Institut für Geodäsie und Geoinformation
Bereich Photogrammetrie

Incremental Map Building with Markov Random Fields and its Evaluation

Inaugural-Dissertation
zur
Erlangung des Grades
Doktor-Ingenieur
(Dr.-Ing.)

der
Landwirtschaftlichen Fakultät
der Rheinischen Friedrich-Wilhelms-Universität
zu Bonn

vorgelegt von:

Maximilian Muffert
aus Lippstadt

Bonn, 2018

Referent:	Prof. Dr.-Ing. Dr. h.c. mult. Wolfgang Förstner
1.Korreferent:	Prof. Dr. Cyrill Stachniss
2.Korreferent:	Prof. Dr. Bastian Leibe
Tag der mündlichen Prüfung:	8. September 2017

Erklärung der Urheberschaft

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Ort, Datum	(Unterschrift)

Zusammenfassung

Inkrementelle Kartenherstellung mit Markov-Zufallsfeldern und dessen Evaluierung

Diese Arbeit präsentiert ein neues Verfahren zur Erstellung von gitterbasierten Belegtheitskarten, das die Abhängigkeit benachbarter Gitterzellen berücksichtigt. Die Belegtheitskarten werden auf Basis einer kompakten Umgebungsdarstellung berechnet, die aus Stereobildern gewonnen wird.

Aktuell ist autonomes Fahren ein grundlegender Arbeitsbereich für Forschungsteams weltweit. Das Ziel dabei ist es, Fahrzeuge intelligenter zu machen, um mehr Sicherheits- und Komfortsysteme in naher Zukunft bereitzustellen. Basierend auf bordeigenen Sensoren müssen autonome Fahrzeuge ihre Umgebung lernen und verstehen, um richtig reagieren zu können. Digitale Karten sind für diese Systeme grundlegend, da diese für die Fahrplanung sowie für eine genaue Selbstlokalisierung des Fahrzeugs verwendet werden. Eine hochmoderne Darstellung von digitalen Karten sind probabilistische Belegtheitskarten, bei denen die Umgebung durch ein regelmäßiges Gitternetz diskretisiert wird. Jede Zelle besitzt eine Wahrscheinlichkeit, ob sie besetzt ist, was eine probabilistische Beschreibung von statischen Hindernissen, Freiraum und unbekannten Bereichen erlaubt. Viele dieser Kartierungsverfahren profitieren von der Annahme, dass die Zellen unabhängig sind. Dies ermöglicht die Umsetzung von effizienten, direkten und inkrementellen Verfahren. Diese Annahme is probabilistisch betrachtet jedoch falsch und führt zu inkonsistenten Karten.

Der Hauptbeitrag dieser Arbeit ist die Entwicklung und Realisierung eines Kartierungsverfahrens, das die Abhängigkeiten benachbarter Zellen berücksichtigt und gleichzeitig ein inkrementelles System mit Echtzeitanforderungen ermöglicht. Ziel ist es, genauere und zuverlässigere Gitterkarten zu erstellen. Darüber hinaus wird auch die Positionsungenauigkeit des Fahrzeugs betrachtet, was zum gleichzeitigen Lokalisierungs- und Kartierungsverfahrens (SLAM) führt.

Das neue Kartierungsverfahren ist als probabilistisches Optimierungsproblem formuliert, bei dem die Karte als ungerichteter Graph interpretiert wird. Um die Abhängigkeiten, und um somit die Korrelationen zwischen benachbarten Gitterzellen zu modellieren, werden Markov-Zufallsfelder (MRFs) verwendet. Um ein effizientes und inkrementales Kartierungsschema zu ermöglichen, werden für jede Zelle die Randwahrscheinlichkeiten geschätzt, welche durch einen schnellen Inferenzalgorithmus basierend auf dem Graph Cut Verfahren realisiert werden. Das SLAM-Problem wird durch einen Rao-Blackwellized Partikelfilter gelöst, der die Kartierung vom Lokalisierungsprozess trennt. Dies ermöglicht, das SLAM-Problem online zu realisieren.

Die Leistungsfähigkeit des neuen Verfahrens wird auf künstlichen und realen Daten ausgewertet. Erkennungsraten sowie die geometrische Genauigkeit von Objekten sind die Grundlagen für die Beurteilung der Qualität der gelernten Karten. Die Leistungsfähigkeit des neuen Ansatzes wird mit den Ergebnissen eines Ansatzes verglichen, der die Abhängigkeiten von Zellen nicht modelliert. Die Ergebnisse zeigen, dass das neue Verfahren eine bessere Leistung hinsichtlich der Erkennungsraten aufweist. Besonders der Freiraum wird präziser, was quantitativ und qualitativ dargestellt wird. Für die Validierung der Leistungsfähigkeit des entwickelten online SLAM-Ansatzes wird die geschätzte Pose des Fahrzeugs berücksichtigt. Der Ansatz ist in der Lage, präzise Positionen mit nur einer geringen Anzahl von Partikeln zu schätzen. Die Grenzen des Kartierungsverfahrens und des SLAM-Ansatzes werden ebenfalls in dieser Arbeit diskutiert.

Summary

Incremental Map Building with Markov Random Fields and its Evaluation

This thesis presents a novel occupancy grid mapping approach which takes the dependencies of neighboring grid cells into account. The grid maps are estimated based on a compact environment representation which is derived from stereo image sequences.

Today, autonomous driving is fundamental work of research teams around the globe. Their aim is to make cars more intelligent in order to provide more safety and comfort systems in the future. Based on on-board sensors autonomous cars must learn and understand their environment to be able to react correctly. Digital maps are essential for such systems since these maps are used for motion planning, and for precise self localization of the ego vehicle. A state-of-the art representation of digital maps are probabilistic occupancy grid maps where the environment is discretized in a regular grid. Each grid cell has a probability that the cell is occupied which allows the description of static obstacles, free space, and unknown areas in a probabilistic way.

This assumption allows the realization of efficient and straight forward incremental occupancy grid mapping approaches. Nevertheless, the assumption of independent grid cells is incorrect in a probabilistic way and leads to inconsistent maps.

The main contribution of this thesis is the development and realization of an occupancy grid mapping approach which keeps the dependencies of neighboring grid cells into account, and simultaneously allows an incremental framework with real time requirements. The aim is to produce more accurate and reliable occupancy grid maps. Furthermore, the pose uncertainty of the ego vehicle is also considered which leads to the simultaneous localization and mapping (SLAM) problem.

The novel mapping algorithm is formulated as a probabilistic optimization problem in which the map is interpreted as an undirected graph. To model the dependencies, in other words the correlation between neighboring grid cells, Markov random fields (MRFs) are applied. To allow an efficient, and incremental mapping scheme, marginal probabilities for each grid cell are estimated, which is realized by a fast inference algorithm based on graph cuts. The mentioned SLAM problem is solved by a Rao-Blackwellized particle filter which separates the mapping step from the localization process. This allows the realization of the SLAM problem in an on-line fashion. For the mapping step the novel approach based on MRFs is chosen, the localization part is realized by a sampling importance resampling (SIR) particle filter.

