameras were intrinsically calibrated with the method described in Section 3.2.1.2. Extrinsic calibration for the cameras was also done, by means of special calibration targets as well as additional cameras and bundle adjustment, as described in Section 3.1.4.

Epipolar rectification of the fisheye images for the stereo matching process is done according to the work presented in Section 5.1, and an output resolution of 640 × 480 pixels was used.

In order to evaluate the amount of shared field of view in the present camera configuration, different experiments were carried out. On a first step, the 2D model proposed in Section 5.1 was considered to create an initial estimate of the overlapping areas, based on camera calibration. From a more empirical perspective, a pedestrian walking around the ego vehicle was considered. The pedestrian completed a whole loop in approximately 60 seconds, keeping a distance to the vehicle of approximately 2 meters. This experiment is aimed at evaluating in which areas the system is blind to the pedestrian, and whether this fits to the initial expected results. As a reference, data from a lidar scanner was considered, which was mounted on the roof of the vehicle, and registered to the multicamera system as described in Chapter 4. While the vehicle was static, the 3D measurements were accumulated over time in order to see the path that the pedestrian follows based on the reference lidar sensor and to compare it with the one estimated by the proposed system.

5.2.2 Results

Table 5.3 presents an analysis of the overlapping fields of view of each adjacent camera pair on the presented configuration, according to the 2D model presented in Section 5.1.3. The existing overlaps are between 118 and 148 degrees. For the new configuration, it is estimated that a complete coverage for stereo estimation is possible outside the area limited by an ellipse of half axes equal to 2.6m and 3.2m. Theoretically, this means than any object not contained by this ellipse is observed by more than one camera and therefore its position could be estimated. In practice, there still exist large parallaxes between adjacent cameras on very near ranges that may make the process difficult.

TABLE 5.3: Analysis of the stereo base and overlapping fields of view for each adjacent camera pair. Taken from [Esparza et al., 2014a].

Camera Pair	C_0-C_1	C_1-C_2	C_2-C_3	C_3-C_4
Distance between camera centers $\{m\}$	1.12	1.22	2.61	0.97
Available field of view $\{deg\}$	131.99	143.12	136.96	130.00

Camera Pair	C_4-C_5	C_5-C_6	C_6-C_7	C_7-C_0
Distance between camera centers $\{m\}$	0.96	2.64	1.21	1.04
Available field of view $\{deg\}$	118.61	148.17	136.08	135.08

In Figure 5.15 results of the feature matching process after epipolar rectification are shown for four of the camera pairs in the system.

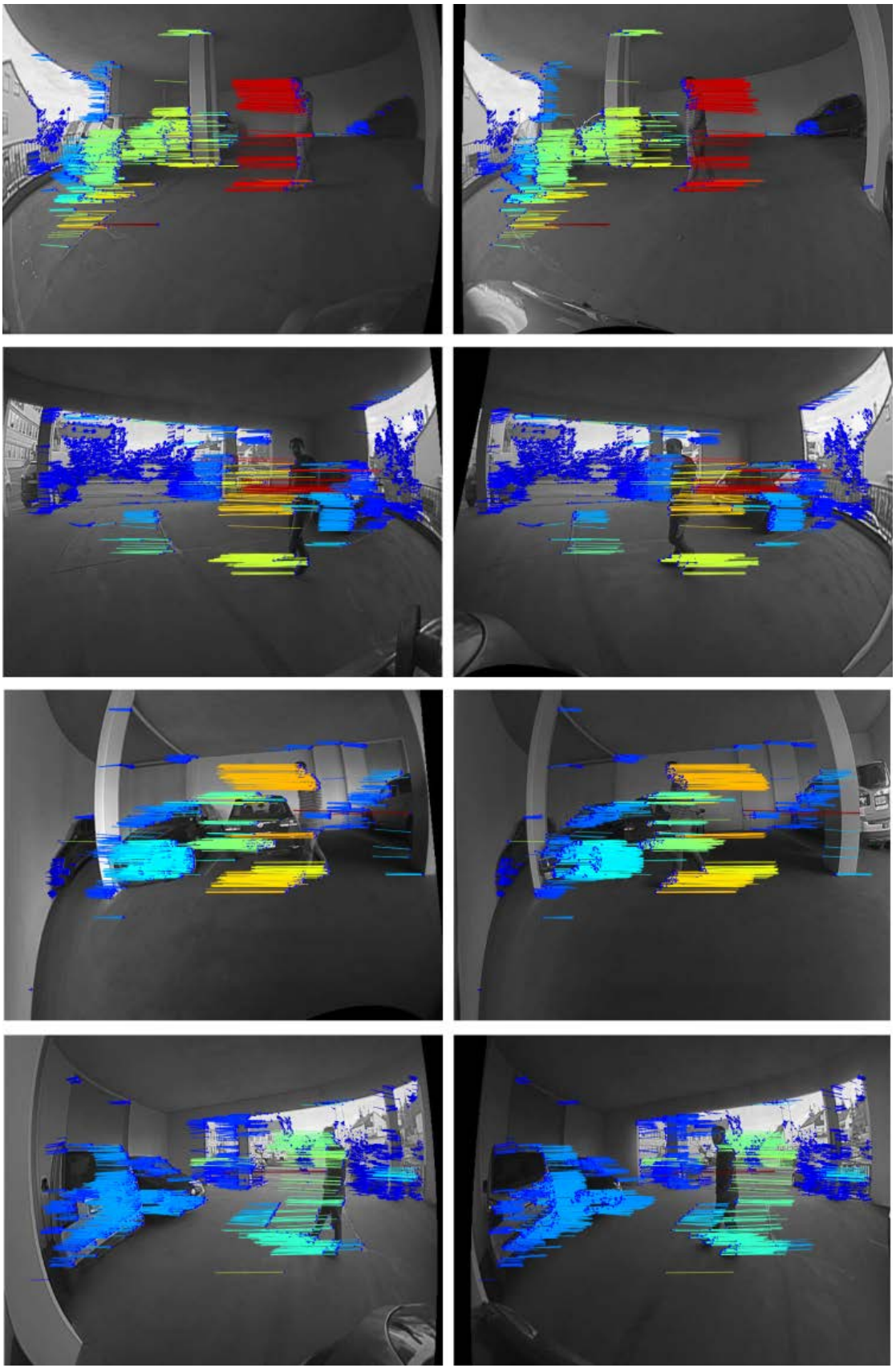


FIGURE 5.15: Result of the feature matching for four different camera-pairs. Color encodes normalized disparity, based on JET scale. Taken from [Esparza et al., 2014a].

The timely accumulated path followed by a pedestrian walking around the ego vehicle is shown in Figure 5.16. Approximately 75% of the path could be measured by the system. Regarding the estimated dephts, the data obtained by means of stereo measurements compares well to the data from the reference lidar sensor.

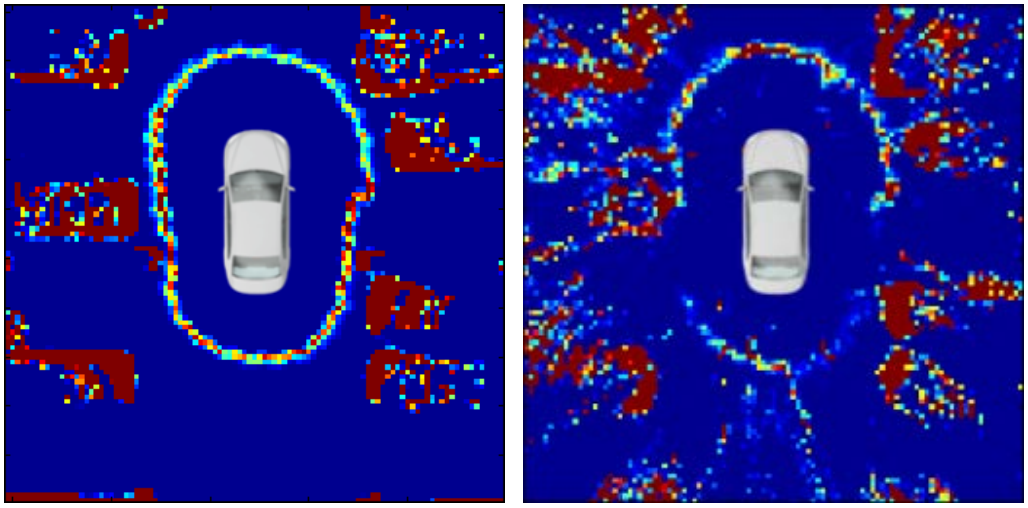


FIGURE 5.16: Comparison of path followed by pedestrian walking around the vehicle. Color encodes the number of detections per area on the tessellated surface and is saturated at the count of 500 detections. Left: Lidar. Right: Stereo Video. Taken from [Esparza et al., 2014a].

Figure 5.17 shows two additional static scenes where lidar measurements and stereo measurements are overlaid, without time accumulation. It can be observed that distances match well to the reference and measurements are possible on every direction around the ego vehicle.

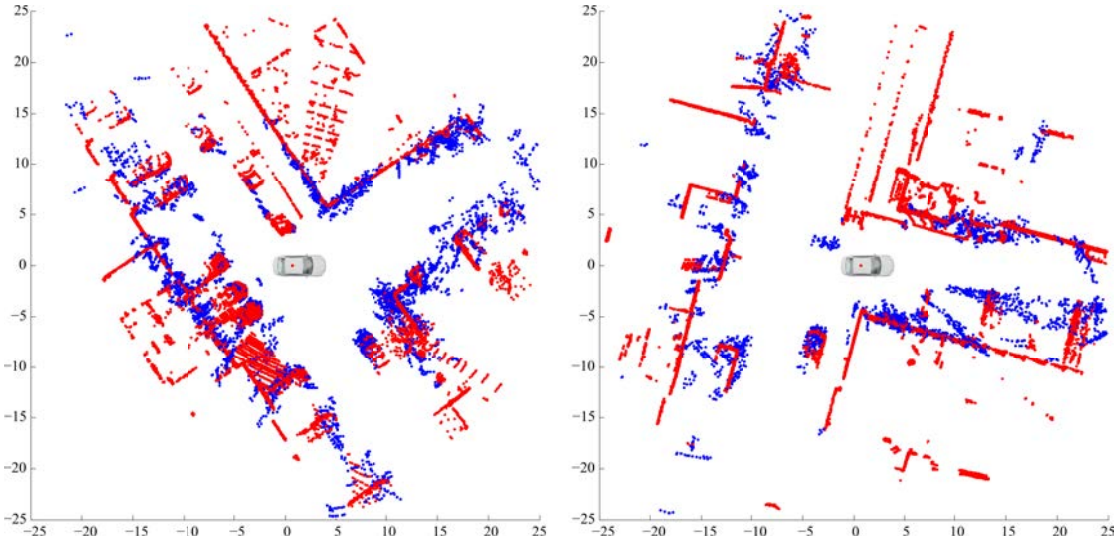


FIGURE 5.17: Comparison of obtained distance information for two different sequences. Blue: Surround stereo measurements. Red: Lidar measurements. All measurements are from a single frame, without time accumulation. Distance indications are given in meters for both x- and y-axes. Taken from [Esparza et al., 2014a].

5.2.3 Comparison with Four-Camera Setup

In order for the results obtained with the eight-camera configuration to be compared to the standard four-camera setup, the same scenes from Figure 5.17 have been processed without taking into consideration the extra cameras. The results can be seen in Figures 5.18 and 5.19.

It can be observed how the density of measurements obtained is noticeable larger on the setup with eight cameras. Furthermore, the coverage of the field of view around the vehicle is largely improved with shorter distances available.

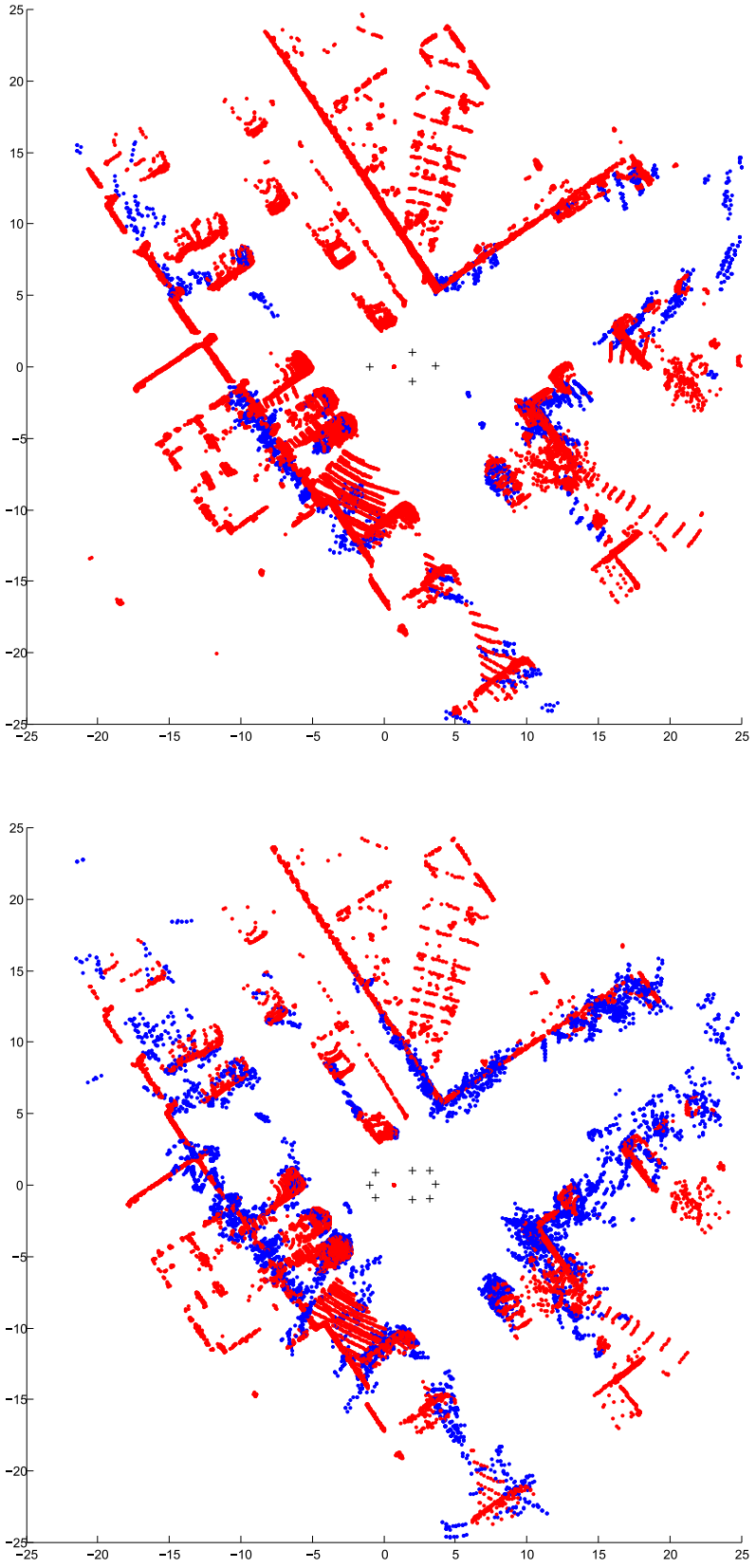


FIGURE 5.18: Comparison of reconstruction results with four and eight camera setup. Top: four camera setup on standard configuration. Bottom: eight camera setup as discussed in this section. Red: reference lidar measurements. Blue: measurements by means of multi-camera stereo. Black crosses: camera positions. Taken from [Esparza et al., 2014a].

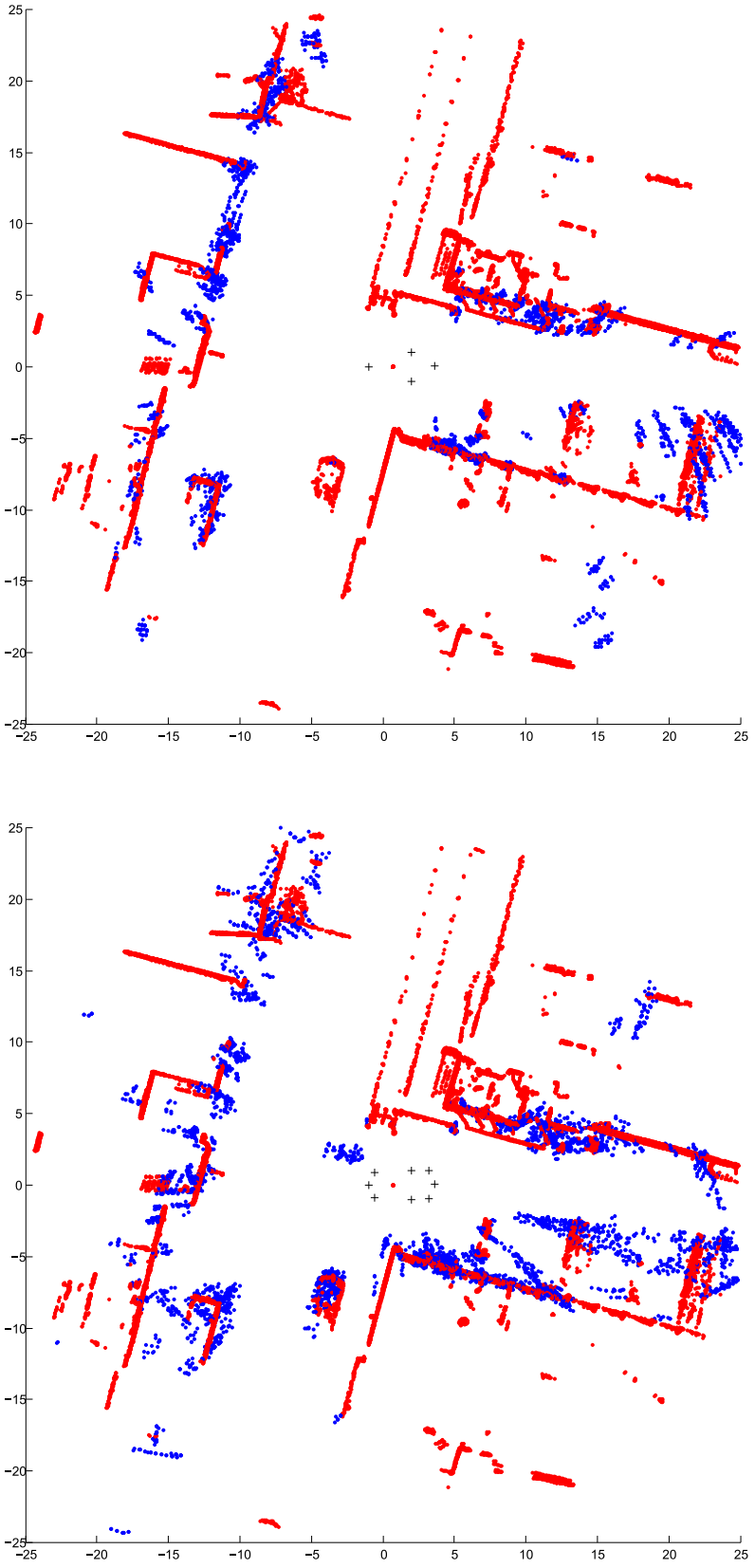


FIGURE 5.19: Comparison of reconstruction results with four and eight camera setup. Top: four camera setup on standard configuration. Bottom: eight camera setup as discussed in this section. Red: reference lidar measurements. Blue: measurements by means of multi-camera stereo. Black crosses: camera positions. Taken from [Esparza et al., 2014a].

5.2.4 Conclusions

In the presented system configuration, the overlap of the fields of view of adjacent cameras has been increased by means of additional cameras from 90 degrees aproximately on previously existing configurations to over 130 degrees on the current one.

Of the 360 degrees followed by the pedestrian on the vicinity of the vehicle, approximately 75% can be covered under the proposed configuration. The distance at which this was evaluated is approximately 2 meters from the vehicle. Nearer objects are in general difficult to consider due to very large parallaxes. However, other sensors typical of Park & Maneuver systems, like ultrasonic sensors, perform well on very short distances, being both systems a good combination for maneuvering assistance in the very rear and near ranges.

In the next section focus will be put on performing 3D measurements on narrow drive ways and parking spots, for which a new camera setup and epipolar rectification model are proposed.

5.3 Mapping of Narrow Driveways and Parking Spaces

This section discusses stereo reconstruction in narrow drive paths and parking slots. For this purpose a new camera configuration is proposed and a new polar epipolar rectification model is introduced.

Using the 4-camera configuration described in Section 2.1.1, recent research has focused in the detection of parking spots [Unger et al., 2014] and automatic parking [Furgale et al., 2013], which involves the mapping of narrow drive ways or parking spaces, prior to the actual vehicle maneuvering. For these tasks, mono-camera techniques like structure from motion are usually considered [Unger et al., 2014].

Although great progress has been done in these fields, certain problems still remain which are not anymore algorithmic, but rather show the physical limitations of the considered setups. In the case of structure from motion, for example, the prerequisite of camera movement in order to achieve 3D measurements can certainly be an inconvenient in parking situations. As for stereo techniques with traditional surround view configurations, in Section 5.1 it has been shown that the areas around the vehicle where 3D measurements can be conducted are highly restricted. Furthermore, narrow driving spaces mean that the surface resolution for side cameras is extremely low and large parallaxes exist across camera views, making this approach nonreliable in practice. As a result of the large parallax, light reflections on the surface of objects can also play an important role in the disparity estimation process, usually increasing the rate of false positives. In Figure 5.20 an example situation is shown, where these problems are depicted.

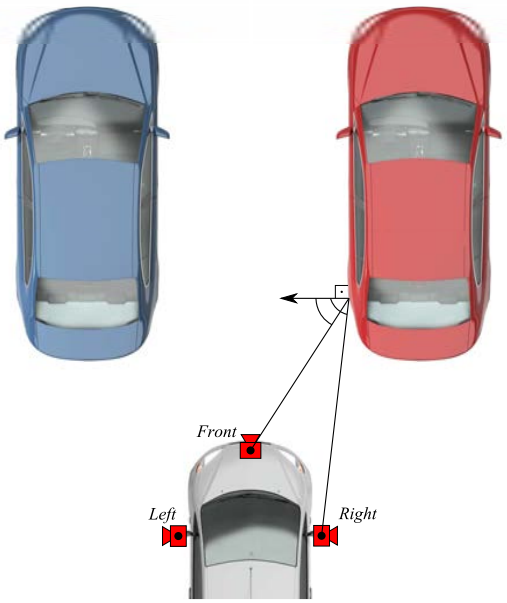


FIGURE 5.20: Scene description: conventional surround view setup for narrow drive ways. There is a large parallax and the surface resolution for the side camera is very low. Performing stereo measurements in the current setup is therefore limited by physical restrictions, rather than by the wellness of the algorithms. Taken from [Esparza et al.,].

In general the mapping of narrow drive ways and parking spaces previous to the vehicle automatic maneuvering remains an open problem. As an example the park-in situation depicted in Figure 5.20 can be considered: the important surfaces possess normal vectors almost perpendicular to the driving direction and are therefore hard to observe in front of the car. Further example images are given in Figures 5.21 and 5.22 from the real camera setup.

FIGURE 5.21: Example of traditional configuration prior to entering a narrow drive path or park space. The images correspond to the Front-Right camera pair, and are epipolar-rectified. The large parallax can be observed on the green vehicle. In particular, the side of the car offers a very different surface resolution to each of the cameras, which reduces the chances of correct keypoint matching.

FIGURE 5.22: Example of traditional configuration inside a narrow drive path or park space. The images correspond to the Front-Right camera pair, and are epipolar-rectified. The lateral side of the white vehicle is observed from very different perspectives by both cameras. This complicates the search of correspondences.

In this section a new camera configuration is proposed that can mitigate the problem of the large parallaxes and low surface resolution. This configuration involves a relative camera pose such that measurements near the line joining both camera centers have to be carried out, namely near the epipoles. A new model to achieve epipolar rectification on the proximities of the epipoles is presented. This model transforms every epipolar plane not into an image raster line, but into a polar direction. It is shown, by means of experimental results, that the proposed model allows for 3D reconstruction in the vicinity of the epipoles without involving large changes of pixel size and measurement of the parking spots can be conducted prior to the vehicle's maneuvering.

5.3.1 Proposed Camera Setup

In this section a new camera mounting configuration is proposed where special focus has been put into the pre-measurement of front-parking spots. As discussed previous sections, stereo vision based on surround view cameras has certain restrictions due to different surface resolutions and large parallax which are not easily solved algorithmically. In particular, the amount of imaged surface area per pixel depends largely on the angular distance between the surface normal and the view ray from the camera center, i.e. the larger the angle, the larger the surface that is covered per pixel. Considering a front-side camera pair as depicted in Figure 5.20, the difference in surface resolutions between both cameras is very large, thus the projections onto both image planes differ strongly. Furthermore, the very different perspective from both cameras makes the illumination conditions especially

relevant. Different light reflections on each camera view usually add confusion problems on the feature matching process.

In order to enhance the similarities across different views, it is proposed to mount two surround view cameras approximately at the top of the windshield. A possible mounting position for the cameras would be directly on the roof the vehicle. Alternative positions can be considered without loss of generality on the algorithmical description presented on the next sections. By considering such a setup, the differences are reduced between the projection of the lateral surfaces onto both cameras, thus the chances for success on the feature matching process are higher. In particular, the proposed configuration is depicted in Figure 5.23.

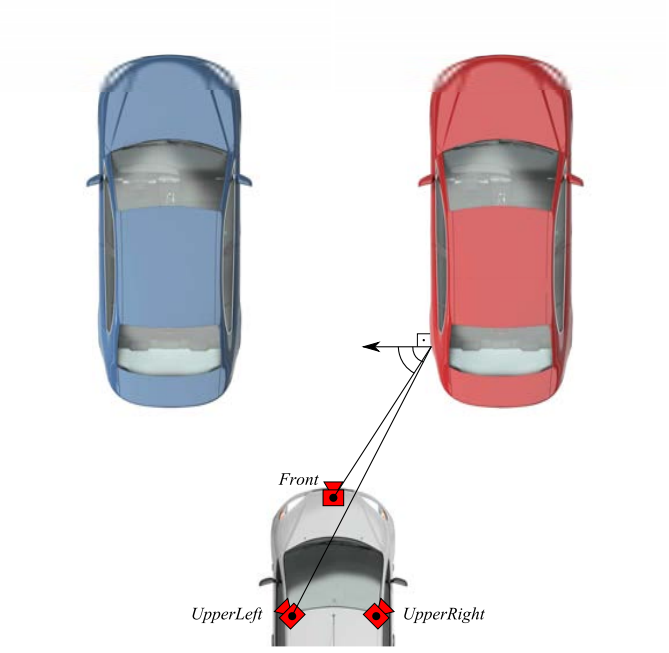


FIGURE 5.23: Proposed new setup: additional cameras are considered, mounted on top of the windshield. The parallax is largely reduced and both cameras have a similar surface resolution. Taken from [Esparza et al.,].

The described configuration does, however, have a drawback if the relative position of both cameras is considered. In particular, the area where 3D measurements are to be conducted corresponds to the vicinity of the line joining both camera centers, i.e. near the epipoles. This situation resembles that of a pure forward motion on a mono-camera configuration or axial stereo [Alvertos et al., 1989], [Nguyen and Huang, 1992], [Dalmia and Trivedi, 1995].

Since lateral stereo implies a much more common camera setup than axial stereo, existing epipolar rectification models for fisheye optics (like the one proposed by [Abraham and Förstner, 2005]) do not account for such a setup. In the next

sections, a polar-epipolar rectification model is proposed that allows for epipolar rectification in the vicinity of the epipoles without involving large changes of pixel size, and is valid for large fields of view in both horizontal and vertical directions.

5.3.2 Coincident Optical Axes

Given the camera mounting proposed in Section 5.3.1, the relevant scene content is mostly perceived near the epipoles, as depicted in Figure 5.24. In general, epipolar rectification models are defined so that epipolar lines are mapped to parallel raster lines. As a result, epipoles and their vicinities are infinitely expanded thus generating large changes in pixel size. This can be explained since the epipoles are contained by all epipolar lines. To overcome this issue, the proposed model is aimed at creating radial epipolar lines with the origin located at the principal point, and common for both virtual cameras. In this way, both virtual optical axes are coincident.

FIGURE 5.24: Epipolar lines on original fisheye images, for the camera configuration depicted in Figure 5.23. Top row: Front-UpperLeft camera pair. Bottom row: Front-UpperRight camera pair. Taken from [Esparza et al.,].

In order to achieve this, the rotation matrix $\mathbf{R}_{C,V}$ in Eq. 5.11 (presented in Section 5.1.4.1) is split into two different matrices $\mathbf{R}_{SF,V}$ and $\mathbf{R}_{C,SF}$, where the first part of the index, SF, refers to the intermediate sensor frame reference coordinate system, which has an orientation common to all cameras on the system, as defined in Section 3.1.1. Equation 5.11 becomes therefore Eq. 5.14.

$$(u,v)_C = \mathbf{T}_C \left[\mathbf{R}_{C,SF}\mathbf{R}_{V,SF}^{-1}\mathbf{T}_V^{-1}\left[(u,v)_V\right]\right] \tag{5.14}$$

To guarantee that both virtual images share epipolar lines on the radial directions, it is sufficient to define a common $\mathbf{R}_{V,SF}$ for both rectifying cameras on a camera pair such that the optical axes are aligned and pass through both camera centers.

The optixal axis $\hat{\mathbf{O}}_V$ of both virtual cameras is defined as the normalization of the vector $\mathbf{b}$ joining both camera centers C_1 and C_2, as in Eq. 5.15.

$$\mathbf{b} = C_2 - C_1 \tag{5.15}$$

Based on the x-, y-, z-components of $\mathbf{b}$ (b_x, b_y, b_z), two of the three parameters defining the camera orientation can be extracted by means of Eqs. 5.16, 5.17.

$$\alpha_V = \arctan\frac{b_x}{b_z} \tag{5.16}$$

$$\gamma_V = \arctan\frac{b_y}{\sqrt{{b_x}^2 + {b_z}^2}} \tag{5.17}$$

The third parameter missing to fully describe $\mathbf{R}_{V,SF}$ accounts for the roll angle, which represents the rotation over the optical axis. This parameter can be set freely, but must be the same for both virtual cameras. The author recommends fixing it such that the x-axis of the camera is parallel to the reference floor plane considered for the extrinsic calibration, so that the up-vector of the camera is as parallel as possible to the floor's normal. The centers of the virtual cameras remain the same as those of the real cameras. In Figure 5.25 a graphic description of the previous parameters is provided.

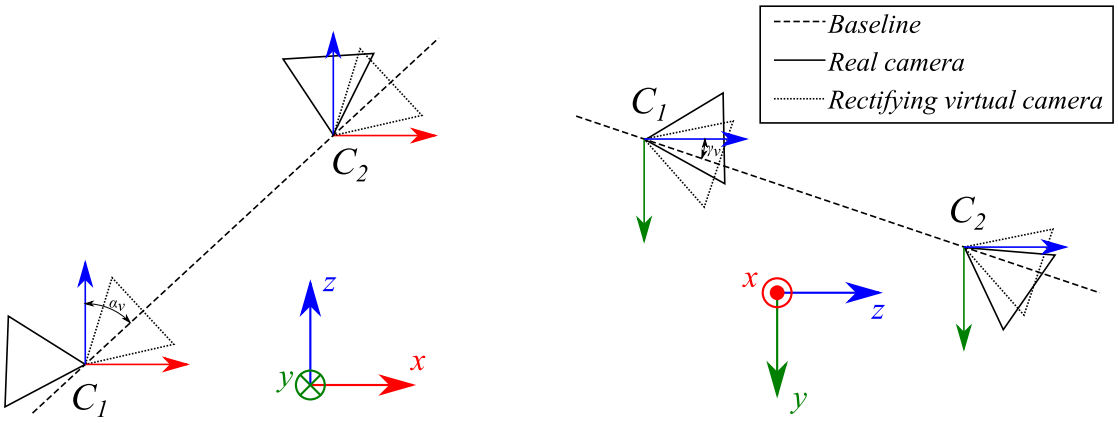


FIGURE 5.25: Coincident optical axes for virtual rectifying cameras. The orientation of the virtual cameras is defined based on Eqs. 5.16 and 5.17. The *roll* angle can be freely set, common to C_1 and C_2. Projection centers of virtual and real cameras remain the same.

Ideal Projection Model

Once the orientation of the virtual rectifying cameras has been defined, different ideal nonlinear models can be considered for $\mathbf{T}_V$ that project the world onto the virtual image plane. Several models have been proposed in literature that allow for an ideal projection of large fields of view (see Section 3.2.1.2.) For these experiments, the equidistant model has been considered, which inverse model is given by Eqs. 5.18, 5.19 and 5.20.

$$X = \frac{u'}{\sqrt{u'^2 + v'^2}} \sin \sqrt{u'^2 + v'^2} \tag{5.18}$$

$$Y = \frac{v'}{\sqrt{u'^2 + v'^2}} \sin \sqrt{u'^2 + v'^2} \tag{5.19}$$

$$Z = \cos \sqrt{u'^2 + v'^2} \tag{5.20}$$

Expressions 5.18, 5.19, and 5.20 are given in CV-coordinates, where (u', v') represent image coordinates normalized with respect to the principal point.

Zooming Correction

The equations presented so far can be directly applied to both cameras of any axial stereo pair. However, experience shows that the apparent forward motion between

the cameras has a zooming effect that has to be accounted for. In conventional lateral stereo vision setups, the rectified optical axes are approximately orthogonal to the stereo baseline, thus the scaling difference across views is neglectable. In this setup, however, the difference in scale of the imaged objects is relevant. In particular, this is especially true for distances at a range comparable to the stereo base between both camera centers. In Figure 5.26 this effect is depicted.

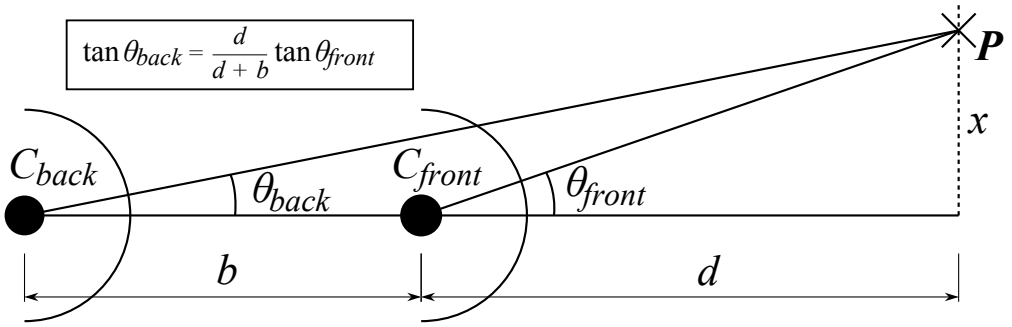


FIGURE 5.26: Description of the zooming effect. The image from the front camera can be seen as a zoomed version of the image from the rear camera. This effect can not be neglected, since $d \approx b$ in the vicinity of the ego vehicle. Taken from [Esparza et al.,].

Although scale-invariant keypoint detectors and descriptors exist in literature [Lowe, 2004], they are computationally expensive. To compensate for the scale differences, it is proposed to use different projection models for the epipolar rectification of each of the cameras that directly account for a correction factor.