The performance of the occupancy grid mapping approach is evaluated on the basis of artificial and real-world data. Detection rates as well as the geometrical accuracy of occupied areas are the foundations of assessing the quality of the learned maps. The performance of the novel approach is compared against the results of an approach which does not model the dependencies of grid cells. The results show that the novel approach has a better performance with regard to the detection rates. Especially free space areas are more precise which is shown in a quantitative and qualitative way. For the validation of the performance of the developed on-line SLAM approach the estimated pose of the ego vehicle is taken into account. It is shown that the approach is able to estimate precise positions using only a small number of particles. The limits of the mapping algorithm, and of the SLAM approach are also discussed in this thesis.

x

Contents

List of Figures

List of Tables

List of Algorithms

Notation

Mathematical Notation

Symbol	Description
x, y, z, ϕ	scalar values
$\mathbf{x}$	homogeneous vector
$\boldsymbol{x}$	Euclidean vector
$x_{i;j}$	i-th element of vector $\boldsymbol{x}$ with label j
X	homogeneous matrix
X	Euclidean matrix
X^{T}	transpose of matrix X
X^{-1}	inverse of matrix X
I_n	identity matrix of dimension n
$\mathbf{0}_n$	zero vector of dimension n
${}^{w}\Delta\mathsf{M}_{t}^{t-1}({}^{w}\boldsymbol{x}_t)$	motion referred to coordinate system w between time step t and $t-1$ defined by control elements ${}^{w}\boldsymbol{x}_t$
$\mathcal{S} = \{..., \mathcal{S}_j, ...\}$	set with sub-set $\mathcal{S}_j$, $j = 1, ..., J$
$\mathcal{S}_j = \{..., \boldsymbol{x}_{ij}, ...\}$	sub-set with vector elements $\boldsymbol{x}_i$, $i = 1, ..., I_j$
$\mathcal{Z} = (..., \boldsymbol{z}_h, ...)$	sorted list of vector elements $\boldsymbol{z}_h$
$x^{(i)}$	i-th sample or particle of the true stochastical variable $\underline{x}$
$\mathcal{N}(\mu, \sigma^2)$	normal distribution governed by mean μ and variance σ^2
$\mathcal{N}(d; \mu, \sigma^2)$	normal distribution governed by mean μ and variance σ^2 evaluated at value d

Acronyms

Acronym	Description
ADAS	advanced driver assistance systems
BA	bundle adjustment
CNN	convolutional neural network
CRF	conditional random field
DARPA	the defense advanced research agency
FPGA	field-programmable gate array
GPS	global positioning system
GN	Gauss-Newton
ICP	iterative closest point
IMU	inertial measurement unit
LM	Levenberg-Marquardt
LIDAR	light detection and ranging
MRF	Markov random field
MAP	maximum-a-posteriori
MCL	Monte Carlo localization
NLS	nonlinear least squares
PF	particle filter
RADAR	radio detection and ranging
RBPF	Rao-Blackwellized particle filter
SGM	semi global matching
SIR	sampling importance resampling
SLAM	simultaneous localization and mapping

Chapter 1

Introduction

1.1 Motivation

Autonomous driving has grown rapidly in the last couple of years and is a fundamental work of research teams around the whole globe. Not only traditional car manufacturers like BMW, Mercedes-Benz or Toyota are active in this field, but also software giants like Apple, Baidu, Google or Uber have an immense interest in this market. The aim is to make vehicles step by step more *intelligent* to take over reliably the driver's tasks up to the time at which the vehicle has the full responsibility about action and reaction.

On the one hand, **safeness** is one major advantage of *intelligent cars*. Safety systems like emergency braking, automated distance control or lane keeping assistance are already available in mid-size and luxury cars these days to support the driver in complex traffic situations and to reduce the risk of accidents. On the other hand, **comfort** is also an advantage of intelligent cars. In future, chauffeur-driven cars will allow the driver to be productive during commute time to legally read newspapers, answer mails or just relax. Autopilot modes will drop off passengers at their desired destinations like airports or rail stations and then the car will park on its own. It is to be expected that these systems will be released into the market in the next 10 years. However, current parking assistance systems which detect parking lots correctly and then maneuver fully automated into them demonstrate how intelligent are our cars already today.

Cars with such systems must learn and understand their environment to be able to act or react correctly in situations like the above mentioned parking scenario. The environment information can be broken down into (1) *dynamic objects* like other road users or pedestrians, (2) *static information* like contours of buildings or curbs, and (3) possible drivable *free space*. On board sensors like radar, laser scanners or cameras are used to learn and detect these types of information on-line. Figure 1.1 gives an example how a typical urban traffic scenario is classified in these classes.

The learning of static environment information as well as the estimation of reliable free space is also known as digital *map building* in mobile robotics in which sensor readings are integrated into a specific map representation. These maps are essential for current safety and comfort systems as well as for autonomous driving applications: they help during vehicle path planning and by interacting with other traffic participants. They are also used to localize the vehicle relative to the created map.

Figure 1.1: The classification of an urban traffic scenario into dynamic objects (red), static information (blue) and free space (green) which also includes the sidewalk. The detection of these three types is mandatory for advanced driver assistance systems and autonomous driving applications [Ziegler et al., 2014].

In this thesis, the key focus is on probabilistic on-line map building using a compact environment representation as input data which is created from stereo camera image sequences. The generated map is defined as a grid and represents static information as well as free space. The key aspect is that we consider dependencies between neighboring grid cells in a probabilistic fashion. The uncertainty of the vehicle's pose is also taken into account in this thesis. We also concentrate on the comprehensive evaluation of the new mapping technique.

1.2 Related Work

In this section we describe related work which motivated, inspired and influenced the current thesis. First we focus on advanced driver assistance systems (ADAS) and autonomous driving applications (Sec. 1.2.1 and Sec. 1.2.2). In Sec. 1.2.4 we describe map building approaches which also includes a discussion of mapping with know poses (Sec. 1.2.5) as well as mapping with uncertain poses (Sec. 1.2.6). At the end, we give a brief overview of scene understanding with stereo vision (Sec. 1.2.7).

1.2.1 Driver Assistance Systems

In 2015, slightly over 26 000 people were killed by road accidents in the European Union (EU), estimated by the statistical office of the EU (EUROSTAT). In spite of the large number of deaths, the amount of killings is fewer than half as many as 20 years ago, which was also stated in the study. Today's standard safety systems like airbags, anti-lock braking systems (ABS) and electronic stability program (ESP) have contributed considerably to the decrease of the number of killings during this time period.

The "next generation" of safety and comfort systems, also known as ADAS, conquer successfully the automotive market in the last couple of years to reduce the risk of accidents even before they occur. Since systems like ABS and ESP are based only on the vehicle dynamics, ADAS require environment perception and the interaction with other participants which make them more complex in terms of software and hardware requirements.

A good example is adaptive cruise control (ACC) which automatically regulates the distance to leading vehicles using radio detection and ranging (RADAR), or light detection and ranging (LIDAR) to reduce the number of rear-end collisions with high speed (BMW; Mercedes-Benz; VOLKSWAGEN). These systems are mostly coupled with automated "pre-crash" braking systems.