It is possible to differentiate between front and back cameras, ie. the camera which is closer and further from the scene, respectively, as illustrated in Figure 5.26. For the back camera, a modified version of the equidistant model presented in the previous section is introduced. A term α is introduced that accounts for the zooming factor, thus transforming Eq. 5.20 into Eq. 5.21.

$$Z_{back} = \alpha \cos\sqrt{u'^2 + v'^2} \tag{5.21}$$

The term α can be obtained by means of Eq. 5.22, where d represents the expected distance to the scene. Different values can be considered, depending on the expected depth to objects. Since the aim of this work is to measure narrow drive ways and parking spaces, short distances are the most relevant. Therefore, values of α approximately equal to 2 are recommended ($d \approx \|\mathbf{b}\|$).

$$\alpha = \frac{d + \|\mathbf{b}\|}{d} \tag{5.22}$$

The projection $\hat{\mathbf{u}}$ onto the unit sphere is given by the normalization of the (X, Y, Z_{back}) vector, as in Eq. 5.23.

$$\hat{\mathbf{u}}_{back} = \frac{(X, Y, Z_{back})^T}{\|(X, Y, Z_{back})\|} \tag{5.23}$$

Once the rectifying model is defined, the original images can be sampled at the coordinates obtained by means of Eq. 5.14. Since a rectifying model has been used that does not introduce large pixel size changes, normal bilinear interpolation can be considered (see Section 3.2.2.2).

Figure 5.27 shows the results of the rectification based on the proposed model. Visual inspection of the images shows that common objects visible on both views lie on the same polar epipolar lines.

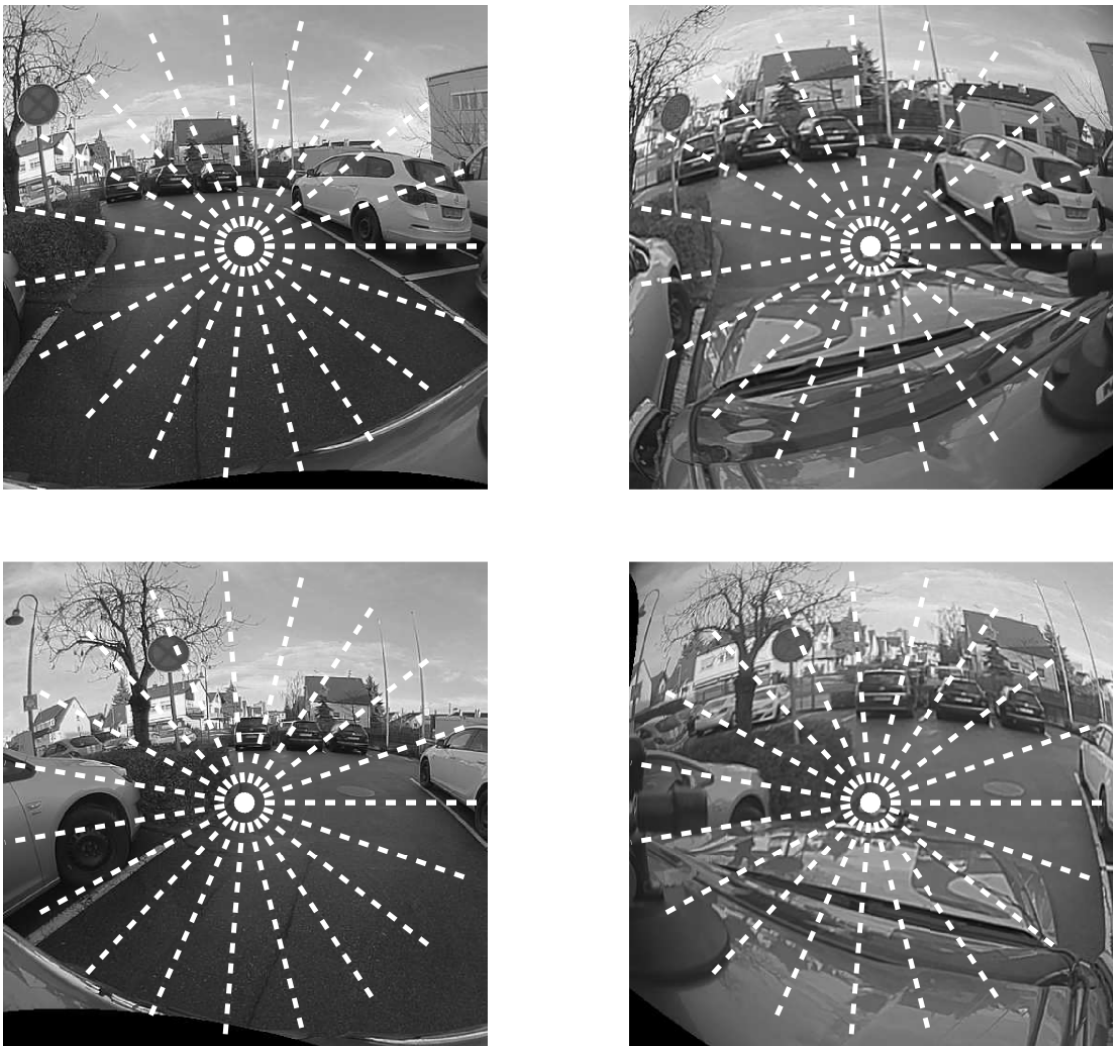


FIGURE 5.27: Result after epipolar rectification. The rectified images correspond to those shown in Figure 5.24. Top row: Front-UpperLeft camera pair. Bottom row: Front-UpperRight camera pair. Reference epipolar lines are drawn in white. As can be observed, epipolar lines are perfectly straight and the epipoles are coincident with the new virtual principal points. Taken from [Esparza et al.,].

5.3.3 Feature matching

After the epipolar rectification is complete, the feature search and matching is performed. Considering conventional epipolar rectification, the matching of features across different images is restricted by the pixel row information. Under the assumption of good calibration, one can be certain that corresponding features lay within one pixel from one another, on the vertical direction. In other words, the disparity can be represented as a one-dimensional offset on the horizontal direction. In the model presented in this section, epipolar lines are not coincident with the horizontal raster lines, but warped on a polar style, centered on the principal

point. Therefore, the correspondence search space is not restricted to a pixel line, but it depends on the angle of each keypoint in normalized image coordinates, with the principal point as reference.

The epipolar line to which an image point (u, v) corresponds is therefore determined by its angular location as seen from the principal point, as in Eq. 5.24.

$$\varphi = \arctan \frac{v'}{u'} \tag{5.24}$$

Although this may appear much less efficient than line search, the search area for every given pixel coordinates can be precomputed offline and efficiently searched based on a look-up table. The area in pixels that corresponds to a fixed polar band depends on the distance ρ to the principal point and can also be accounted for offline. After the correspondence search is performed, 3D reconstruction is conducted by means of triangulation, as described in Section 3.2.3.3.

In Figure 5.28 the curves representing depth as a function of disparity are represented, both with and without the proposed zoom correction factor. It can be observed that a similar range of disparities covers a much smaller range of depths when the zooming effects are corrected. This causes the measurements to be more accurate in the near range of the vehicle.

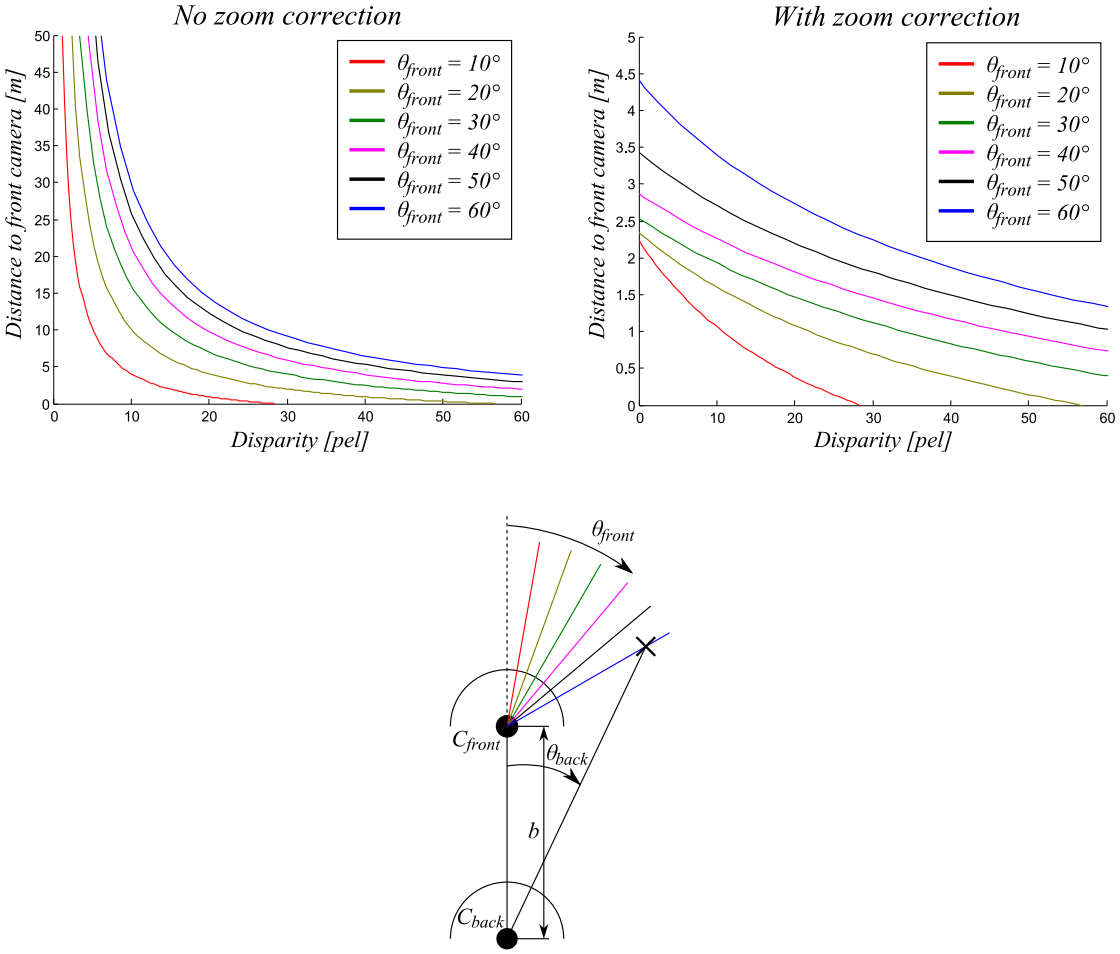


FIGURE 5.28: Estimated depth as function of disparity. The curves represent the distance to the front camera center. Both cases have been considered, with and without zoom correction. With zoom correction it can be observed that disparities cover a smaller range of depths, thus increasing the accuracy in this area, ie. in the vicinity of the vehicle. The curves have been generated according to the real stereo setup, namely $\alpha = 2$, rectified resolution of 512×512 pixels, and maximum distance to virtual optical axis $\theta_{max} = \pi/2$ rad. Taken from [Esparza et al.,].

5.3.4 Experiments

In order to evaluate the proposed camera configuration and polar-epipolar rectification model, different scenes have been recorded with the camera configuration shown in Figure 5.23. A lidar sensor has been considered for reference 3D measurements. The lidar was mounted on the roof of the vehicle and registered to the global common reference with the method proposed in Chapter 4.

The cameras considered in these experiments are similar to those used in Sections 5.1 and 5.2 under the same operating configuration. For each adjacent camera pair on the system setup, previous intrinsic and extrinsic calibration was

available. Intrinsic and extrinsic camera calibration were performed as described in Sections 3.2.1.2 and 3.1.4.

The output resolution of the epipolar-rectified images was set to 512x512 pixels and the zoom correction factor α was set for an expected distance equal to the stereo baseline $d = \|\mathbf{b}\|$, which in the considered setup is equal to 2.24 and 2.28 meters for the two camera pairs. As keypoint detector and descriptor, those proposed by [Rosten and Drummond, 2005] and [Calonder et al., 2010] were considered, respectively and matching was performed based on hamming distance.

Three different scenes have been used for evaluation under the described configuration. The scenes represent a situation: 1) prior to entering a front parking spot, 2) while in a narrow drive way, 3) approaching the end of a parking space. The rectification model presented was applied to the images and results of the 3D reconstruction are shown in the next section.

5.3.5 Results

Results of the rectification process as well as the 3D stereo reconstruction are shown in Figure 5.29. It can be observed how the disparity vectors are radial and virtually intersect on the principal points, which are coincident with the epipoles. Length of the vectors, just like in conventional stereo, encodes depth. In Figure 5.28 it has been shown what the depth-disparity relation looks like in the present setup. Features can be matched in the near vicinity of the epipoles, which is the main goal of the proposed rectification model. The zoom effect was corrected in short distances - comparable to the base between cameras - thus most of the detected common features correspond to areas near the ego vehicle.

Results of the 3D reconstruction are also shown on the right-most column of Figure 5.29, together with the reference measurements of the lidar sensor. It can be observed that drive ways can be measured prior to the vehicle's movement, even in very narrow situations. Furthermore, visual comparison of the point clouds shows a good degree of correspondence with the reference measurements.

In Table 5.4, results are shown for the estimated depths, compared to the 3D reference measurements obtained with the lidar. The results correspond to a L1-norm error metric obtained by projecting all 3D measurements onto the front camera image. Since the lidar was registered to the global reference system, a one-dimensional comparison is possible. This has been explained in detail in

Chapter 4. In all evaluated scenes the absolute error is below 1.5 meters for 90% of the measurements and the distance-relative error under 20%. The second scene, corresponding to the narrow driving path, shows the best overall performance with a relative error below 14% and an absolute error below 0.6m for 90% of all measurements.

TABLE 5.4: Accumulated absolute and relative error analysis based on a L1-norm error metric. The results on this table correspond to the distance to the front camera center after reprojection of every 3D measurement, compared to the reference lidar measurements. Quartile information is included since it is representative for discussion of results. Data from [Esparza et al.,].

Frame ID	Masurements	Q50 [m]	Q75 [m]	Q90 [m]	Q50 [%]	Q75 [%]	Q90 [%]
1	10203	0.24	0.51	1.40	05.86	11.01	18.21
2	10619	0.18	0.30	0.60	04.43	07.55	13.23
3	15407	0.08	0.16	0.63	03.59	06.62	15.82

All values in Table 5.4 correspond to measurements without time accumulation and were obtained from the single frames shown in Figure 5.29.

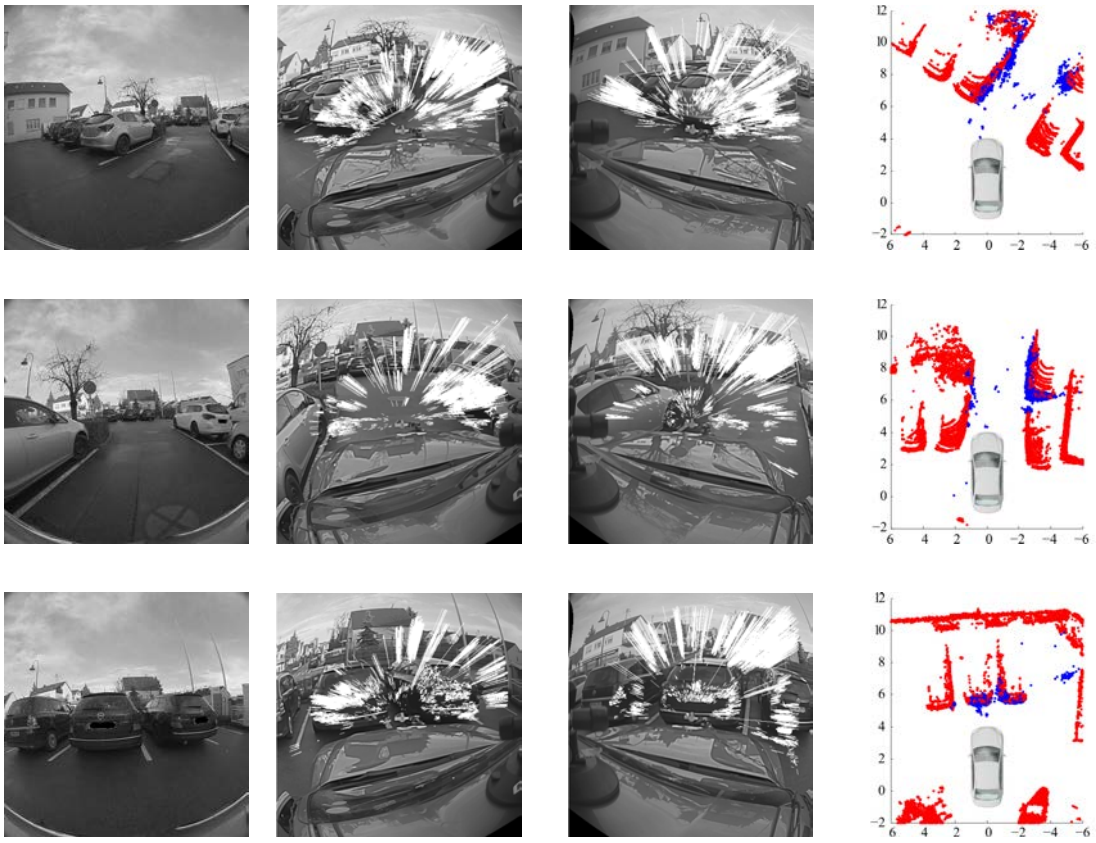


FIGURE 5.29: Results obtained on three different scenes. Top row: Before entering a parking spot. Middle row: on a narrow drive way. Bottom row: End of front parking, with a very close front vehicle. Left column: Section of the original front fisheye image. Central left and central right columns: Epipolar-rectified images (white: result of the feature matching). Right column: 3D reconstruction based on stereo measurements (blue) and reference lidar (red); distance indications are given in meters for both x- and y-axes. Taken from [Esparza et al.,].

For comparison, the same scenes have been recorded with the conventional four-camera setup, and 3D reconstruction has been carried out in the way described in Section 5.1. Measurements are presented in Figures 5.30, 5.31, and 5.32.

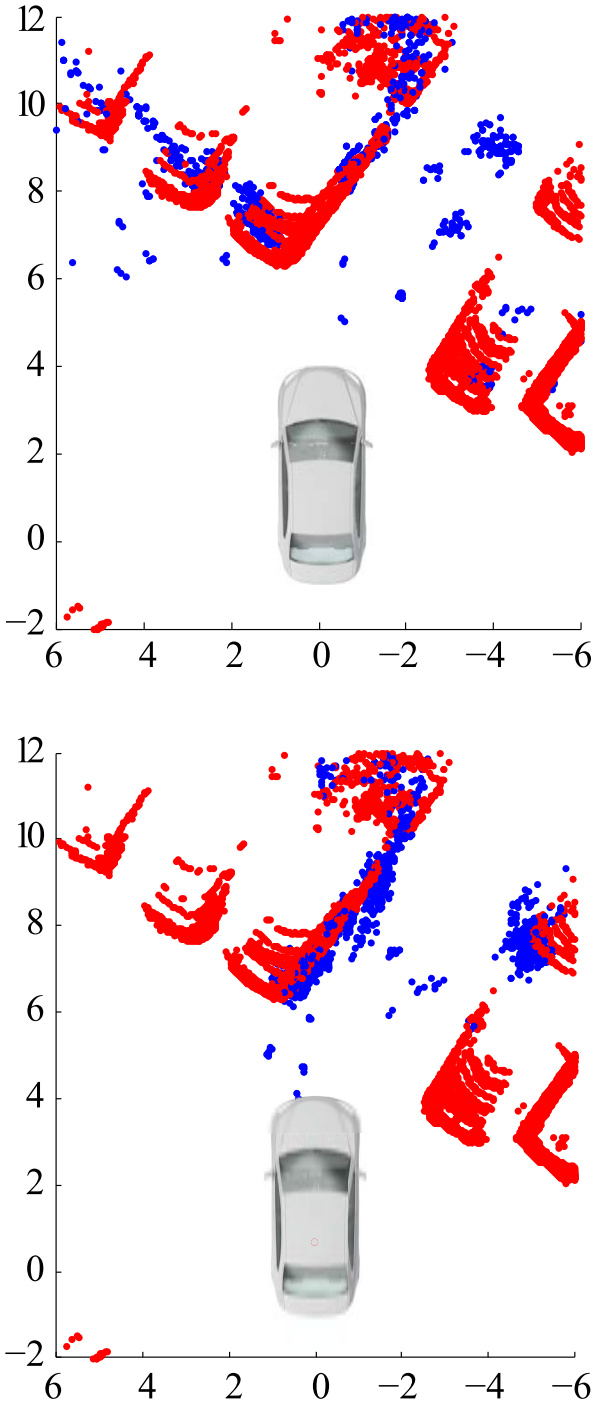


FIGURE 5.30: Comparison of the standard four-camera configuration with the extended setup presented in this section. Top: measurements from the standard four-camera configuration, considering the reconstruction scheme proposed in Section 5.1. Bottom: measurements obtained with the configuration presented in this section. Blue: stereo measurements. Red: reference lidar. Distance indications are given in meters for both x- and y-axes.

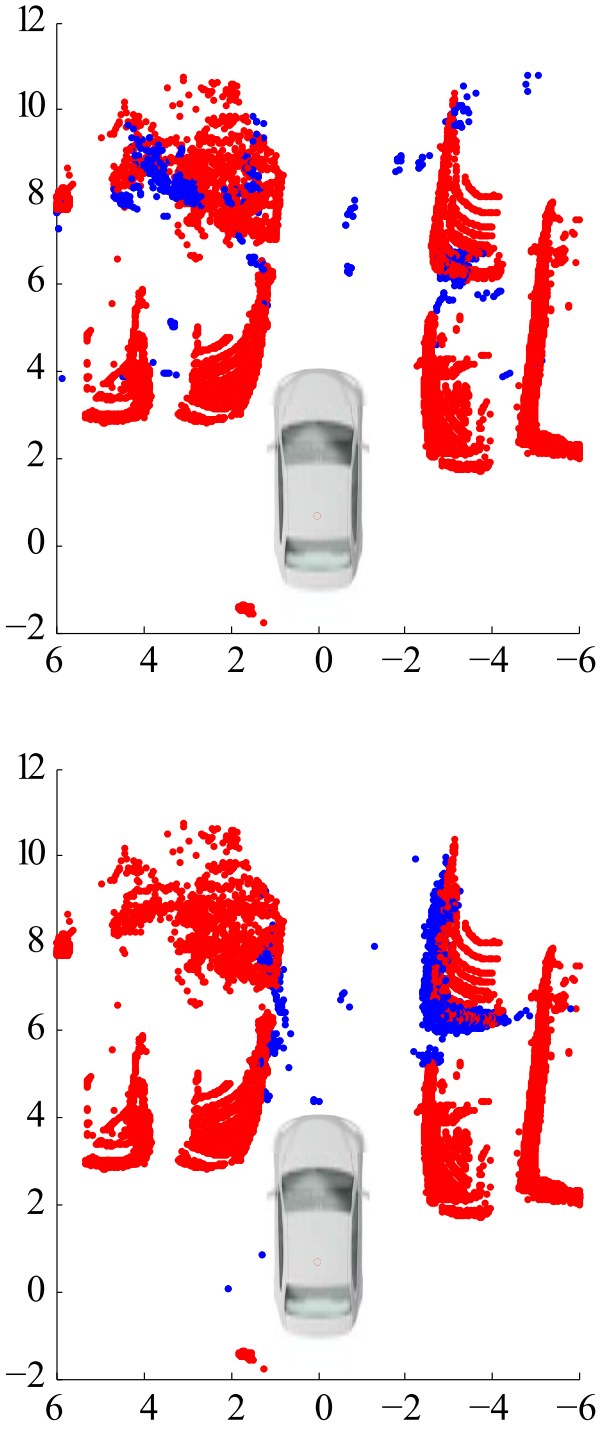


FIGURE 5.31: Comparison of the standard four-camera configuration with the extended setup presented in this section. Top: measurements from the standard four-camera configuration, considering the reconstruction scheme proposed in Section 5.1. Bottom: measurements obtained with the configuration presented in this section. Blue: stereo measurements. Red: reference lidar. Distance indications are given in meters for both x- and y-axes.

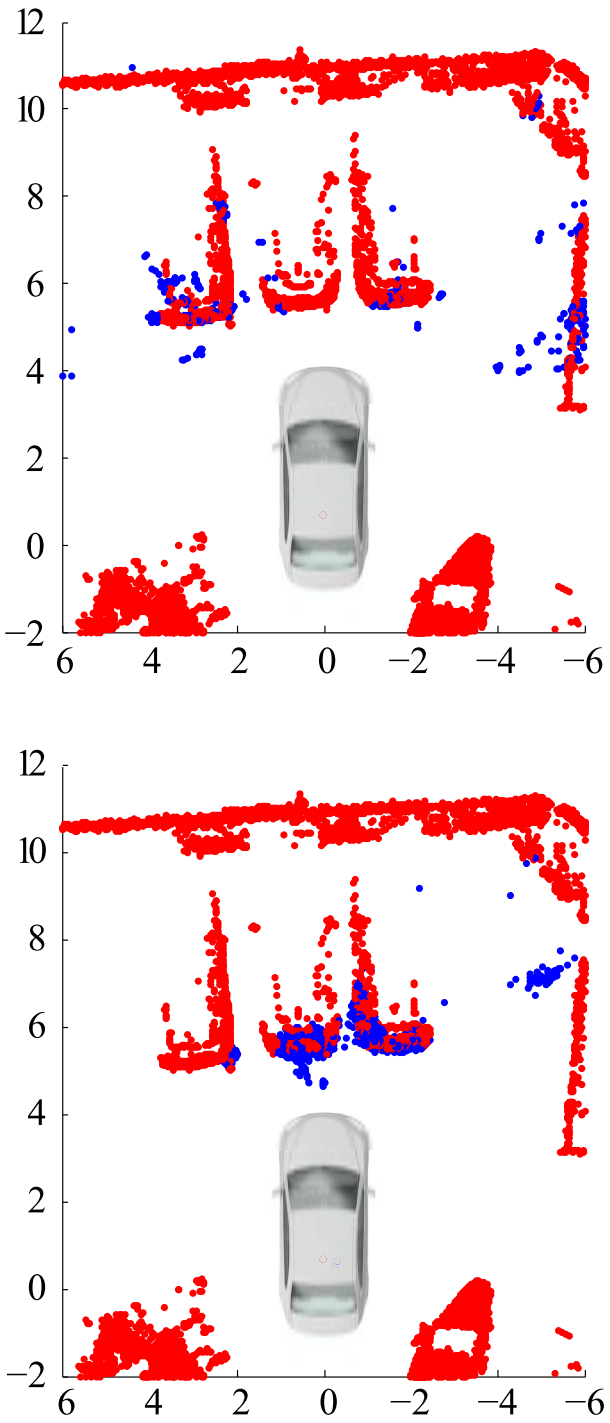


FIGURE 5.32: Comparison of the standard four-camera configuration with the extended setup presented in this section. Top: measurements from the standard four-camera configuration, considering the reconstruction scheme proposed in Section 5.1. Bottom: measurements obtained with the configuration presented in this section. Blue: stereo measurements. Red: reference lidar. Distance indications are given in meters for both x- and y-axes.

The results presented in Figures 5.30, 5.31, and 5.32 shows the limitations of the conventional four-camera setup on narrow drive paths and parking situations,

compared to the extended setup presented in this section. Much narrower paths can be measured with objects closer to the ego vehicle.

5.3.6 Discussion

The proposed camera setup has shown to allow for 3D reconstruction on narrow drive ways and parking spaces, and parallax was reduced compared to standard surround view configurations. Based on the described polar-epipolar rectification model, good reconstruction rates have been observed in image areas in the vicinity of the epipoles.

A zoom correction factor has been introduced that accounts for the scale differences across views, due to the relative forward translation that exists between the cameras. In particular, a value of $\alpha = 2$ was considered, which enhances the matching of features that correspond to objects located a distance $d = \|\mathbf{b}\|$ from the reference front camera. Depth curves with respect to disparity values have been generated for the current setup, both with and without zoom correction for comparison. It has been shown that the proposed correction allows for larger disparities to represent shorter distances, which is a benefit for the near-range application under consideration.

In order to evaluate the accuracy of the 3D measurements in the three situations considered, a lidar sensor was used for benchmarking. All measurements were conducted without time accumulation and the results shown for each scene correspond to a single frame from each camera. These results compare well to other state of the art parking assistance systems [Unger et al., 2014] and the new configuration largely outperforms the four-camera setup, as considered in Section 5.1. The proposed camera mounting allows to share the cameras with standard surround visualization systems and the estimated depth can directly be utilized by other higher level functionalities, like pedestrian detection or autonomous parking.

Chapter 6

Visualization

This chapter is aimed at describing both the algorithmical processing required to give an interpretation to the 3D data acquired by means of stereo vision, as well as the approaches utilized for optimal visualization. The chapter is divided in four main sections. Firstly, in Sections 6.1 and 6.2, an approach is proposed in order to describe the geometry that can be used as support mesh for the Image Based Rendering (IBR), with or without depth information, and in Section 6.3 a series of visual enhancements based on depth information are proposed. Section 6.4 presents a front inspection view which does not rely on surrounding spatial information.

6.1 Static Mesh Definition for IBR

As described in Section 3.3.1, a wireframe model has to be defined as support for the IBR. This model is given as a mesh of triangles, which are described in terms of vertices. This section presents how the vertices can be defined in absence of depth information and is based on the work of [Shimizu et al., 2010]. Since the authors did not provide implementation details, the own solution of the author is presented in the following.

In the following the two main parts in the projection surface are differentiated: the flat *floor* area, and the elevated *wall* area. The floor area is used to represent the vicinity of the vehicle and is based on the assumption of near-range free space. The surface wall is used to create the elevation effect that allows to project high objects, e.g. other vehicles, pedestrians or buildings. Assuming no information about the distance to these objects, a fixed projection depth ρ_{max} can be considered.

The construction of both floor and wall surfaces is based on a polar mesh that represents the environment, thus a set of N radial directions is used as support. Each n-th direction corresponds to an angle θ_n, defined such that the 360° FoV around the vehicle is uniformly covered.

Similarly, a uniformly distributed set of distances ρ_m and angles α_q are defined for floor and wall as in Eqs. 6.1 and 6.2, respectively.

$$\rho_m = \frac{m}{M-1}\rho_{max}, \quad \text{with} \quad m \in [0, M-1] \tag{6.1}$$

$$\alpha_q = \frac{q}{Q-1}\frac{\pi}{2}, \quad \text{with} \quad q \in [0, Q-1] \tag{6.2}$$

A graphic representation of the vertices that define the surface mesh is shown in Figure 6.1.

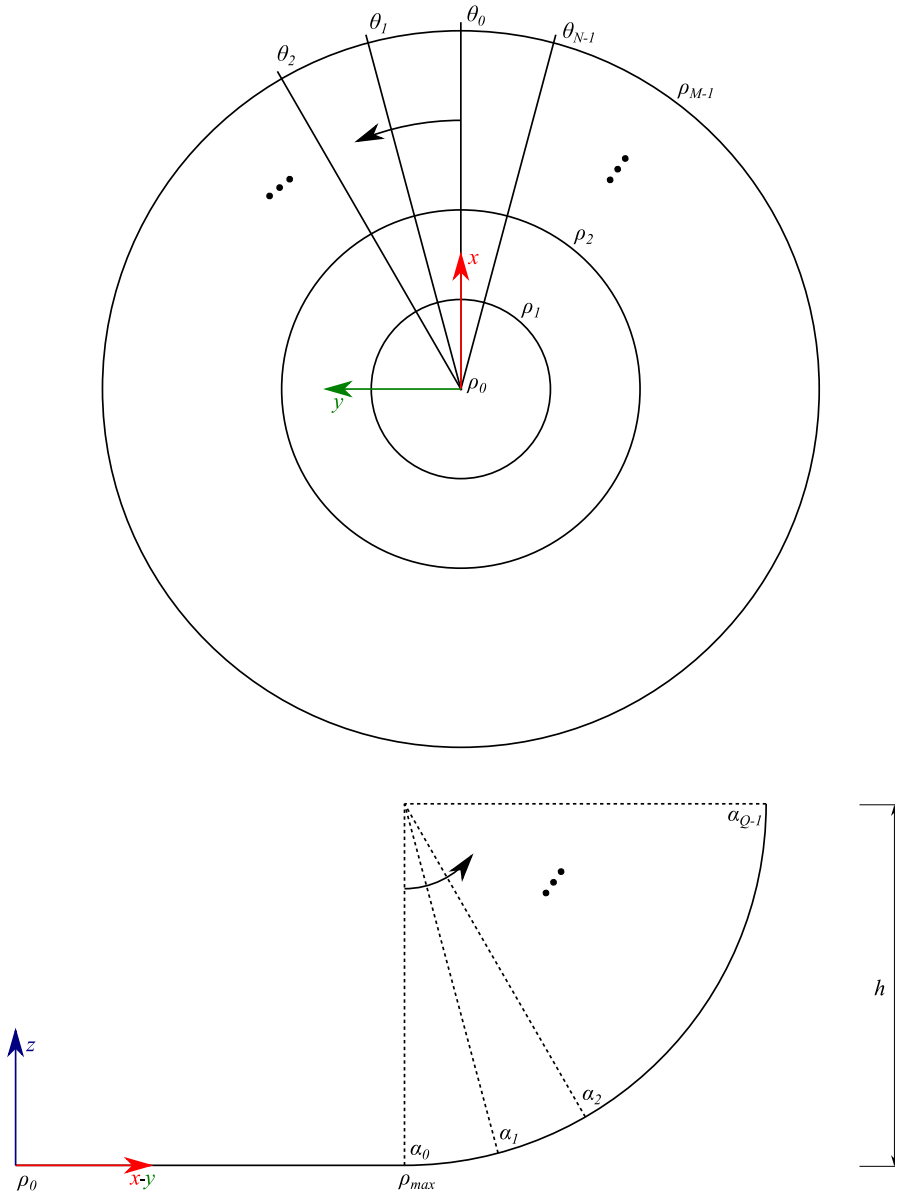


FIGURE 6.1: Definition of the static projection surface construction. Top: flat *floor* surface. The polar mesh is defined by a set of N reference directions and M distances. Bottom: elevated *wall* surface. The mesh is defined by the same N reference directions as the floor and a set of Q reference angles.

The construction of vertices $\boldsymbol{V}^f_{m,n}$ belonging to the flat floor in the vicinity of the vehicle (superindex f) is given by Eqs. 6.3 and 6.4, with a constant height value of $z^f = 0$.

$$x^f_{m,n} = \rho_m \cos\theta_n \tag{6.3}$$

$$y^f_{m,n} = \rho_m \sin\theta_n \tag{6.4}$$

On the elevated walls of the surface (superindex w), the vertex coordinates of $\boldsymbol{V}^w_{q,n}$ are computed by means of Eqs. 6.5, 6.6 and 6.7, where the parameter h represents the maximal height of the projection surface.

$$x^w_{q,n} = [\rho_{max} + h \sin \alpha_q] \cos \theta_n \tag{6.5}$$

$$y^w_{q,n} = [\rho_{max} + h \sin \alpha_q] \sin \theta_n \tag{6.6}$$

$$z^w_{q,n} = h\,(1 - \cos \alpha_q) \tag{6.7}$$

The wireframe mesh is then defined as a list of triangles built with the vertices previously described. According to these coordinates, the floor mesh is defined with the steps described in pseudocode in the following:

triang_list_floor = {}
for m = 0..M-1 **do**
  for n = 0..N-1 **do**
    triang_list_floor ← *triang* $\left(\boldsymbol{V}^f_{m,n},\, \boldsymbol{V}^f_{m+1,n},\, \boldsymbol{V}^f_{m+1,n+1} \right)$
    triang_list_floor ← *triang* $\left(\boldsymbol{V}^f_{m,n},\, \boldsymbol{V}^f_{m+1,n+1},\, \boldsymbol{V}^f_{m,n+1} \right)$
  end for
  triang_list_floor ← *triang* $\left(\boldsymbol{V}^f_{m,N-1},\, \boldsymbol{V}^f_{m+1,N-1},\, \boldsymbol{V}^f_{m+1,0} \right)$
  triang_list_floor ← *triang* $\left(\boldsymbol{V}^f_{m,N-1},\, \boldsymbol{V}^f_{m+1,0},\, \boldsymbol{V}^f_{m+1,0} \right)$
end for

Pseudocode for wireframe mesh definition of the surface's floor.