In 2016, Mercedes-Benz presented its new E-class with steering pilot and active lane change assists at the north America international auto show (NAIAS) which helps to keep the car autonomously in the driving lane, and also helps the driver to make a safe semi-automatic lane change. It already allows the driver to keep the hands off the steering wheel for a while. For these complex systems, RADAR and stereo camera information is used. Similar systems are also available from TOYOTA or BMW. Blind spot assists which supervise areas the driver cannot see, pedestrian detection systems and self-parking assists are examples for ADAS which are also available in the automobile mid-range segment.

Bengler et al. [2014] give a comprehensive overview of ADAS and how these systems evolved over the last three decades: research fosters the development of already existing and future systems and the way from prototypes to a final product in the automotive market takes years or even decades. ADAS must pass strong quality and safety standards and need to be fully transparent. As mentioned above and also stated in [Bengler et al., 2014], the difference between autonomous driving applications and the functionality of current ADAS is vanishing in these days. Because of this fact, we present an overview of autonomous driving applications in the following.

1.2.2 Autonomous Driving Applications

Fully autonomous driving means that a vehicle is able to manage all possible traffic scenarios without any interaction with the driver. Research groups and car manufactures are working with great enthusiasm and passion to realize this researcher's dream of a self driving car. In the last two decades autonomous driving has grown rapidly, from initial prototypes towards highly embedded software systems, which will be the new generation of ADAS.

Between 1986 and 1995, European universities and car manufactures worked closely together in the PROMETHEUS project ("program for an European traffic of highest efficiently and unprecedented safty") to realize the first autonomous drives in Europe. Franke et al. [1994] contributed to this project in a way that image processing was used the first time to control the lateral position of the car relatively to the lane markings.

An important impact for autonomous driving were the DARPA (the defense advanced research agency) challenges, held in 2004, 2005 and 2007. In 2005, the robot Stanley [Thrun et al., 2006] won the challenge using machine learning and probabilistic reasoning. Stanley was equipped with LIDARs, RADARs, cameras, inertial measurement units (IMUs) and global positioning system (GPS). In 2007, the first urban DARPA challenge was held in Victorville, CA, where the cars had to interact with other participants the first time. As an example, the team AnnieWay [Kammel et al., 2008] had successfully entered the DARPA finals. The car was able to pass parked cars,

performing u-turns, and merge into traffic again. For the environment perception, a roof mounted 360^o laser scanner was used. The range measurements were mainly used to reconstruct the scene geometry, whereas the reflectivity values of the laser helped to detect the lane markings on the ground.

In 2010, Google achieved high publicity with their impressive autonomous vehicles which recorded and logged over 140 000 miles at that time. The tech company employed software developers and researchers which participated in the DARPA challenges already. Google benefits from their experiences even today. One of Google's key success is their highly detailed digital maps and their knowledge how to handle enormous amounts of data. In 2013, Alberto Broggi and his team [VisLab, 2013] also presented impressive autonomous driving in urban scenarios around Parma, Italy.

In 2014, the Mercedes-Benz S 500 Intelligent Drive [Ziegler et al., 2014] followed the historic Bertha Benz Memorial Route fully autonomously. The total length of the route was 103 km long and passed villages and major cities like Mannheim and Heidelberg. Compared to previous autonomous driving projects, e.g. [Kammel et al., 2008; Google, 2010; VisLab, 2013], close to production sensor hardware, like RADAR and stereo camera systems were used during this drive. Similar to all previously mentioned autonomous driving applications, the Mercedes-Benz team also relied on accurate and detailed digital maps. However, the maps were generated off-line in a semi-automated way which is not scalable and far away from a commercial roll-out.

Aeberhard et al. [2015] presented their lessons learned of autonomous driving on highways since 2011. One of their major problems also deals with digital maps: similar to Ziegler et al. [2014] the process of remapping must be improved to keep the maps up-to-date. Furthermore, the sensor range and capability should also be improved to allow autonomous driving up to a speed of 150 km/h. Therefore, Aeberhard et al. [2015] suggest the use of sensor redundancy, more stable sensors, and also allow a possible car-to-X communication.

In December 2016, the Google self driving car project became WAYMO. At this time the company reached the 2 000 000 autonomous miles border. A controversial discussion started after the company stated that their final product will neither have a steering wheel or floor pedals. However, it can be assumed that WAYMO will release their product in the next years and it seems that the dream of a complete autonomous vehicle will become true.

1.2.3 Key Components for Autonomous Driving

Autonomous driving systems [Google, 2010; VisLab, 2013; Ziegler et al., 2014] rely in general on six key components which are introduced shortly. Figure 1.2 also visualizes the named components and shows the interaction between them.

Sensors. Sensors like RADAR, LIDAR, or cameras are essential to allow self driving cars to "see". Sensors like GPS and IMUs are also important for (broad) localization and to measure vehicle dynamics. Sensors are also indispensable for map learning techniques.

Digital Maps. Detailed digital maps includes all important static information of the environment, like road and lane geometries, curbs, and traffic islands, to allow correct and precise motion

planning. Digital maps can also include static information for precise localization and, in most of the cases, potential free space. In this context, free space does not take dynamic objects into account.

Environment Perception and Object Detection. Environment perception and object detection is essential to handle complex traffic scenarios. Based on the sensor readings, the self driving car knows what it sees in its surroundings. This also includes the detection of dynamic objects like vehicle detection, pedestrian detection. The detection of traffic lights and speed limits is also desirable.

Localization. Precise localization answers the question where the car is in a given map with a position accuracy of 20 cm or better and a heading accuracy of 0.2 deg or better. One possibility to self localize the robot is to match sensor readings with the digital map. Deeply coupled filter systems, like differential GPS in combination with IMUs, can also be used for this task.

Motion and Trajectory Planning. If the car knows where it is (self localization) and also knows its dynamic and static environment (environment perception and object detection) the computation of the desired future trajectory to a designated destination as a function of time is possible. The trajectory planner produces the input for the vehicle control unit and how the car should move.

Vehicle Control and Reaction. The control unit smoothly guides the car along the planned trajectory. A separation into a longitudinal and lateral control component is common practice.