As for the wall mesh, a similar definition exists:

$triang\_list\_wall = \{\}$
for q = 0..Q-1 **do**
  for n = 0..N-1 **do**
    $triang\_list\_wall \leftarrow triang\left(\boldsymbol{V}^{w}_{q,n},\, \boldsymbol{V}^{w}_{q+1,n},\, \boldsymbol{V}^{w}_{q+1,n+1}\right)$
    $triang\_list\_wall \leftarrow triang\left(\boldsymbol{V}^{w}_{q,n},\, \boldsymbol{V}^{w}_{q+1,n+1},\, \boldsymbol{V}^{w}_{q,n+1}\right)$
  end for
  $triang\_list\_wall \leftarrow triang\left(\boldsymbol{V}^{w}_{q,N-1},\, \boldsymbol{V}^{w}_{q+1,N-1},\, \boldsymbol{V}^{w}_{q+1,0}\right)$
  $triang\_list\_wall \leftarrow triang\left(\boldsymbol{V}^{w}_{q,N-1},\, \boldsymbol{V}^{w}_{q+1,0},\, \boldsymbol{V}^{w}_{q+1,0}\right)$
end for

Pseudocode for wireframe mesh definition of the surface's wall.

Once the vertex coordinates are computed and the triangles defined, the corresponding 2D texture coordinates can be computed. This step has already been described in detail in Sections 3.3.2 and 3.2.1 by means of the model-to-camera transfomation and fisheye projection model, respectively. It is therefore not explained here.

In the next section an approach is proposed in order to dynamically adapt the shape of the projection surface based on 3D depth information obtained either by means of stereo measurements, or through fusion with measurements from other vehicle sensors.

6.2 Dynamic Render Geometry from Depth Measurements

In previous chapters it has been shown that several approaches exist to recover depth information in the context of surround view systems. These are either based on camera information or on measurements from other sensors on a vehicle.

At the same time, in the previous section it has been described how a static mesh of triangles can be built as support for the IBR. The next step is to design an approach that allows the projection surface to dynamically adapt to the available depth information.

The solution adopted in this thesis is an occupancy grid based approach, although other possibilities exist [George and Borouchaki, 1998], [Shewchuk, 2002]. In the following, an introduction into occupancy grids is given together with the description of their application for dynamic mesh generation.

6.2.1 Occupancy Grids

An occupancy grid can be defined as an approach for world perception and modelling that uses a probabilistic tesselated representation of spatial information [Elfes, 1989].

Occupancy grids are especially interesting because they provide an abstraction layer between the sensors and the function layers. Any sensor that can provide depth information on a given direction can feed it into the grid, which acts as support for data fusion.

Most occupancy grid approaches are based on two-dimensional maps [Thrun, 2003]. For this thesis, the basic case of a 2D binary map is considered, where the state of every bit defines whether a location on space is occupied by an object or not. In Figure 6.2 an example of 2D binary grid is given.

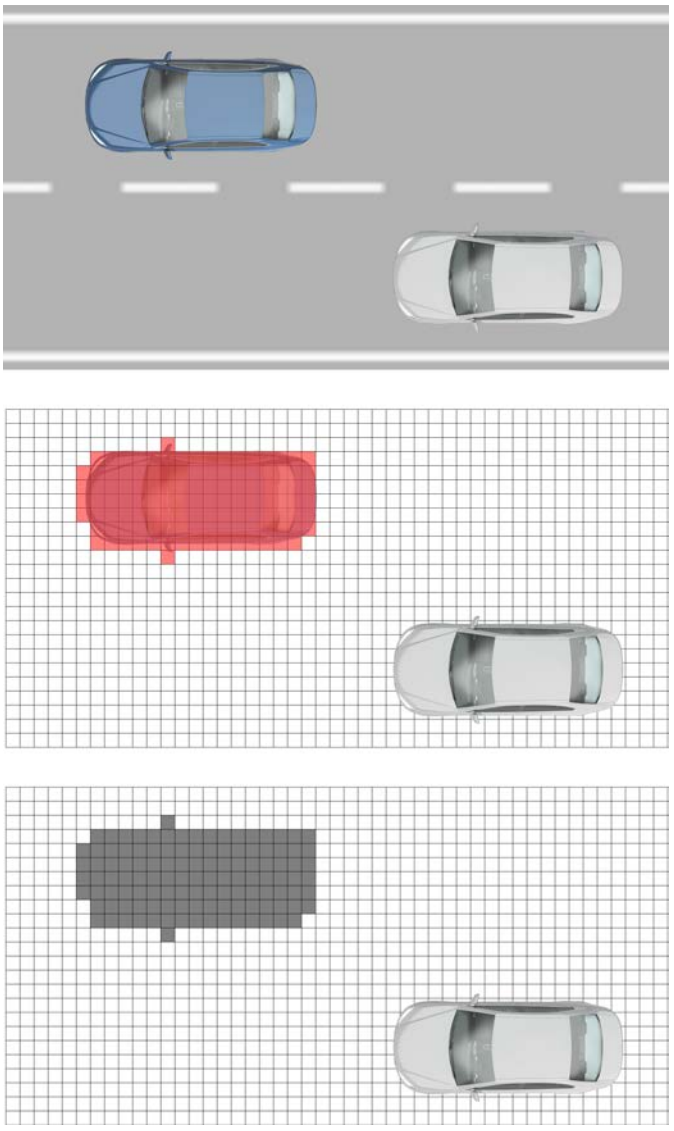


FIGURE 6.2: Occupancy grid. Every cell on the cartesian grid corresponds to a space in the real world around the ego vehicle. The binary state of the cell represents whether this space is occupied or not by an object.

Based on the grid data, different approaches can be considered to dynamically generate a render geometry. The approach considered in this thesis is based on the static radial surface described in Section 6.1, extended to adapt its depth on every radial direction. In this way, the occupied environment can be approximated by the mesh of triangles. This is presented in detail in the following section.

6.2.2 Dynamic Adaptation of the Projection Surface

In order to dynamically adapt the projection surface to the occupancy grid on each instant, a radial extractor function is introduced that allows for iteration over all

key directions that define the mesh construction as it was described in Section 6.1.

In this way, ρ_{max} can be replaced by ρ^n_{max} for each radial extractor over θ_n, thus a good depth approximation can be achieved on every radial direction with independence from the others.

It is possible to define b_{ij} as the binary state of the (i, j)-th grid cell and ϕ_{ij} and R_{ij} its angle and distance as from the grid's center, respectively. A tuple Θ_{ij} is defined as in 6.8 that represents the two consecutive radial directions between which the (i, j)-th cell is contained.

$$\Theta_{ij} = \{\theta_n, \theta_{n+1}\} \text{ such that } \theta_n \leq \phi_{ij}, \theta_{n+1} > \phi_{ij} \tag{6.8}$$

For every radial direction θ_n, a group of distances f_{θ_n} can be defined that contains occupancy information along it.

$$f_{\theta_n} = \{R_{ij} \mid \theta_n \in \Theta_{ij}, b_{ij} = 1\} \tag{6.9}$$

In the same way, a test $\tau(i, j, n)$ can be defined that describes whether the $n-$th radial extractor over the radial direction θ_n intersects the (i, j)-th grid cell. This information is especially useful near the center of the extractor functions, since not every cell intersected by a direction θ_n contributes to f_{θ_n} according to its definition.

$$\tau(i, j, n) = \begin{cases} 1 & \text{if } \theta_n \text{ intersects cell } (i, j) \\ 0 & \text{otherwise} \end{cases} \tag{6.10}$$

The set of distances to occupied cells that are intersected by the direction θ_n is defined by g_{θ_n} as in Eq. 6.11.

$$g_{\theta_n} = \{R_{ij} \mid \tau(i, j, n) = 1, b_{ij} = 1\} \tag{6.11}$$

In Figure 6.3 a representation is given for the grid cells that may determine f_{θ_n} and g_{θ_n} on a given direction θ_n. From these, only the occupied cells will contribute to the final estimated depth.

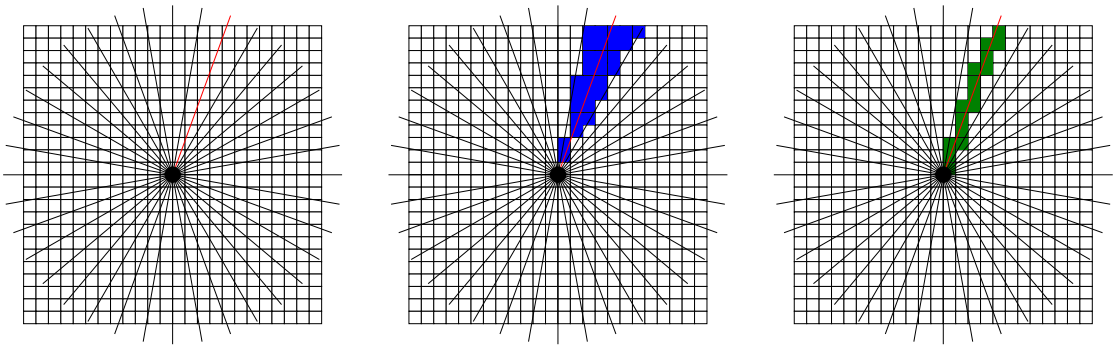


FIGURE 6.3: Grid cells C_{ij} that can determine f_{θ_n} and g_{θ_n} for a given direction θ_n.
Left: Reference direction θ_n.
Center: $\{C_{ij} \mid \theta_n \in \Theta_{ij}\}$.
Right: $\{C_{ij} \mid \tau(i,j,n) = 1\}$.

The resulting depth ρ^n_{max} corresponding to θ_n is computed by means of Eq. 6.12. Distances representing the closest objects are given preference, since these are the most relevant for maneuvering.

$$\rho^n_{max} = \min \{ \ f_{\theta_n} \ \cup \ g_{\theta_n} \ \} \tag{6.12}$$

In Figure 6.4 a visual description of the resulting radial distances after processing the occupancy grid is given.

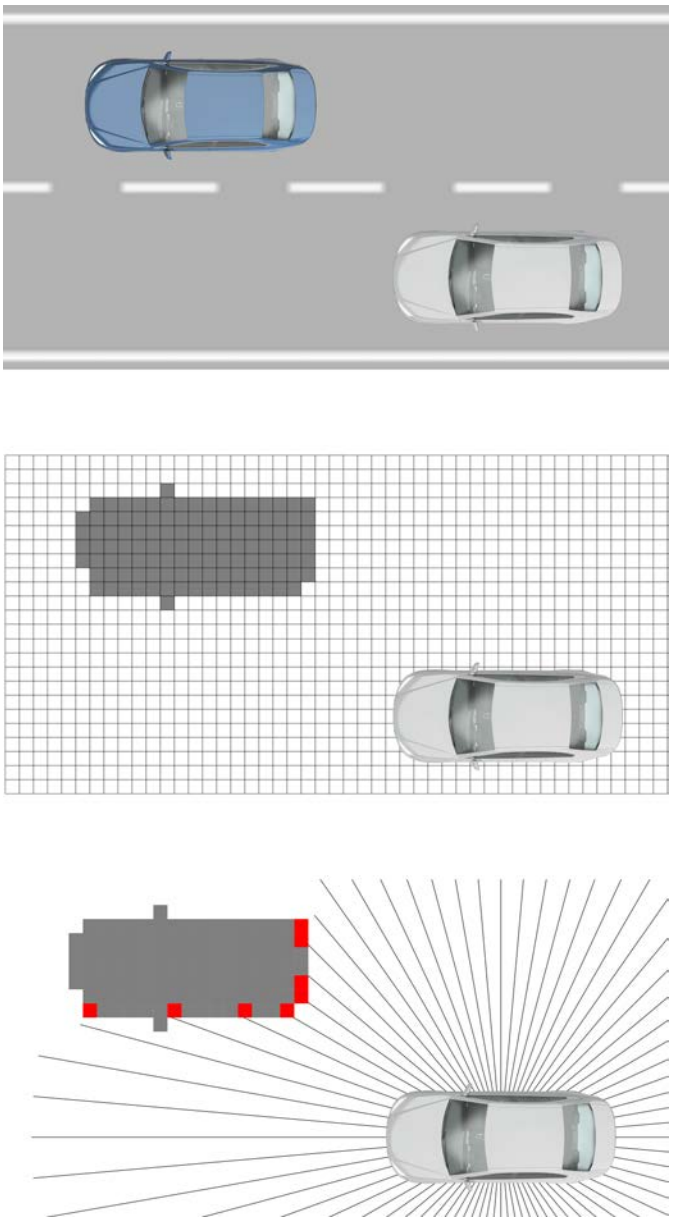


FIGURE 6.4: For every radial direction described by its angle θ_n the shortest distance to an occupied cell on the grid is searched. The corresponding maximum radial depth ρ^n_{max} is estimated so that the mesh can be adapted accordingly. Red: occupied cells that define the radial depth ρ^n_{max} in the current frame.

Based on the depths extracted from the occupancy grid, Eq. 6.1 is replaced by Eq. 6.13.

$$\rho^n_m = \frac{m}{M-1}\rho^n_{max}, \quad \text{with} \quad m \in [0, M-1] \tag{6.13}$$

Similarly, Eqs. 6.3, 6.4 are replaced by Eqs. 6.14, 6.15, and Eqs. 6.5, 6.6 by Eqs. 6.16, 6.17, respectively.

$$x^f_{m,n} = \rho^n_m \cos\theta_n \tag{6.14}$$

$$y^f_{m,n} = \rho^n_m \sin\theta_n \tag{6.15}$$

$$x^w_{q,n} = [\rho^n_{max} + h \sin\alpha_q] \cos\theta_n \tag{6.16}$$

$$y^w_{q,n} = [\rho^n_{max} + h \sin\alpha_q] \sin\theta_n \tag{6.17}$$

With the new expressions, a depth is considered on every direction from the center of the vehicle, so the mesh of triangles is dynamically adapted to fit the available occupacy information. Although these expressions are thought to be used independently for each new frame, it is common practice that the occupancy grids accumulate occupancy information over time in order to increase their accuracy. Nevertheless, with the proposed approach certain artefacts still remain. In following sections extra visualization enhancements are proposed, based on the available depth information.

6.3 Depth-based Visualization Enhancements

This section focuses on the rendering enhancements utilized for ultimately improving the comprehension and understanding of the scene by the driver. The problems discussed include the density of triangles after dynamic mesh adaptation and stitching of multiple images on a composite view.

6.3.1 View-dependent Projection Surface

In the description of the system done until now, a projection surface centered with respect to the vehicle has been considered, where a virtual camera freely navigates around it. For smooth surfaces, like the depth-independent one presented in [Shimizu et al., 2010], this is a good approach since the projection of the mesh onto the virtual rendering camera produces a quite homogeneous distribution of triangles over the image plane. Under dynamic solutions, like the one presented in 6.2.2, the density of polygons on the viewport area is not uniform due to the different depths considered. This effect is depicted in Figure 6.5.

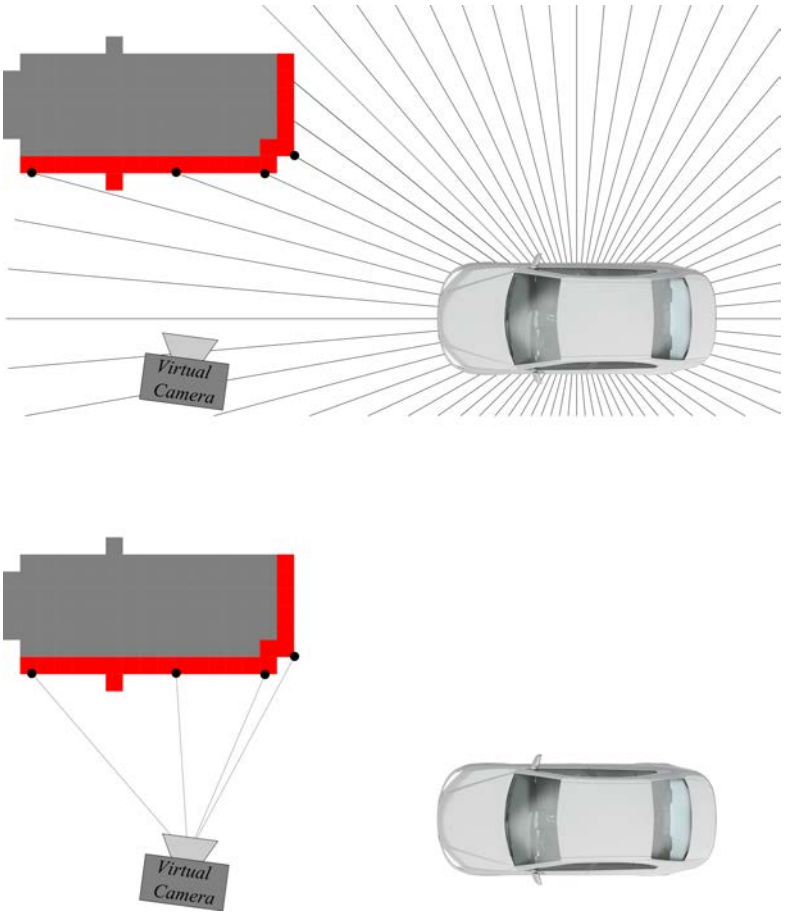


FIGURE 6.5: Radial depth extraction, independently from render camera position. Top: Radial depth extractor functions. Bottom: Projection of the depth reference vertices to the render camera. The observed mesh does not have a uniform distribution of triangles over the image plane.

In order to correct this effect, a new mesh-camera paradigm is presented, where a link is established between the virtual camera position and the center of the radial mesh. In particular, this is implemented by means of an XY cartesian offset of the polar coordinates considered in the previous sections. In Figure 6.6 a comparison of both approaches is shown.

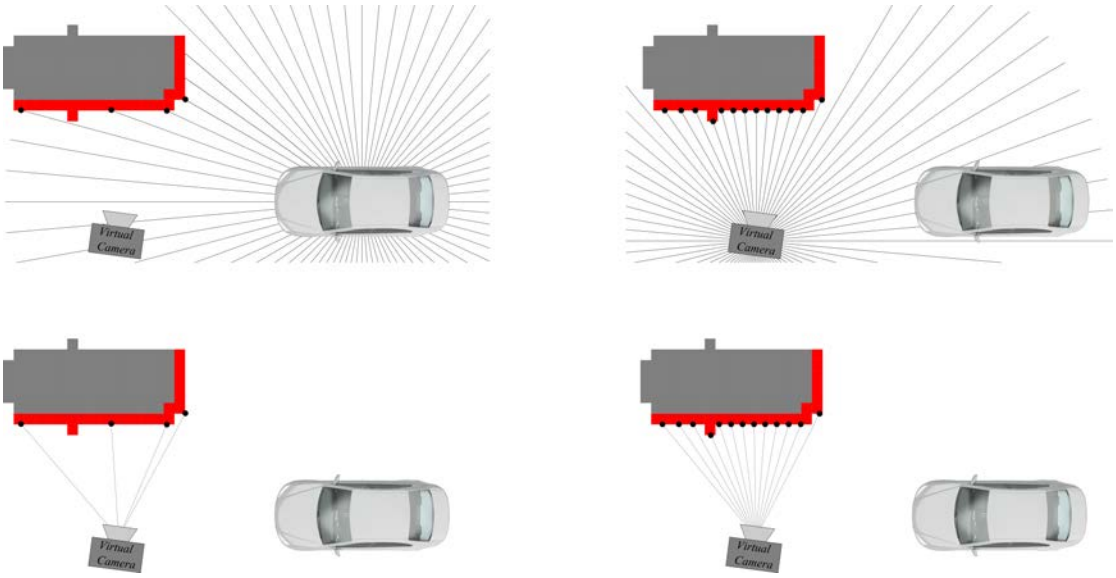


FIGURE 6.6: New mesh definition paradigm. Left: Standard case, with the center of the radial depth extractor functions aligned with the vehicle. Right: New paradigm - the center of the radial extractors is displaced based on the render camera position. The apparent density of triangles is always uniform with respect to the render camera.

By considering a set of radial extractors which center is aligned with the virtual render camera, the apparent density of mesh triangles on the image plane is again uniform. With the described approach, artefacts arising from the dynamic surface deformation can be partly removed, improving the final visualization.

One of the problems that still remain open is the seam that exists between images from the different cameras. In order to combine multiple images on a composite single view, a stitching scheme seems quite necessary. This issue is addressed in the next section.

6.3.2 Stitching

Image stitching can be defined as the process of combining different images of a common scene obtained from different positions and orientations [Szeliski, 2006].

Given the large parallax that exists on conventional surround-view camera systems, there is usually no single 2D image transformation that can stitch images from adjacent cameras at all depth levels. A solution commonly adopted is to define a static seam with an alpha blending area around it [Shimizu et al., 2010].

In this section, a novel method is proposed that allows for efficient dynamic stitching, based on occupancy grid data. The main idea of this method is not to perform

additional image transformations (appart from the projection onto the render surface), but to dynamically search for the optimal seam location. To achieve this, a situation like in Figure 6.7 is considered.

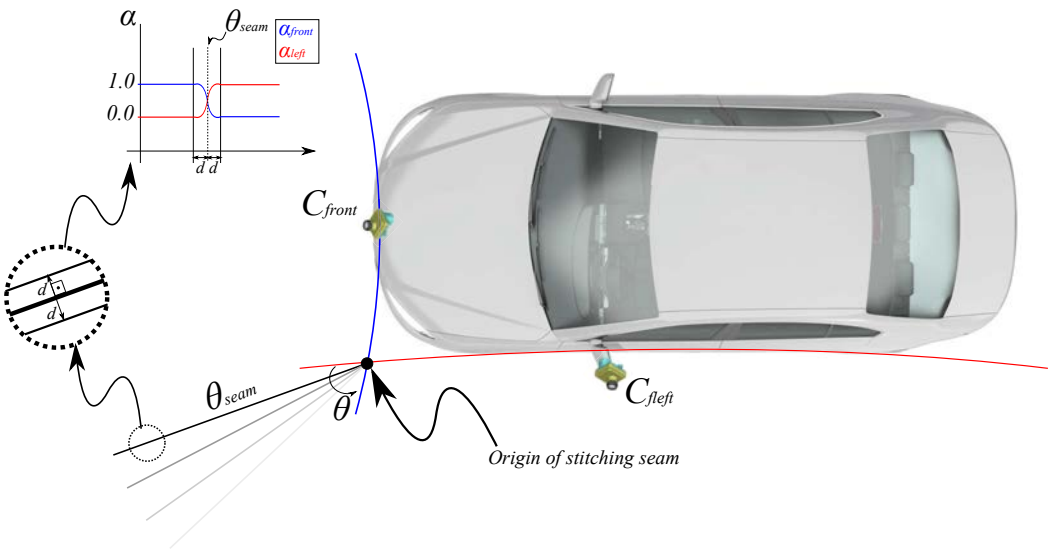


FIGURE 6.7: Dynamic stitching seam. In order to stitch images from different cameras on the composite surround visualization, stitching seams are defined between adjacent cameras. Every seam defines a certain width around itselft, where alpha blending is applied. The dynamic approach is based on occupancy grid information and aims to avoid elevated objects lying on the stitching band.

The origin of the stitching seam is defined as the closest point to the vehicle where the fields of view of two adjacent cameras overlap. From this point, a radial extractor as the one presented in Section 6.2.2 is used to perform a polar analysis of the occupancy on the area where the fields of view overlap.

A binary classification is performed for every radial direction, either as occupied or free, attending to depth. Based on this classification, a clustering is done in order to detect large occupied and non occupied areas.

In the current configuration, the focus is to avoid stitching seams going through nearby elevated objects, since the projections of these differ mostly across different camera views, making the seam very visible on the composite view. Therefore, the stitching seam is transferred to the widest non occupied cluster. The algorithm can be summarized in the following steps:

1: Extract depth information in a polar style, from the seam origin
2: Classify each reference direction as occupied or non occupied
3: Perform clustering of occupied ($C_{occupied}$) and non occupied (C_{free}) radial directions
4: Select the widest non ocupied cluster to contain the seam $C_{seam} \leftarrow \text{argmax}_C \, width\{C_{free}\}$
5: Set the seam at the center of the chosen cluster $\theta_{seam} \leftarrow center\{C_{seam}\}$
6: Assign a stitch band around the seam
7: Perform alpha-blending on the stitching band

Pseudocode for stitching algorithm. Steps required to dynamically compute the optimal seam based on occupancy information and blend a composite view.

With this approach, it can be avoided that elevated objects in the near vicinity of the vehicle are intersected by the stitching seam, which effects are very perceptible to the eye.

The depht-based visual enhancements proposed in this section can be summarized in the flow chart shown in Figure 6.8.

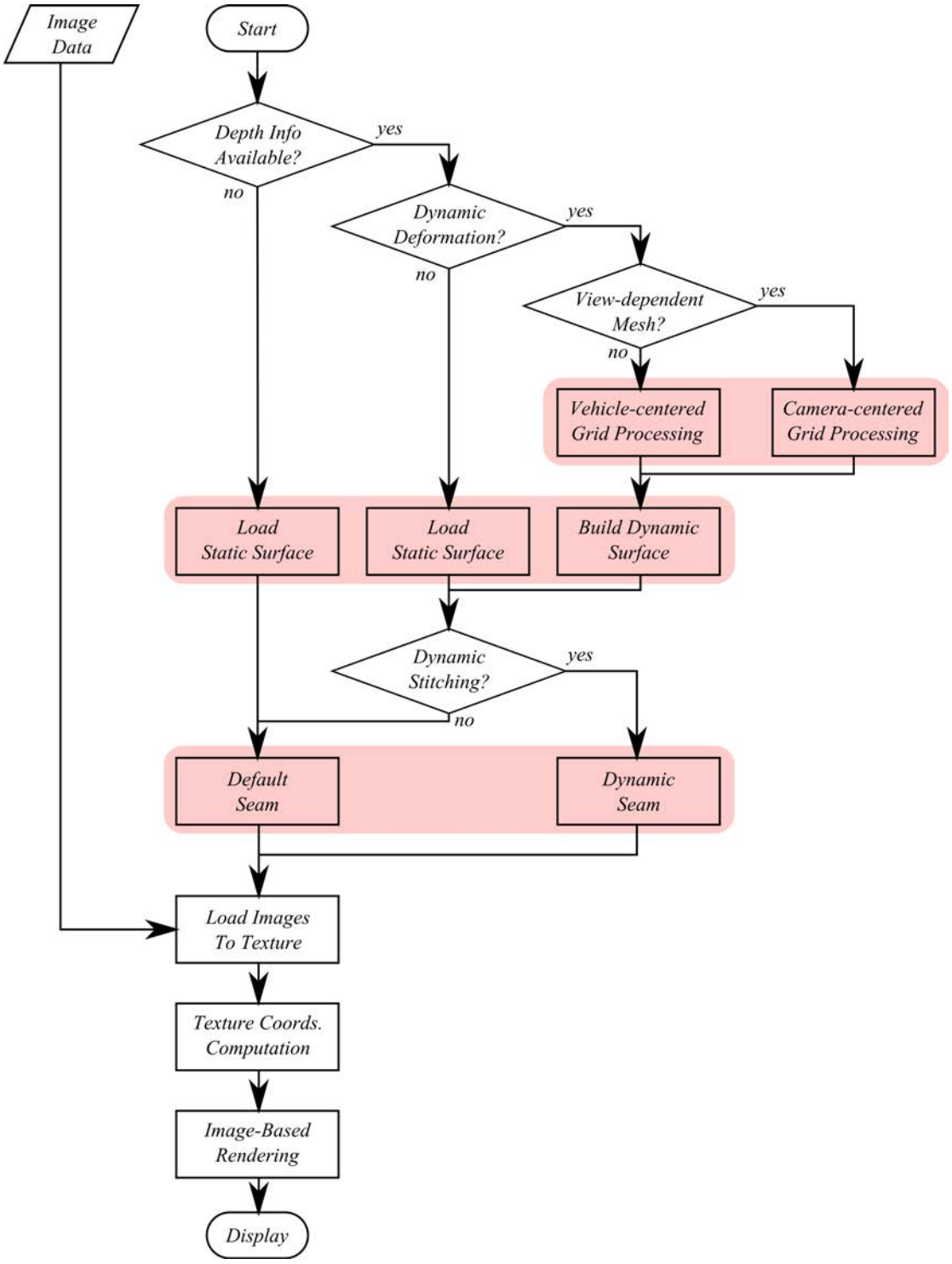


FIGURE 6.8: Flowchart for the visualization of each frame. The different branches show the options on each step previous to the rendering process, depending on the availability of depth information. All the enhancements that have been proposed in this section are highlighted in red.

6.4 Special Views Without Depth Information

Despite the benefits of having an external virtual camera that can be freely moved around the vehicle, there also exist situations that do not necessarily require a change of perspective. As an alternative, an image warping can be conducted that enhances certain visualization aspects or corrects a given kind of distortion.

If a unique center of projections can be considered for a camera, a 2D nonlinear transformation function can be applied to the original images that fulfills some requirements with respect to distortions, without need of 3D depth information.

In the following, an example is given of a virtual view that can be described independently from environmental information.

6.4.1 Front Inspection View

In the standard surround view configuration, as presented in Section 2.1.1, there is usually an exposed front camera with a very wide field of view. In driving situations where the path of view of the driver is reduced laterally, eg. when leaving a garage or entering a road crossing, the front camera normally gets an earlier view than the driver. Given the large fields of view provided, a 180° front inspection view can be generated.

The method proposed here is aimed at fulfilling these requirements:

- Fisheye distortions have to be corrected, so that vertical lines are displayed vertical.
- A complete 180° horizontal field of view must be preserved.
- The horizon line must be centered in the image and must be horizontal.

This is achieved by conducting a cylindrical warping of the original image. The projection cylinder is defined such that its axis is parallel to the floor's normal. In this way it can be guaranteed that the vertical lines are projected as vertical on the new image. Furthermore, the horizon line is described by the intersection of the image plane with the horizontal plane passing through the camera's center of projections. Since the normal of this plane is parallel to the axis of the cylinder, the horizon line is perfectly horizontal on the new warped image. The height of

the cylinder can be used to control the position of the horizon line over the new projection.

6.5 Results

Prototypic implementation of the concepts presented in this chapter has been carried out, and this section contains visualization results in order to evaluate the level of enhancement that can be achieved.

Figures 6.9, 6.10 and 6.11 show real examples of the dynamic depth-based projection surface deformation. As comparison a static mesh is considered as described in Section 6.1.

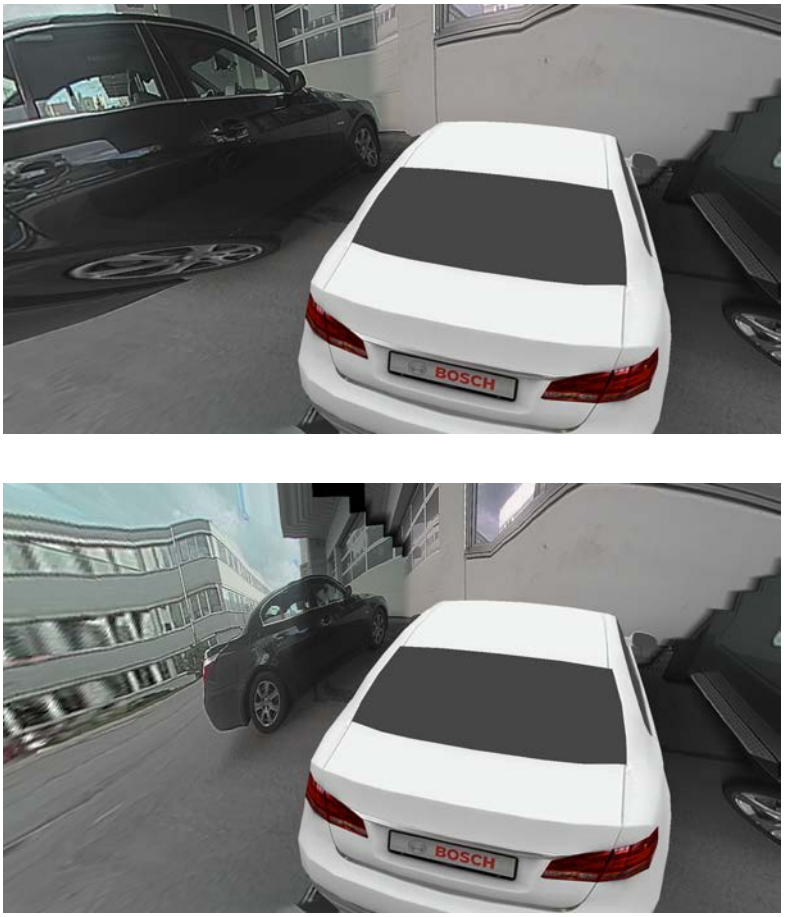


FIGURE 6.9: Example of dynamic projection surface adaptation - Scene 1. Based on occupancy grid information a depth extractor is applied as presented in Section 6.2. The effects of the adaptive mesh can be observed on the dark vehicle, especially on the back wheel. Without depth information the projection surface is set to its default shape, thus wheels are projected on the floor. On the contrary, with correct depth information, the walls of the surface approximate well the side of the car, providing a more intuitive representation. Top: default static mesh. Bottom: Adapted mesh.

FIGURE 6.10: Example of dynamic projection surface adaptation - Scene 2. Based on occupancy grid information a depth extractor is applied as presented in Section 6.2. The effects of the adaptive mesh can be observed on the gray vehicle, especially on the lower part of the main body and wheels. Without depth information the projection surface is set to its default shape, thus the car body and wheels are projected on the floor. On the contrary, with correct depth information, the walls of the surface approximate well the side of the car, providing a more intuitive representation. Top: default static mesh. Bottom: Adapted mesh.

FIGURE 6.11: Example of dynamic projection surface adaptation - Scene 3. Based on occupancy grid information a depth extractor is applied as presented in Section 6.2. The effects of the adaptive mesh can be observed on the rear parts of both vehicles. Without depth information, the projection surface is set to its default shape, which does not necessarily fit the real scene. Both vehicles are therefore wrongly projected on the floor surface. On the contrary, with correct depth information the walls of the surface approximate well the sides of the cars, providing a more intuitive representation. Top: default static mesh. Bottom: Adapted mesh.

Figures 6.12 and 6.13 show real examples of a parking situation where the dynamic stitching is applied. Comparison of default static seam location with dynamically adapted solution is given that shows the level of enhancement achieved.