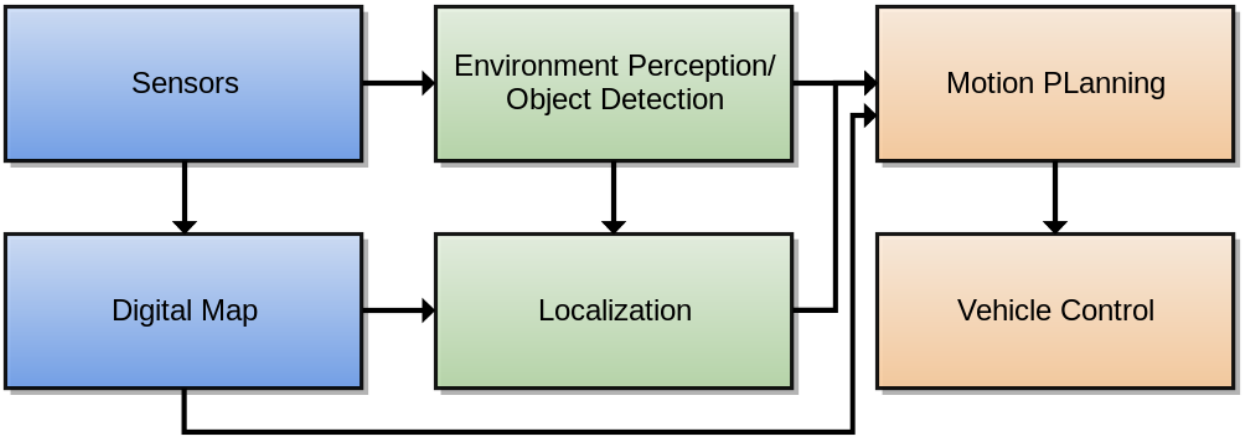


Figure 1.2: Key components of autonomous driving applications and how they interact with each other. The arrows indicate information flow. As one can see, digital maps play an important roll for the localization and motion planning task.

This thesis focuses on the creation of digital maps using stereo camera image sequences from an earth bound moving platform. The created maps represent static environment information as well as free space information which can be used for autonomous driving applications. In this case, dynamic objects are excluded from free space. Grid based map approaches have proven useful in the field of robotics and autonomous driving applications as stated above. Therefore, we use the probabilistic occupancy grid map representation in this thesis.

1.2.4 Digital Maps

The most common representations of digital maps are feature based maps, topological/geometrical maps and grid based maps. In all these cases maps are learned from the robot's sensor readings. The task of on-line map creation with the robot sensors is also known as *mobile mapping*.

Feature based Maps. Feature based maps contain features, also called landmarks, of the environment which are extracted from sensor readings. In general, these features have a global position with reference to the map's coordinate system. Each feature can also carry semantic information in order to make them more unique: features can be classified as objects, like trees, street lamps, traffic signs or house corners. If cameras are used features can also be characterized by e.g. SIFT [Lowe, 2004] or HoG [Dalal and Triggs, 2005] descriptors. Feature based maps are very suitable for on-line localization as stated in [Ziegler et al., 2014; Lategahn and Stiller, 2014]. However, feature based maps do not include dense information about free space and static information by default. Furthermore, feature based maps which include abstract descriptors are suffering from the fact that they are sensor type specific.

Geometrical Maps. Geometrical maps include entities like road markings, street boundaries or shape of buildings and are more comprehensive than today's navigation maps [Ziegler et al., 2014]. Based on raw sensor readings, these geometrical entities can be extracted fully- or semi-automatically [Ziegler et al., 2014]. They can also include topological information, like the connectivity between road segments and provide information for path planning. Additional information like speed limits or school zones are also often stored in these maps. Geometrical maps are particularly suitable for motion and path planning [Ziegler et al., 2014]. Furthermore, they can also be used for localization purposes, as described in [Schreiber et al., 2013; Rabe et al., 2016].

Grid based Maps. By using grid based maps, the continuous environment is discretized in a consistent grid structure. Each map element is called a grid cell which includes information about its environment, for example if the cell is covered by an obstacle or not. Grid based maps are usually in 2D using the planar ground plane assumption. Nevertheless, grid based maps can also be used to represent the full 3D space. If this is this case, a single grid element is called a voxel. Grid based maps are applicable in a variety of ways. For example, occupancy grid maps (see Sec. 2.5) are predestined to represent free space, occupied areas as well as unknown regions in a probabilistic way [Moravec and Elfes, 1985; Elfes, 1989; Thrun, 2003]. Another example are digital elevation grid maps [Kweon and Kanade, 1992; Lacroix et al., 2002] which are used to model the height

information of the ground surface. Grid based maps benefit from the fact, that raw sensor readings are integrated into the grids without any predefined object assumptions, like it is the case by the generation of feature and geometrical maps. The sensor specific noise and sensor failure behavior can be considered directly. Based on this fact, grid based approaches are especially suitable for low level data fusion [Matthies and Elfes, 1988; Munz et al., 2010], because they do not rely on any object extraction or specific feature definition.

As stated in Stachniss [2006, Chapter 2.2], grid based maps are important for robot exploration purposes because they can represent unknown areas. Grid based maps are also suitable for motion planning [Elfes, 1989; Torabi et al., 2007; Čikeš et al., 2011] and for self-localization purposes [Levinson and Thrun, 2010; Roewekaemper et al., 2012; Rapp et al., 2016].

Besides these benefits, two major drawbacks occur if the environment is represented by grid maps. First, their memory requirements are huge, especially for large scale environments which are represented in the full three dimensional space. To handle this burden,e.g. [Hornung et al., 2013] created an efficient level-of-detail data structure by using an octree map compression method, the so-called OctoMap. The map only provides high spacial resolution if really needed which keeps the overall memory low.

Because of the map's finite grid resolution, the second notable drawback are discretization errors which occur during the mapping of sensor readings into the discrete map structure. The resolution of a single grid cell strongly depends on the desired application: the user has to answers questions like how much memory is available, and is the map creation an on-line or off-line process.

1.2.5 Probabilistic Grid Mapping with Known Poses

Moravec and Elfes [1985] introduced occupancy grid maps at first using wide angle sonar sensors under the assumption that the pose of the robot is known. Each grid cell has an occupancy probability, which means how likely it is that the cell represents static obstacles or not. Free space is also represented in these maps under the assumption, that sensor rays first pass free space area before they hit an obstacle.

Areas which are not influenced by sensor readings are defined as unknown areas. Besides the use of sonar sensors [Moravec and Elfes, 1985; Thrun, 2003], the occupancy grid mapping idea works also well for LIDAR [Limketkai et al., 2002; Yguel et al., 2006; Schmid et al., 2010], RADAR [Homm et al., 2010; Werber et al., 2013], and stereo vision [Moravec, 1996; Badino et al., 2007; Perrollaz et al., 2010; Lategahn et al., 2011; Muffert et al., 2014].

The Paradigm of Independent Grid Cells Thrun et al. [2005] give a detailed description of 2D occupancy grid mapping techniques. As stated in Thrun et al. [2005, Chapter 9.2], the estimation of the full Bayesian map posterior is intractable, because of the enormous amount of possible maps which can be defined over a grid. Therefore, the standard occupancy grid mapping approach is based on a binary (the grid cell is occupied or not) Bayes filter with static state (Thrun et al. [2005, Chapter 4.2]) which also assumes that all grid cells are independent. This allows a fast and incremental estimation of maps which is necessarily for on-line mapping applications [Moravec and Elfes, 1985; Grisetti et al., 2007; Yguel et al., 2006; Muffert et al., 2014].

However, the decomposition into independent grid cells is a strong assumption which is incorrect from a probabilistic point of view. All named approaches from above suffer from this assumption and inconsistencies in the map can occur. There are only a few approaches who deal with this issue and discuss this topic in detail [Thrun, 2003; Merali and Barfoot, 2012, 2013; Dhiman et al., 2014].