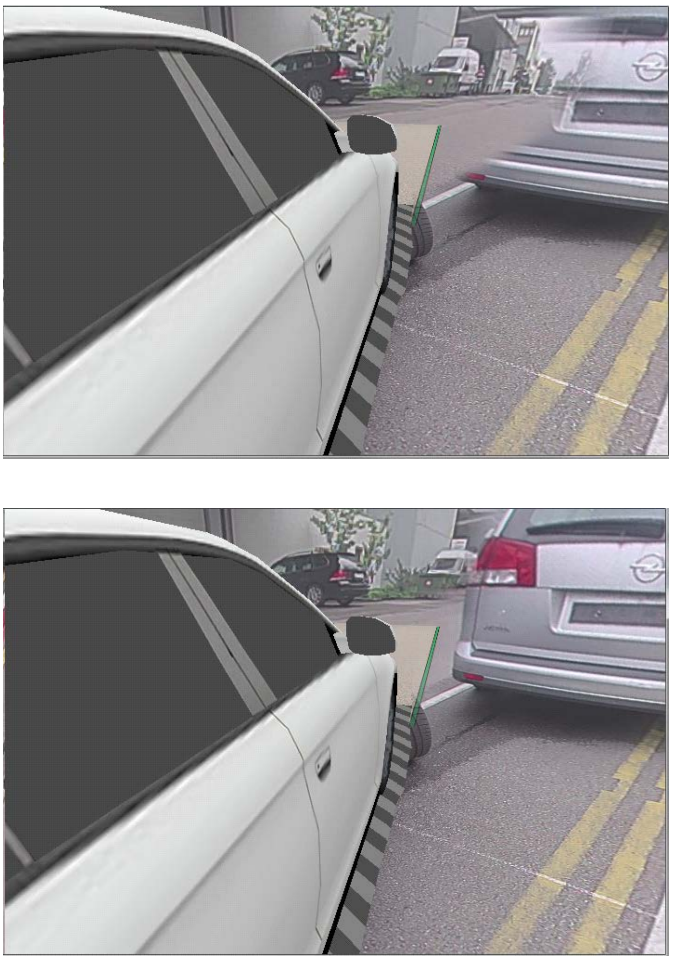


FIGURE 6.12: Dynamic stitching. Based on occupancy grid information, a dynamic stitching scheme has been designed. In areas of the render surface located within the field of view of more than one camera a radial depth extractor function is defined and the optimal seam location is searched such that no close-range elevated object is intersected by it. Top: static default stitching seam. Bottom: dynamic stitching seam.

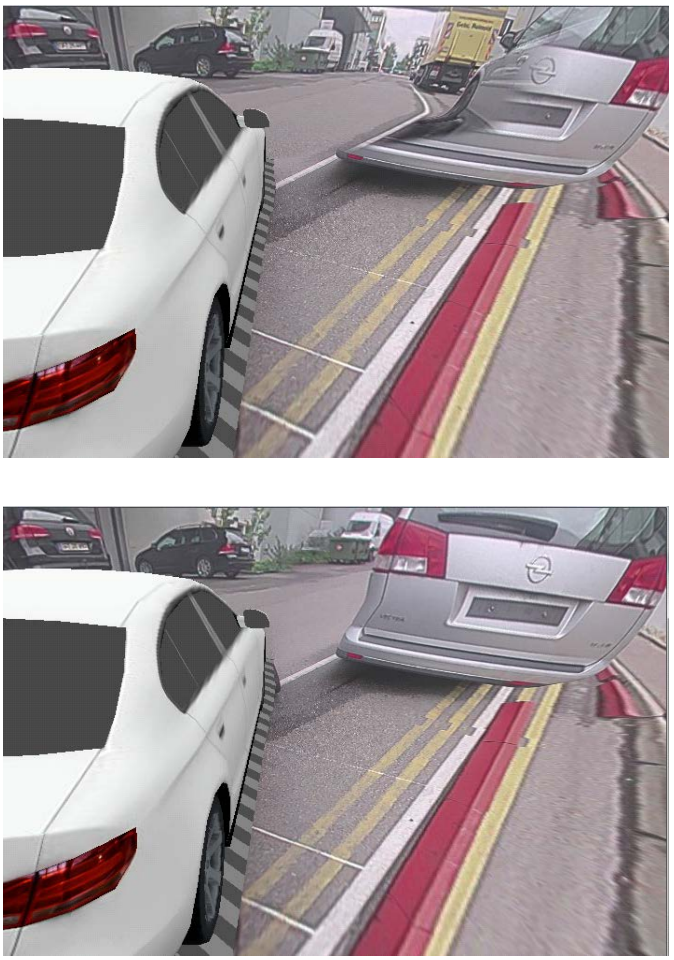


FIGURE 6.13: Dynamic stitching. Based on occupancy grid information, a dynamic stitching scheme has been designed. In areas of the render surface located within the field of view of more than one camera a radial depth extractor function is defined and the optimal seam location is searched such that no close-range elevated object is intersected by it. Top: static default stitching seam. Bottom: dynamic stitching seam.

Figures 6.14 and 6.15 present the results obtained by applying a front inspection view transformation. Distortions on the vertical direction are corrected while keeping the full horizontal field of view. This transformation does not rely on depth information and is statically computed offline, based on intrinsic and extrinsic camera calibration data. Original images are shown for comparison.

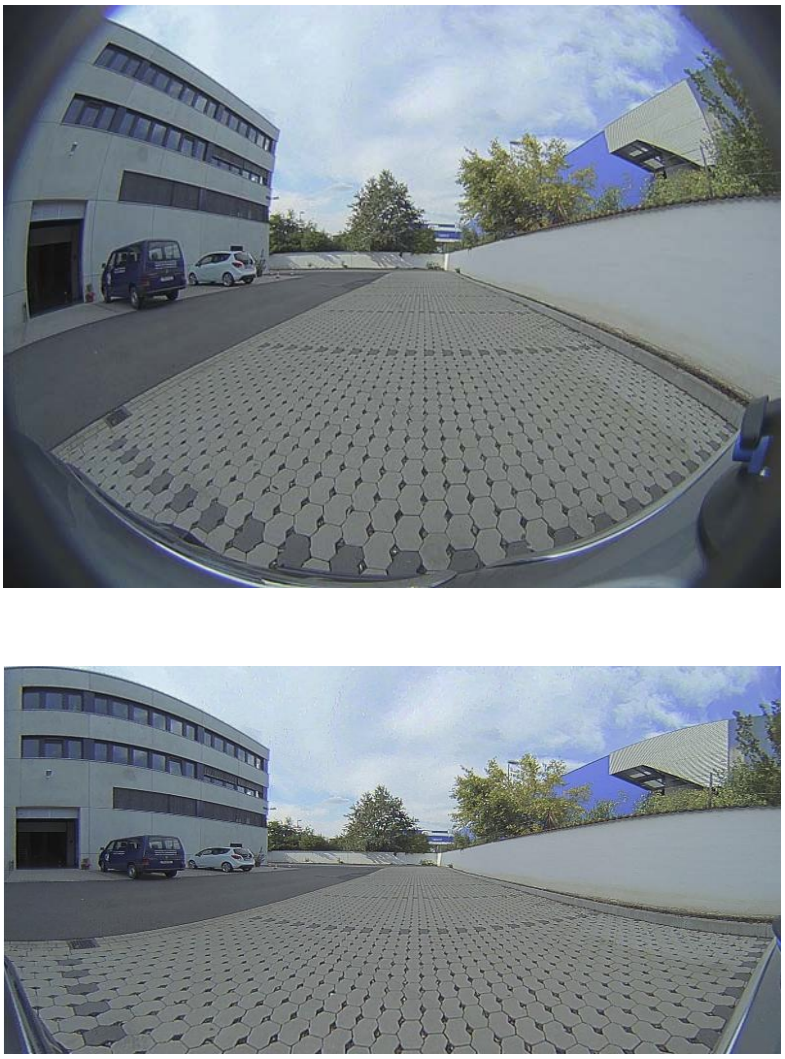


FIGURE 6.14: Example Front Inspection View - Scene 1. Top: original fisheye image. Bottom: Undistorted view. Considering intrinsic and extrinsic camera calibration, no depth information is required in order to correct camera distortion and meet verticality constraints. This is accomplished by means of a cylindrical warping.

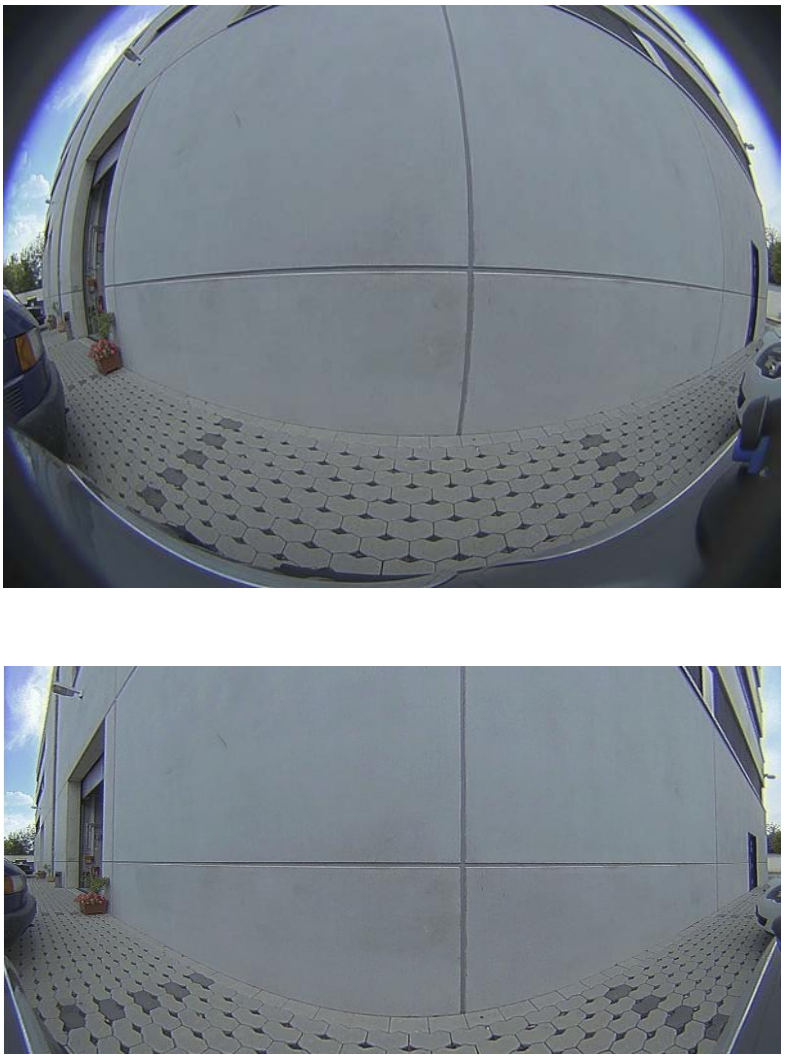


FIGURE 6.15: Example Front Inspection View - Scene 2. Top: original fisheye image. Bottom: Undistorted view. Considering intrinsic and extrinsic camera calibration, no depth information is required in order to correct camera distortion and meet verticality constraints. This is accomplished by means of a cylindrical warping.

Chapter 7

Discussion

In the context of driver assistance, surround view systems have gained popularity in recent years since they can aid the driver in the tasks of parking and maneuvering on a visually intuitive way. Common setups include up to four wide angle cameras that allow for a 360° visualization around the vehicle, where areas out of the line of sight of the driver can observed.

One of the main handicaps encountered in existing systems is the lack of real spatial information on the available visualization. This is a common problem that stems from the imaging process, in which depth information is lost.

The work carried out in this thesis was aimed a studying the possibility to use the surround view cameras in combination with state of the art computer vision algorithms, in order to estimate the spatial information that was lost during the imaging process. The ultimate goal is to use this information to enhance the visualization that can be presented to the driver for aiding with the tasks of parking and maneuvering.

Certain approaches were previously proposed in literature, where mono-camera solutions - like structure from motion - were studied for the described camera configurations. Up to the knowledge of the author there was, however, no previously existing work on real-time stereo vision for the overlapping fields of view of surround view cameras. This has been the main field of research for the work presented in this thesis.

This work includes a detailed analysis of the advantages and disadvantages of applying stereo vision techniques to camera setups where the stereo bases are as large as on a vehicle. In particular, discussion has been presented regarding the

accuracy of time synchronization and of the feature detection. A thorough model has also been proposed to describe the existing overlap on the fields of view of adjacent cameras, based on intrinsic and extrinsic calibration data.

The main reconstruction scheme has been designed based on epipolar rectification, detection and matching of keypoints, and triangulation of correspondences. As a result, robust 3D spatial information was recovered without time accumulation, thus the update rate is given by the frame rate of the imager.

Especially relevant limitations have been detected with respect to the field of view of the proposed system. In particular, existing four-camera setups enable for approximately 90° overlaps only, for each pair of adjacent cameras, thus not allowing for a complete coverage of the surrounding of a vehicle. What is more, due to the large parallax existing between cameras, measurements cannot be conducted on very near objects and narrow drive paths.

To overcome these limitations, new camera configurations have been proposed. In particular, the restricted overlap on the fields of view has been addressed by means of additional cameras mounted on the vehicle. By considering eight instead of four cameras, distributed over the exterior of the vehicle, a coverage of approximately 360° horizontally has been demonstrated and evaluated. In a similar way, additional cameras have been mounted on the roof of the vehicle in order to improve the detection rate on narrow drive paths and parking slots. A new polar-epipolar rectification model has been proposed to deal with the apparent forward motion existing in the last-mentioned camera configuration.

Given the lack of ground truth for the measurements under consideration, a 3D laser range finder has been considered as a reference sensor. The 3D LRF can deliver very accurate distances to reflecting 3D surfaces, with its virtual center of projections as reference. In order to be able to compare the depth measurements obtained by means of stereo with the reference ones, a method has been developed to register the 3D LRF with respect to the multi-camera system. The robustness of the registration process has been evaluated and an error metric has been proposed to evaluate the accuracy of the stereo measurements. In all the experiments, the proposed metric has been considered.

Once the scheme for 3D spatial reconstruction has been proposed and solutions have been evaluated for the lateral limitations, different visualization aspects have been discussed and analyzed with respect to the available depth information. In particular, a novel concept has been proposed to combine occupancy information

with image-based rendering techniques, by means of an occupancy grid and a dynamic rendering surface.

It has been observed that dynamics on the projection surface do not come at no cost, and new image artefacts are introduced by varying the observable density of mesh triangles. A solution has been proposed for this, and the problem of stitching has been discussed, also based on occupancy information.

The output visualization can be delivered to the driver by means of a display, commonly embedded in the control panel inside the vehicle. An overview of the complete system can be seen in Figure 7.1 that includes all stages discussed in this work.

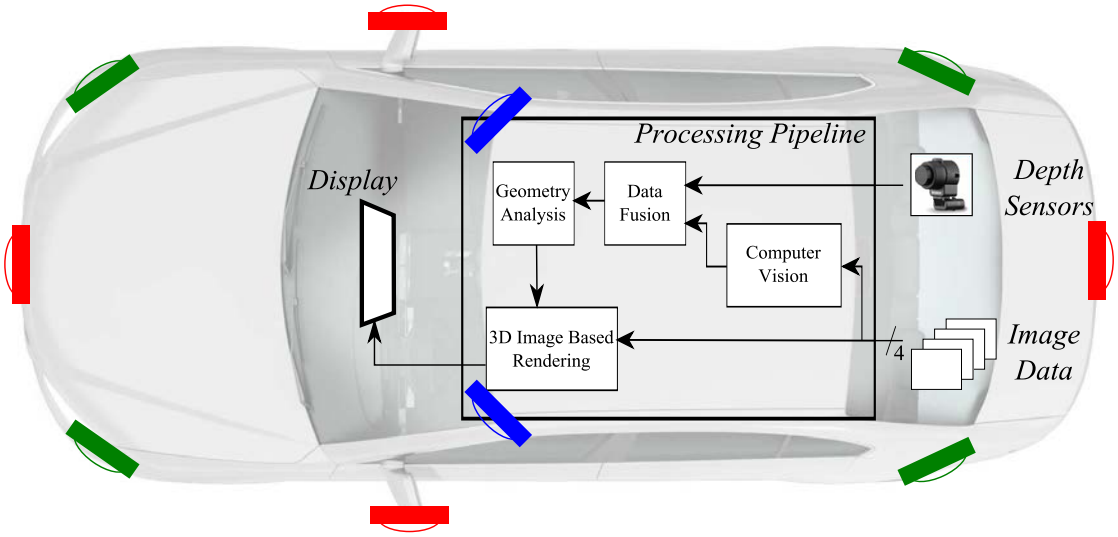


FIGURE 7.1: Depth-based surround view visualization. Depth is obtained by means of Computer Vision and other sensors on a vehicle. Occupancy information is processed and image-based rendering technique are conducted and shown to the driver by means of a display. Different kinds of sensors in different configurations can be considered. For the visualization tasks in this work, a data fusion layer is assumed that makes the depth analysis independent from the type of sensor. The different camera configurations utilized throughout this thesis are also shown. Red: convendtional four-camera setup, used in Section 5.1. Green: additional cameras aimed at improving the surrounding field of view coverage, presented in Section 5.2. Blue: additional upper cameras aimed at performing 3D measurements on narrow drive paths and parking spaces, presented in Section 5.3.

Based on the presented methods and configurations, a complete system can be built. All main aspects, from camera mounting up to rendering and displying have been addressed on this work and the basis for potential new products have been set.

Open points still exist that could not be worked out within this thesis. In particular, initial evaluation of the proposed methods based on proprietary keypoint

disparity estimators has shown room for improvement with respect to the open source functionality considered for this thesis. Furthermore, a real time prototype could not be accomplished yet due to the complexity of such systems in addition to the algorithmical design. Aspects of the system not considered until now would most likely become prominent under real time constraints and would have to be addressed separately as a completely new topic.

In line with the real time prototype, the author sees special interest in conducting a proper user experience study, where physiological conclusions can be obtained with the accuracy and methodology that such a perception-focused project requires.

Chapter 8

Summary

The aim of this thesis was the development of new concepts for environmental 3D reconstruction in automotive surround-view systems, where information of the surroundings of a vehicle can be displayed to a driver for assisting on the tasks of parking and low-speed manouvering.

Different aspects of the system have been addressed, in particular, computer vision techniques have been applied to fisheye images in order to obtain depth measurements. Based on the depth measurements, a geometry analysis has been performed that allows for render surfaces to be dynamically defined. Considering camera extrinsic and intrinsic calibration, image-based rendering has also been conducted.

Restrictions with respect to the system field of view have been identified within the present camera configuration. New camera setups have been proposed to solve these limitations and have been evaluated with respect to a reference lidar sensor. Based on the lidar, a ground truth generation approach has been presented.

Visual aspects of the system have been dedicated a chapter, where enhancements were proposed. These enhancements are aimed at improving the perception of real depth in a comprehensive manner.

Prototypic realization was carried out that shows an approximate measure of the results achieved and prove the feasibility of the proposed concept.

In the experiments carried out throughout this thesis only automotive qualified cameras have been considered in configurations equal or similar to existing products commercially available, thus the adoption of the methods proposed could easily be adopted by the industry. Furthermore, not only visualization systems would benefit from this work, but many other driver assistance functions, too,

since no degradation of other system aspects are introduced and additional accurate depth information is generated based on exsisting sensors.

The density of keypoint correspondences as well as real time implementations remain open points of this work, which could be possible future research lines. Additionally, much work can still be done to understand the real physiological requirements of environmental perception systems in the field of driver assistance.

List of Publications

Conference Papers

Extrinsic Calibration of a 3D Laser Range Finder to a Multi-Camera System
J. Esparza, L. Vepa, M. Helmle, and B. Jähne
9. Workshop Fahrerassistenzsysteme (FAS), March 2014

Extrinsic Calibration of a Fisheye Multi-Camera Setup Using Overlapping Fields of View
M. Knorr, J. Esparza, W. Niehsen, C. Stiller
IEEE Intelligent Vehicles Symposium (IV), June 2014

Wide base stereo with fisheye optics: a robust approach for 3D reconstruction in driving assistance
J. Esparza, M. Helmle, and B. Jähne
German Conference on Pattern Recognition (GCPR), September 2014

Towards Surround Stereo Vision: Analysis of a New Surround View Camera Configuration for Driving Assistance Applications
J. Esparza, M. Helmle and B. Jähne
17th International IEEE Conference on Intelligent Transportation Systems (ITSC), October 2014

Polar Epipolar Rectification for Fisheye Images in Automotive ParkIn Stereo Applications
J. Esparza, M. Helmle and B. Jähne
To be submitted

Patent Applications

Verfahren und Vorrichtung zur dreidimensionalen Abbildung zumindest eines Teilbereichs eines Fahrzeugumfelds
J. Esparza, M. Helmle
Registration ID: DE 102013203404; Application date: February 2013

Verfahren zum räumlichen Darstellen eines Umfelds eines Objekts
J. Esparza, L. Vepa, R. Cano, S. Lang
Registration ID: DE 102013203402
Application date: February 2013

Verfahren zum räumlichen Darstellen des Umfelds eines Objekts
J. Esparza, R. Cano
Registration ID: DE 102013203405
Application date: February 2013

Detektion erhabener Objekte durch Inverse Perspective Mapping in Multikamerasysteme and Erweiterung durch Lichtquellensteuerung.
J. Esparza, L. Vepa, M. Helmle
Registration ID: DE 102013210607
Application date: June 2013

System und Verfahren zum Zusammenfügen mittels mehrerer optischer Sensoren aufgenommener Bilder
J. Esparza, R. Cano
Registration ID: DE 102013211271
Application date: June 2013

Title not public yet
J. Esparza, L. Vepa, M. Helmle
Registration ID: DE 102013224954
Application date: December 2013

Title not public yet
J. Esparza, R. Cano, D. Liepelt, M.J. Esparza, D. Hucker
Registration ID: DE 102014204303
Application date: March 2014

Title not public yet
J. Esparza, R. Cano, D. Liepelt, D. Hucker
Registration ID: DE 102014204652
Application date: March 2014

Title not public yet
J. Esparza, L. Vepa, M. Helmle
Registration ID: DE 102014206677
Application date: April 2014

Title not public yet
J. Esparza, M. Helmle
Registration ID: DE 102014209782
Application date: May 2014

Bibliography

[Abraham and Förstner, 2005] Abraham, S. and Förstner, W. (2005). Fish-eye-stereo calibration and epipolar rectification. *ISPRS Journal of photogrammetry and remote sensing*, 59(5):278–288.

[Alvertos et al., 1989] Alvertos, N., Brzakovic, D., and Gonzalez, R. C. (1989). Camera geometries for image matching in 3-d machine vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 11(9):897–915.

[Bargen and Donnelly, 1998] Bargen, B. and Donnelly, P. (1998). *Inside DirectX: in-depth techniques for developing high-performance multimedia applications*. Microsoft Press.

[Barreto and Araujo, 2001] Barreto, J. P. and Araujo, H. (2001). Issues on the geometry of central catadioptric image formation. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–422. IEEE.

[Beis and Lowe, 1997] Beis, J. S. and Lowe, D. G. (1997). Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 1000–1006. IEEE.

[Blinn, 1989] Blinn, J. (1989). Jim blinn's corner-return of the jaggy (high frequency filtering). *Computer Graphics and Applications, IEEE*, 9(2):82–89.

[Bradski, 2000] Bradski, G. (2000). The opencv library. *Dr. Dobb's Journal of Software Tools*.

[Bresenham, 1965] Bresenham, J. E. (1965). Algorithm for computer control of a digital plotter. *IBM Systems journal*, 4(1):25–30.

[Calonder et al., 2010] Calonder, M., Lepetit, V., Strecha, C., and Fua, P. (2010). BRIEF: Binary Robust Independent Elementary Features. In *Computer Vision–ECCV 2010*, pages 778–792. Springer.

[Dalmia and Trivedi, 1995] Dalmia, A. and Trivedi, M. (1995). Depth extraction using lateral or axial camera motion: an integration of depth from motion and stereo. In *Image Processing, 1995. Proceedings., International Conference on*, volume 1, pages 414–417 vol.1.

[De Villiers et al., 2008] De Villiers, J. P., Leuschner, F. W., and Geldenhuys, R. (2008). Centi-pixel accurate real-time inverse distortion correction. In *International Symposium on Optomechatronic Technologies*, pages 726611–726611. International Society for Optics and Photonics.

[Ehlgen and Pajdla, 2007] Ehlgen, T. and Pajdla, T. (2007). Maneuvering aid for large vehicle using omnidirectional cameras. In *Applications of Computer Vision, 2007. WACV '07. IEEE Workshop on*, pages 17–17.

[Elfes, 1989] Elfes, A. (1989). Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57.

[Esparza et al.,] Esparza, J., Helmle, M., and Jähne, B. Polar Epipolar Rectification for Fisheye Images in Automotive Park-In Stereo Applications. Working Paper.

[Esparza et al., 2014a] Esparza, J., Helmle, M., and Jähne, B. (2014a). Towards Surround Stereo Vision: Analysis of a New Surround View Camera Configuration for Driving Assistance Applications. In *Intelligent Transportation System, 2014. ITSC 2014., IEEE Conference on*. IEEE. Accepted for publication.

[Esparza et al., 2014b] Esparza, J., Helmle, M., and Jähne, B. (2014b). Wide base stereo with fisheye optics: a robust approach for 3D reconstruction in driving assistance. In *Pattern Recognition*. Springer.

[Esparza et al., 2014c] Esparza, J., Vepa, L., Helmle, M., and Jähne, B. (2014c). Extrinsic Calibration of a 3D Laser Range Finder to a Multi-Camera System. In *9. Workshop Fahrerassistenzsysteme*.

[Esquivel et al., 2007] Esquivel, S., Woelk, F., and Koch, R. (2007). Calibration of a multi-camera rig from non-overlapping views. In *Pattern Recognition*, pages 82–91. Springer.

[Fernando et al., 2004] Fernando, R., Haines, E., and Sweeney, T. (2004). *GPU Gems: Programming Techniques, Tips and Tricks for Real-Time Graphics*.

[Furgale et al., 2013] Furgale, P. et al. (2013). Toward Automated Driving in Cities using Close-to-Market Sensors, an Overview of the V-Charge Project. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 809–816, Gold Coast, Australia.

[Gandhi and Trivedi, 2005] Gandhi, T. and Trivedi, M. M. (2005). Dynamic panoramic surround map: motivation and omni video based approach. In *Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on*, pages 61–61. IEEE.

[Gehrig, 2005] Gehrig, S. K. (2005). Large-field-of-view stereo for automotive applications. *Omnivis 2005*, 1.

[Geiger et al., 2012a] Geiger, A., Lenz, P., and Urtasun, R. (2012a). Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3354–3361. IEEE.

[Geiger et al., 2012b] Geiger, A., Moosmann, F., Car, O., and Schuster, B. (2012b). Automatic camera and range sensor calibration using a single shot. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3936–3943. IEEE.

[George and Borouchaki, 1998] George, P.-L. and Borouchaki, H. (1998). Delaunay triangulation and meshing: application to finite elements.

[Geyer and Daniilidis, 2000] Geyer, C. and Daniilidis, K. (2000). A unifying theory for central panoramic systems and practical implications. In *Computer Vision ECCV 2000*, pages 445–461. Springer.

[Harris and Stephens, 1988] Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK.

[Hartley and Zisserman, 2000] Hartley, R. and Zisserman, A. (2000). *Multiple view geometry in computer vision*, volume 2. Cambridge Univ Press.

[Haselich et al., 2012] Haselich, M., Bing, R., and Paulus, D. (2012). Calibration of multiple cameras to a 3D laser range finder. In *Emerging Signal Processing Applications (ESPA), 2012 IEEE International Conference on*, pages 25–28.

[Heckbert, 1989] Heckbert, P. S. (1989). Fundamentals of texture mapping and image warping. Master's thesis, Citeseer.

[Heng et al., 2013] Heng, L., Li, B., and Pollefeys, M. (2013). Camodocal: Automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

[Herrera et al., 2009] Herrera, P. J., Pajares, G., Guijarro, M., Ruz, J. J., Cruz, J. M., and Montes, F. (2009). A featured-based strategy for stereovision matching in sensors with fish-eye lenses for forest environments. *Sensors*, 9(12):9468–9492.

[Hughes et al., 2009] Hughes, C., Glavin, M., Jones, E., and Denny, P. (2009). Wide-angle camera technology for automotive applications: a review. *IET Intelligent Transport Systems*, 3(1):19–31.

[Jähne, 1997] Jähne, B. (1997). *Digital Image Processing: Concept, Algorithms, and Scientific Applications*. Springer.

[Kaempchen et al., 2002] Kaempchen, N., Franke, U., and Ott, R. (2002). Stereo vision based pose estimation of parking lots using 3d vehicle models. In *Intelligent Vehicle Symposium, 2002. IEEE*, volume 2, pages 459–464. IEEE.

[Ke and Sukthankar, 2004] Ke, Y. and Sukthankar, R. (2004). Pca-sift: A more distinctive representation for local image descriptors. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–506. IEEE.

[Kneip and Furgale, 2014] Kneip, L. and Furgale, P. (2014). Opengv: A unified and generalized approach to real-time calibrated geometric vision. In *International Conference on Robotics and Automation (ICRA)*. IEEE.

[Knorr et al., 2014] Knorr, M., Esparza, J., Niehsen, W., and Stiller, C. (2014). Extrinsic calibration of a fisheye multi-camera setup using overlapping fields of view. In *Intelligent Vehicles Symposium (IV)*. IEEE.

[Liu et al., 2008] Liu, Y.-C., Lin, K.-Y., and Chen, Y.-S. (2008). Birds-eye view vision system for vehicle surrounding monitoring. In *Robot Vision*, pages 207–218. Springer.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110.

[Mei and Rives, 2006] Mei, C. and Rives, P. (2006). Calibration between a central catadioptric camera and a laser range finder for robotic applications. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 532–537.

[Mei and Rives, 2007] Mei, C. and Rives, P. (2007). Single view point omnidirectional camera calibration from planar grids. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 3945–3950.

[Mikolajczyk and Schmid, 2004] Mikolajczyk, K. and Schmid, C. (2004). Scale & affine invariant interest point detectors. *International journal of computer vision*, 60(1):63–86.

[Mikolajczyk and Schmid, 2005] Mikolajczyk, K. and Schmid, C. (2005). A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(10):1615–1630.

[Morales and Klette, 2011] Morales, S. and Klette, R. (2011). Ground truth evaluation of stereo algorithms for real world applications. In *Computer Vision–ACCV 2010 Workshops*, pages 152–162. Springer.

[Morel and Yu, 2009] Morel, J.-M. and Yu, G. (2009). Asift: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences*, 2(2):438–469.

[Nene and Nayar, 1997] Nene, S. A. and Nayar, S. K. (1997). A simple algorithm for nearest neighbor search in high dimensions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(9):989–1003.

[Nguyen and Huang, 1992] Nguyen, T. and Huang, T. (1992). Quantization errors in axial motion stereo on rectangular-tessellated image sensors. In *Pattern Recognition, 1992. Vol.I. Conference A: Computer Vision and Applications, Proceedings., 11th IAPR International Conference on*, pages 13–16.

[Oja et al., 1999] Oja, E., Hyvaerinen, A., and Hoyer, P. (1999). Image feature extraction and denoising by sparse coding. *Pattern Analysis & Applications*, 2(2):104–110.

[Okutomi and Kanade, 1993] Okutomi, M. and Kanade, T. (1993). A multiple-baseline stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 15(4):353–363.

[Osfield et al., 2004] Osfield, R., Burns, D., et al. (2004). Open scene graph.

[Pandey et al., 2010] Pandey, G., McBride, J., Savarese, S., and Eustice, R. (2010). Extrinsic calibration of a 3D laser scanner and an omnidirectional camera. In *7th IFAC Symposium on Intelligent Autonomous Vehicles*, volume 7.

[Paul, 2008] Paul, B. (2008). Technical Concepts Orientation, Rotation, Velocity and Acceleration, and the SRM. Technical report, SEDRIS.

[Pless, 2003] Pless, R. (2003). Using many cameras as one. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–587. IEEE.

[Ray, 2002] Ray, S. F. (2002). *Applied photographic optics*, volume 3. Focal Press Oxford.

[Rost et al., 2009] Rost, R. J., Licea-Kane, B. M., Ginsburg, D., Kessenich, J. M., Lichtenbelt, B., Malan, H., and Weiblen, M. (2009). *OpenGL shading language*. Pearson Education.

[Rosten and Drummond, 2005] Rosten, E. and Drummond, T. (2005). Fusing Points and Lines for High Performance Tracking. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1508–1515. IEEE.

[Samet, 1990] Samet, H. (1990). *The design and analysis of spatial data structures*, volume 85. Addison-Wesley Reading, MA.

[Scaramuzza et al., 2007] Scaramuzza, D., Harati, A., and Siegwart, R. (2007). Extrinsic self calibration of a camera and a 3D laser range finder from natural scenes. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 4164–4169.

[Shewchuk, 2002] Shewchuk, J. R. (2002). Delaunay refinement algorithms for triangular mesh generation. *Computational geometry*, 22(1):21–74.

[Shi and Tomasi, 1994] Shi, J. and Tomasi, C. (1994). Good features to track. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on*, pages 593–600. IEEE.

[Shimizu et al., 2010] Shimizu, S., Kawai, J., and Yamada, H. (2010). Wrap-around view system for motor vehicles. *Fujitsu Sci. Tech. J*, 46(1):95–102.

[Snyder, 1997] Snyder, J. P. (1997). *Flattening the earth: two thousand years of map projections*. University of Chicago Press.

[Stein, 2012] Stein, F. (2012). The challenge of putting vision algorithms into a car. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 89–94. IEEE.

[Sturm et al., 2011] Sturm, P., Ramalingam, S., Tardif, J.-P., Gasparini, S., and Barreto, J. (2011). Camera models and fundamental concepts used in geometric computer vision. *Foundations and Trends® in Computer Graphics and Vision*, 6(1–2):1–183.

[Suhr et al., 2007] Suhr, J. K., Bae, K., Kim, J., and Jung, H. G. (2007). Free parking space detection using optical flow-based euclidean 3d reconstruction. In *MVA*, pages 563–566.

[Suhr and Jung, 2014] Suhr, J. K. and Jung, H. G. (2014). Sensor fusion-based vacant parking slot detection and tracking. *Intelligent Transportation Systems, IEEE Transactions on*.

[Suhr et al., 2010] Suhr, J. K., Jung, H. G., Bae, K., and Kim, J. (2010). Automatic free parking space detection by using motion stereo-based 3d reconstruction. *Machine Vision and Applications*, 21(2):163–176.

[Szeliski, 2006] Szeliski, R. (2006). Image alignment and stitching: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 2(1):1–104.

[Thrun, 2003] Thrun, S. (2003). Learning occupancy grid maps with forward sensor models. *Autonomous robots*, 15(2):111–127.

[Tobler, 1973] Tobler, W. (1973). The hyperelliptical and other new pseudo cylindrical equal area map projections. *Journal of Geophysical Research*, 78(11):1753–1759.

[Turkowski, 1990] Turkowski, K. (1990). Filters for Common Resampling Tasks. In *Graphics gems*, pages 147–165. Academic Press Professional, Inc.

[Unger et al., 2014] Unger, C., Wahl, E., and Ilic, S. (2014). Parking assistance using dense motion-stereo: real-time parking slot detection, collision warning and augmented parking. *Machine Vision and Applications*, 25(3):561–581.

[Unnikrishnan and Hebert, 2005] Unnikrishnan, R. and Hebert, M. (2005). Fast Extrinsic Calibration of a Laser Rangefinder to a Camera. Technical Report CMU-RI-TR-05-09, Robotics Institute, Pittsburgh, PA.

[Van Ouwerkerk, 2006] Van Ouwerkerk, J. (2006). Image super-resolution survey. *Image and Vision Computing*, 24(10):1039–1052.

[Wang and Qian, 2010] Wang, R. and Qian, X. (2010). *OpenSceneGraph 3.0: Beginner's Guide*. Packt Publishing Ltd.

[Wang and Qian, 2012] Wang, R. and Qian, X. (2012). *OpenSceneGraph 3 Cookbook*. Packt Publishing Ltd.

[Woo et al., 1999] Woo, M., Neider, J., Davis, T., and Shreiner, D. (1999). *OpenGL programming guide: the official guide to learning OpenGL, version 1.2*. Addison-Wesley Longman Publishing Co., Inc.

[Zhang and Pless, 2004] Zhang, Q. and Pless, R. (2004). Extrinsic calibration of a camera and laser range finder (improves camera calibration). In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2301–2306. IEEE.