Thrun [2003] presented a maximum a posteriori (MAP) solution using the expectation maximization (EM) algorithm which maintain dependencies between neighboring grid cells. This results in more consistent maps than using the standard occupancy grid mapping approach (Thrun et al. [2005, Chapter 4.2]). However, the MAP solution needs all available sensor readings and is consequently a batch approach. Furthermore, the MAP solution does not include the uncertainties of the grid cells by default. Based on these facts, this batch approach is not suitable for on-line approaches.

Merali and Barfoot [2012] introduced the *patch map* which also takes into account that neighboring cells are dependent. The patch map algorithm solves the full Bayesian posterior map solution only for a small area, the patch map, and then iterates over the possible patches.

A drawback of this approach is that it needs a MAP or ground truth (GT) solution of the map to determine the first occupied cell and is not real time capable. Similar to [Thrun, 2003], this approach is also a batch solution. The authors stated, that their approach lies between the full Bayesian solution and the standard occupancy grid mapping approach and is suitable for benchmarking. Based on their previous work, Merali and Barfoot [2013] introduced an occupancy grid mapping approach with Markov chain Monte Carlo (MCMC) Gibbs sampling which allows to sample from the full Bayesian posterior. This also allows to estimate occupancy probabilities for the cells by drawing many samples. Compared to the patch map solution this method does not need any MAP or ground truth solution for the initialization step. The authors stated that it is still open which of these two algorithms performs better. Dhiman et al. [2014] propose an occupancy grid mapping approach using higher order factor graphs to handle dependencies between neighboring grid cells. Belief propagation is used for inference estimation. It turned out that their approach results in more accurate maps than using standard occupancy grid mapping approaches. However, it is still unclear how efficient the algorithm is in terms of computation time.

The research group around F. Ramos also model the dependencies between nearby locations in grid maps [O'Callaghan and Ramos, 2012, 2014; Senanayake et al., 2016]. They use Gaussian process occupancy grid mapping to overcome the disadvantages of the general occupancy grid mapping approach, namely the discretization of the environment and the independence of grid cells. Gaussian process occupancy grid mapping allows a continuous probabilistic representation. Nevertheless, the use of Gaussian process in occupancy grid mapping is expensive and therefore no real-time capable.

The Handling of Dynamic Obstacles. The standard occupancy grid mapping approach assumes that all observed occupied areas are static environment which is not true for most applications. Therefore, many research work deals with the issue how to handle dynamic obstacles like walking pedestrians or cars in occupancy grid maps. Biswas et al. [2002] did pioneer work in the field of detecting dynamic obstacles in occupancy grid maps. They used map differences to detect changes in the environment and also applied an EM algorithm to learn dynamic object models and its location in the grid maps. In the same year, Hähnel et al. [2002] presented an approach which detects and tracks multiple people in the environment. The results were incorporated into

the map building process to reduce the number of dynamic objects in the resulting maps. Stachniss and Burgard [2005] clustered local grid maps to detect spurious measurements which represents dynamic objects. These clusters were also used to localize the robot in dynamic environments using particle filters. Brechtel et al. [2010] used a particle filter like approach for grid based occupancy tracking with LIDAR measurements. Because of using importance sampling the approach is real-time capable. Nuss et al. [2015] presented a sequential Monte Carlo Bayesian occupancy filter (SMC-BOF) to represent and track dynamic obstacles in grids. Their main contribution compared to previous work (e.g. Brechtel et al. [2010]; Danescu et al. [2011]) is that they fuse LIDAR and RADAR measurements together. The Doppler effect of the RADAR improves the estimation of dynamic obstacles significantly.

Another possibility is to extract dynamic obstacles from the raw sensor data before measurements are integrated into grid maps. As an example, Muffert et al. [2013] used stereo vision in combination with semantic labeling to exclude dynamic obstacles before data is integrated into occupancy grid maps. This results in occupancy grid maps with only static environment information which is more reliable for localization and motion planing purposes.

1.2.6 Probabilistic Mapping with Unknown Poses

In the previous section we presented approaches which assume that the pose of the moving platform is given, and is consequently correct. For small environments, and for vehicle centered grid maps (Nuss et al. [2015]; Muffert et al. [2014]) this assumption is acceptable as long as sensor based ego motion estimation or the car's odometry information is precise enough. But, nevertheless, to map large scale environments and to create precise global occupancy grid maps, this assumption is untenable. In robotics, this leads to the well-known simultaneous localization and mapping (SLAM) issue where both, the pose of the vehicle and the map are estimated at the same time. Since there exists an unbelievably large number of SLAM approaches in literature, we only focus on work which motivated us to solve the SLAM issue during this thesis [Smith and Cheeseman, 1986; Durrant-Whyte and Bailey, 2006; Murphy, 1999; Montemerlo et al., 2002; Hähnel et al., 2003; Thrun et al., 2005; Grisetti et al., 2007; Choi, 2014].

A good overview of fundamental SLAM techniques can be found in [Thrun et al., 2005, Chapter 10-13]. The authors state that the SLAM problem can be traced back to geodetic surveying tasks and can be formulated as a nonlinear least squares (NLS) problem where all unknowns are estimated *en bloc*. In SLAM, the unknowns are defined by the landmarks of the map and the poses of the vehicle. The observations are the sensor readings and the control information of the vehicle. The dependencies between observations and unknowns can be represented in a graph structure in an intuitive way [Thrun et al., 2005, Chapter 11, p.338]. To solve the SLAM problem *en bloc* via NLS, state-of-the-art methods like Gauss-Newton (GN) or Levenberg-Marquardt (LM) are used in the research community. It is also known as the Graph SLAM or Full SLAM problem. Open source libraries like the *g2o* framework [Kümmerle et al., 2011] or the *ceres* solver [Sameer Agarwal, 2016] provide such solvers and have proven their worth in autonomous driving applications [Ziegler et al., 2014; Kerl et al., 2013].

To solve the on-line SLAM problem extended Kalman filters (EKFs) are used successfully in 2D scenarios ([Smith and Cheeseman, 1986; Durrant-Whyte and Bailey, 2006; Thrun et al., 2005]). Here, maps are represented as feature based maps and the algorithm works well, if the number of

features is small and correspondences between map features and sensor readings are known [Thrun et al., 2005, Chapter 10.2.1].

Next to EKFs, particle filters (PFs) are also very suitable to solve the 2D SLAM problem. PFs are in general easy to implement, and are non-parametric which allows multi-modal solutions compared to EKFs. However, the number of particles would explode if the SLAM problem would be implemented without any conditional assumptions about the map and the pose of the vehicle. Therefore, Murphy and Doucet et al. applied the well-known Rao-Blackwellized particle filters (RBPFs) in the SLAM problem. Here, RBPFs separate the estimation of the pose from the estimation of the map which reduce the number of particles enormously. The RBPF approach works well with feature-based maps [Montemerlo et al., 2002] as well as with grid based maps [Hähnel et al., 2003; Grisetti et al., 2007; Choi, 2014]. An insight of RBPFs is given in Sec. 2.7.2.

In the research field of photogrammetry and computer vision the SLAM problem also occur during the 3D reconstruction of environments with image sequences [Frahm et al., 2010]. Here, it is better known as bundle adjustment (BA), where the unknown state vector is defined by the relative 3D motion and the intrinsic parameters of the camera(s), and the 3D object points of the environment. The minimization of the re-projection error is formulated as a NLS problem. Therefore, the frameworks of Kümmerle et al. [2011] and Sameer Agarwal [2016] are also very suitable for bundle adjustment tasks. Comprehensive insights in the field of bundle adjustment are given in [Hartley and Zisserman, 2004; McGlone et al., 2004; Förstner and Wrobel, 2016].

1.2.7 Scene Understanding with Stereo Vision

Next to LIDAR and RADAR approaches ([Yguel et al., 2006; Hermes et al., 2010; Homm et al., 2010; Nuss et al., 2015]), vision has proven a powerful solution for urban traffic scene understanding (Ess et al. [2009]; Hermes et al. [2010]; Badino et al. [2009]; Pfeiffer and Franke [2011]; Erbs et al. [2012]; Scharwächter and Franke [2015]; Schneider et al. [2016]; Cordts et al. [2016]). Especially in the last three years, caused by the emergence of convolutional neural networks (CNNS), single frame image classification and scene understanding improved a lot. Fundamental work was done by Shelhamer et al. [2016] where CNNs where used for semantic scene labeling the first time. As stated in [Cordts et al., 2016], a fully-convolutional network [Shelhamer et al., 2016] reaches the semantic labeling on the KITTI benchmark [Geiger et al., 2012] and on the CamVid database [Brostow et al., 2009] easily. Many work rely on open source implementations [Girshick, 2015]. Before CNNs came up, state-of-the art image understanding was mainly driven by Markov random fields/conditional random fields (MRFs/CRFs) [Lafferty et al., 2001; Kohli and Torr, 2007; Erbs et al., 2012] or by random decision forest [Scharwächter and Franke, 2015]. The work of Chen et al. [2014] and Schwing and Urtasun [2015] combine the strengths of CRFs and CNNs.

Another research branch in image based scene understanding is the estimation of 3D compact scene representations based on dense disparity images. Here, we focus on the *Stixel World* idea ([Badino et al., 2009; Pfeiffer and Franke, 2011; Benenson et al., 2012]) which is useful for autonomous driving applications [Ziegler et al., 2014] or occupancy grid mapping [Muffert et al., 2013, 2014]. It is also possible to estimate Stixels without computing disparity maps with very high frequency [Benenson et al., 2012]. Schneider et al. [2016] present a neat way to combine both pixel-level semantic labeling via CNNs and depth information to generate a detailed 3D scene representation which is called the semantic Stixel World.

1.3 Contribution of this Thesis

With the background of related work, our contribution is a novel 2D on-line feasible grid based mapping approach to represent free space and static obstacles in a probabilistic way. Free space means in this case areas where the vehicle can physically drive without causing major damages. This can also include clear sidewalks. Static obstacles are structured environments which must be avoided by the vehicle. Dynamic obstacles are excluded in our grid maps.

The main achievement of this thesis is, that we model the dependencies between neighboring grid cells to achieve more accurate and more robust free space and static obstacles. From our point of view and as stated in Sec. 1.2.5, this research topic is still underrepresented and should get greater attention. As input data we rely on the use of the compact environment representation Stixel World which is based on stereo vision. It allows us to distinguish between static and dynamic obstacles in the current scene which will make our grid maps more accurate. Furthermore, the used camera system is close to those used for current serial production. Next to the pure mapping approach, we also address the research field "mapping under uncertain poses" which leads to the well-known SLAM issue. In our case, we apply state-of-the-art particle filter techniques to solve the SLAM problem on-line. To the best of our knowledge, we are the first who are using Markov Random Fields to model cell dependencies in a grid based SLAM approach in combination with the compact environment representation Stixel World as input data. The created maps are applicable for ADAS and autonomous driving applications, especially for the subtasks of motion planning and localization (see Sec. 1.2.3). Our new mapping approach is well suited for mid and large scale environments like structured residential areas.

1.4 Organization of this Thesis

This thesis is structured as follows: In the first part we give a comprehensive overview of mathematical techniques which are essential for the new mapping approach. This chapter is separated into environment perception with stereo vision, probabilistic undirected graphical models, recursive state estimation, probabilistic map learning, Rao-Blackwellized particle filters and evaluation techniques. In Chapter 3 we present the concept of our new mapping approach which considers dependencies between neighboring grid cells. We explain in detail preprocessing steps, the optimization formulation of our approach, and the definition of the unary and binary terms. Chapter 3.6 presents how we solve the optimization problem with dynamic graph cuts. Afterwards, we solve the SLAM problem by combining our new mapping technique with RBPFs (Chapter4). In Chapter 5, the new mapping approach is evaluated based on artificial and real world data, followed by the evaluation of the new SLAM approach (Chapter 6). Finally, Chapter 7 gives an conclusion of this thesis and discusses future work.

Chapter 2

Technical Background

In this chapter, we present techniques which are the basis for the new mapping approach with MRFs. The first section introduces common sensor types for robotic applications and presents their characteristics in a brief overview (Sec. 2.1). Subsequently, we explain the depth estimation from stereo image sequences (Sec. 2.2). This chapter also includes the description of the compact environment representation Stixel World and its segmentation into static and dynamic obstacles. Further, Chapter 2 outlines probabilistic graphical models (Sec. 2.3), recursive existence estimation techniques (Sec. 2.4), occupancy grid mapping (Sec. 2.5), and the SLAM problem (Sec. 2.6). We also discuss particle filter (Sec. 2.7.1) and graph based approaches (Sec. 2.8) to solve the SLAM problem. Relevant evaluation criteria are presented at the end.

2.1 Sensors of Mobile Platforms

In the research field of autonomous driving different sensor types are used to capture information of the environment of the mobile platform. In connection with this thesis, the term mobile platform is equivalent to the term vehicle. A comprehensive overview of research sensor setups and their main objectives can be found in [Thrun et al., 2006] and [Ziegler et al., 2014]. ADAS must be able to rely on automotive sensor technology. Lindgren and Chen [2006] and Fleming [2008] give a survey of possible serial production sensors. In this thesis sensors are applied which are close in specification to those used for current serial production.

Stereo Vision. Stereo vision imitates human vision by capturing pairwise images using two synchronized cameras. This system is applied for the 3D reconstruction of the environment which is captured in the horizontal and vertical field of view of the camera. Since stereo vision is used during the thesis, we describe the stereo camera configuration, the dense disparity estimation, the computation of the Stixel World, and the object segmentation in Sec. 2.2 in detail.

LIDAR. LIDAR is the most widely used sensor technique in mobile mapping to reconstruct the vehicle's environment [Thrun et al., 2006; Geiger et al., 2012]. In contrast to stereo vision LIDAR sensors are active sensors which scan the environment with the help of laser beams. As an example, Geiger et al. [2012] use the laser scanner Velodyne HDL-64E S2 which has a full 360° horizontal,

and a 26.8° vertical field of view, and a frame rate up to 15 Hz. Because of the high precision of the Velodyne the data set of Geiger et al. [2012] is used as reference data in Sec. 5.2.

Other Sensors. IMUs record the dynamics of the ego vehicle. As an example, the company iMAR employs accelerometers and gyroscopes in their IMUs for navigation and surveying tasks. These sensors are characterized by a high relative precision and are used to estimate the ego motion of the vehicle in a very accurate way. A drawback is the long-term drift behavior of accelerometers and gyroscopes. The consequences of sensor drift to the mapping procedure are discussed in Sec. 4.1.1. Next to IMUs, GPS is often applied to estimate the position of the ego vehicle, but with reference to a global reference coordinate system. A precise overview of different GPS measuring techniques, their advantages, and drawbacks can be found in Bauer [2011]. In practice, GPS information and IMUs are often fused together to estimate a high accurate global ego position of the vehicle [Geiger et al., 2012; Ziegler et al., 2014; iMAR, 2017]. RADAR sensors measure the relative motion of other moving objects directly via Doppler shift frequency modulation. Because of this active and precise velocity estimation technique RADARs are indispensable for modern ADAS ([Schmid et al., 2010; Homm et al., 2010; Continental, 2011]) and they have a high maturity level in the automotive serial production.

2.2 Environment Perception with Stereo Vision

Stereoscopic vision allows for the relative 3D reconstruction of a scene with the help of image sequences taken from two or more different viewpoints [Hartley and Zisserman, 2004; McGlone et al., 2004; Förstner and Wrobel, 2016]. In contrast to LIDAR, cameras are passive sensors which means depth information could not be measured directly. But if the exposure geometry as well as the geometric relationship between different viewpoints is taken into account the 3D reconstruction of the scene for nearly every image pixel is possible. In this section we describe the complete pipeline from the modeling of stereo camera configuration (Sec. 2.2.1 and Sec. 2.2.2) to an efficient and robust representation of dynamic scenes based on stereo vision. It also includes the description of the Stixel World (Sec. 2.2.3).

2.2.1 Projective Camera Model

The projection from a 3D homogeneous scene point ${}^{w}\mathbf{X} = [x, y, z, 1]^{\mathsf{T}}$ in reference to a right handed world coordinate system S_w into the two dimensional sensor coordinate system S_s with a 2D image point ${}^{s}\mathbf{x} = [u, v, 1]^{\mathsf{T}}$ is defined by

$$ {}^{s}\mathbf{x} = \underbrace{{}^{s}\mathsf{K}_c \left[I_3 \mid \mathbf{0}_3 \right]}_{\text{IO}} \underbrace{{}^{c}\mathsf{M}_w}_{\text{EO}} \, {}^{w}\mathbf{X} \tag{2.1} $$

and corresponds to the projective camera model which is described in detail in [McGlone et al., 2004, Chapter 3.2] and in [Hartley and Zisserman, 2004, Chapter 6.1]. In this context, we use the pinhole camera model which means that all projection rays pass only through the projection center $\boldsymbol{X}_O$ which is the origin of the right handed system S_c. This camera model is also visualized in Fig. 2.1. Equation (2.1) is partitioned into the interior (IO) and the exterior (EO) orientation. The

matrix I_3 denotes the 3×3 unity matrix and the vector $\mathbf{0}_3$ represents a three dimensional zero vector. The IO defines the orientation between the camera system S_c and the sensor coordinate system S_s and is characterized by the calibration matrix

$$ {}^{s}\mathsf{K}_c = \begin{bmatrix} f & fs & u_0 \\ 0 & (1+m)f & v_0 \\ 0 & 0 & 1 \end{bmatrix} . \tag{2.2} $$

The principal distance f describes the distance between the projection center $\boldsymbol{X}_O$ and the image plane I. The optical axis intersects the image plane in the principal point $[u_0, v_0]^\mathsf{T}$. The scale difference m and the shear factor s of S_s complete the definition of the affine projection matrix. Apart from the affine intrinsic parameters, we have to consider non-linear projection errors caused by lens distortions or refraction effects. Again, we refer to [McGlone et al., 2004, Chapter 3.2] for more detailed information. The EO represents the transformation between the global world coordinate system S_w and the camera coordinate system S_c and is defined by the homogeneous matrix

$$ {}^{c}\mathsf{M}_w = \begin{bmatrix} R & -R\boldsymbol{X}_O \\ \mathbf{0}_3^\mathsf{T} & 1 \end{bmatrix} \tag{2.3} $$

with the 3D rotation matrix R. To estimate the intrinsic parameters we use the camera calibration tool of Bouguet [2000] which is an implementation of the approach presented by Zhang [2000]. In this context, images at different positions of a planar calibration rig with a checkerboard pattern are captured as shown in Fig. 2.2. For the estimation of the EO correspondences between geometrical entities in the camera system S_c and geometrical entities in the world coordinate system S_w are needed which leads to registration techniques and is not discussed in detail at this point.

2.2.2 Stereo Vision

As mentioned in Sec. 2.1, stereo vision adapts human seeing with the help of a calibrated, synchronized camera pair. A detailed overview of stereo vision concepts and the epipolar geometry can be found in [McGlone et al., 2004, Chapter 3.2.2] and in [Hartley and Zisserman, 2004, Chapter 9]. In the following, we give a brief overview of the ideal stereo configuration (Sec. 2.2.2.1), stereo vision methods (Sec. 2.2.2.2) and the precision of stereo vision (Sec. 2.2.2.3).

2.2.2.1 Ideal Stereo Configuration

The ideal stereo configuration is shown in Fig. 2.3 and represents the *normal case of the image pair* [McGlone et al., 2004, Chapter 3.2.2.5]. This means that the left image plane I^l and the right image plane I^r are parallel. In addition, the transformation between both camera systems S_c^l and S_c^r is only defined by a translation, the baseline b, along the ${}^{c}x^l$-axis. Furthermore, the IO of both cameras is identical and is only defined by the principal distance $f^l = f^r = f$ and the principal point $[u_0^l, v_0^l]^\mathsf{T} = [u_0^r, v_0^r]^\mathsf{T} = [u_0, v_0]^\mathsf{T}$. In order to achieve the normal case of the image pair, a stereo calibration must be performed which can also be done by the tool of Bouguet [2000].

For the 3D reconstruction of the captured scene it is essential to estimate the displacement between corresponding image points ${}^{s}\mathbf{x}^r$ and ${}^{s}\mathbf{x}^l$, as can be shown in Fig. 2.3. Since we employ the normal case of the image pair, the displacement is only defined by a scalar value, the disparity

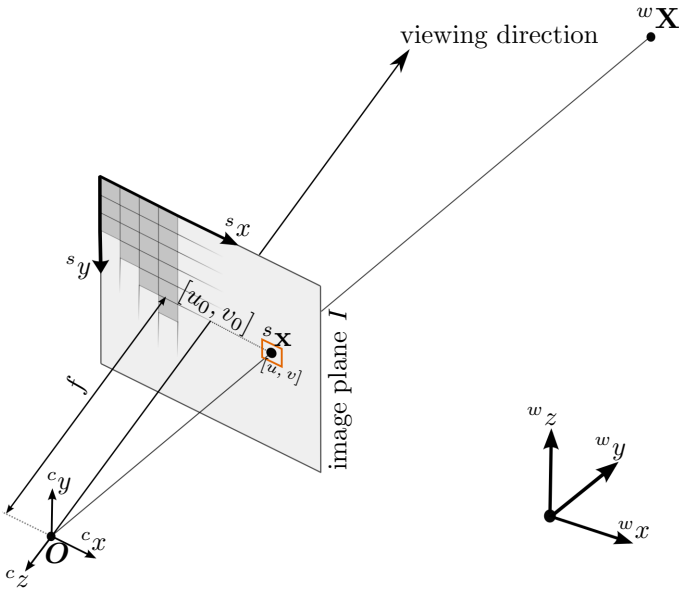


Figure 2.1: The utilized projective model for a single camera. A 3D point $^w\mathbf{X}$ is projected into the two dimensional discrete image plane I to the point $^s\mathbf{x}$ in the sensor. The illustrated pinhole camera model takes the assumption that all rays pass only through the projection center $\boldsymbol{X}_O$. The transformation between the sensor system S_s and the camera system S_c is the interior orientation. The transformation between world and camera system $S_c \rightarrow S_w$ is the exterior orientation.

Figure 2.2: Example images of a planar calibration rig. These images are used to estimate intrinsic parameters for mono cameras (left image) as well as to calibrate stereo camera system (middle and right image).

$d = u^l - u^r$. The disparity d represents the horizontal displacement of the corresponding image points. If the disparity is known and a stereo system is calibrated, the 3D scene point in with

respect to the left camera system is given by

$$ {}^{c}x^{l} = \frac{b}{d}\left(u^{l} - u_0\right) , \tag{2.4} $$

$$ {}^{c}y^{l} = \frac{b}{d}\left(v^{l} - v_0\right) \text{ and} \tag{2.5} $$

$$ {}^{c}z^{l} = \frac{f\ b}{d} . \tag{2.6} $$

To simplify the notation the upper right index of the point coordinates is neglected and we refer all scene points to the left camera system, and define ${}^{c}\boldsymbol{X} = [{}^{c}x, {}^{c}y, {}^{c}z]$.

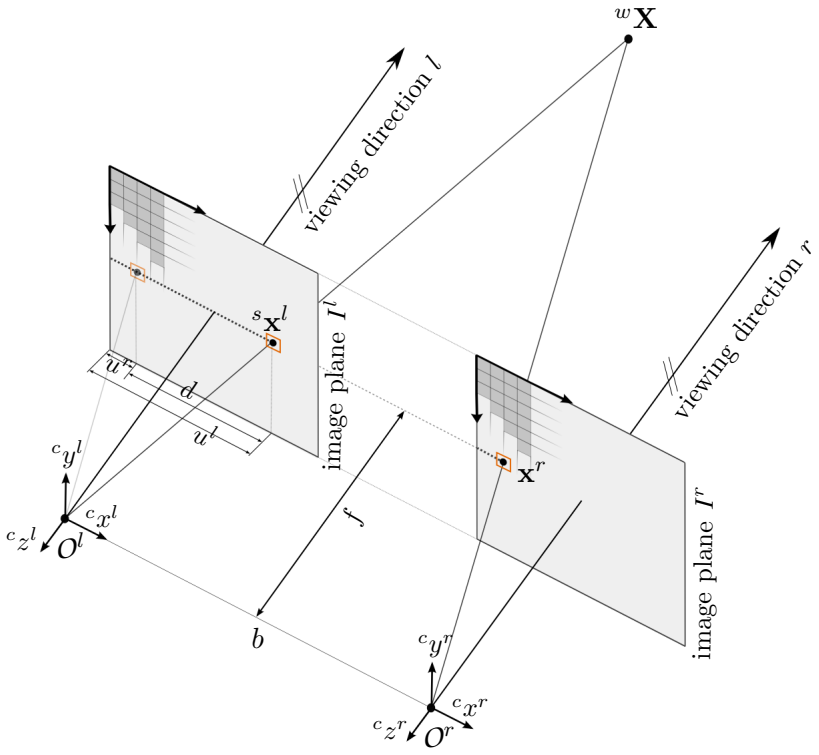


Figure 2.3: The normal stereo case of an image pair. The left image plan I^{l} and the right image plane I^{r} are parallel. The exterior transformation between both camera systems is only defined by the baseline b. The intrinsic parameters of both cameras are identical and are defined by the principal distance f and by the principal point $[u_0, v_0]^{\mathsf{T}}$. The displacement between the corresponding image points ${}^{s}\mathbf{x}^{r}$ and ${}^{s}\mathbf{x}^{l}$ is the disparity d. If the disparity is known and the stereo system is calibrated, the 3D scene point ${}^{c}\boldsymbol{X}$ can be estimated with the help of (2.4)-(2.6).

2.2.2.2 Stereo Vision Methods

The objective of dense stereo vision methods is the estimation of the disparities for each pixel. This results in a 1D matching problem along image rows as long as the stereo system is calibrated (see Sec. 2.2.2.1). Brown et al. [2003] and Nalpantidis et al. [2008] give a detailed review of stereo vision methods and classify them in local stereo and global stereo approaches.

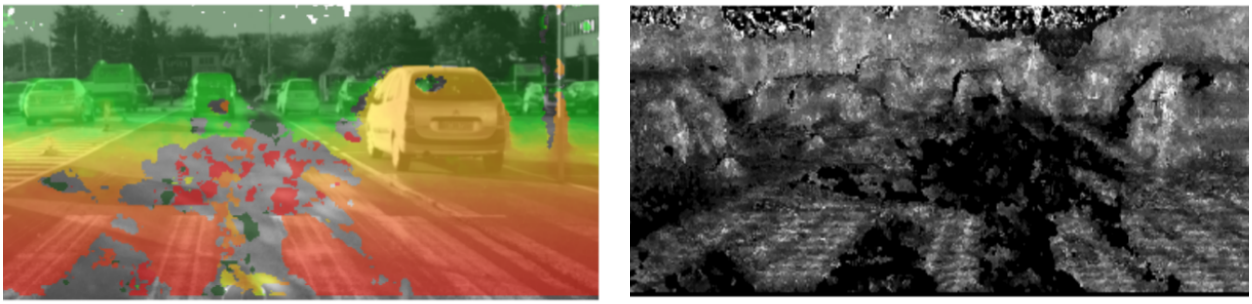
Today's stereo approaches usually cast the matching problem as a global energy optimization problem. These methods take into account not only local distance, but also smoothness and ordering constraints defined over larger image regions [Nalpantidis et al., 2008]. In 2005, Hirschmüller [2005] introduced the very efficient stereo matching method semi-global matching (SGM) which works in a dynamic programming-like fashion [Bellman, 1954] and optimize only along a finite set of lines which passing through the pixel of the reference image. This global optimization technique yields a dense disparity image which is shown in Fig. 2.4. Since only a small set of lines is used, a significant speed up in the computation time is achieved [Hirschmüller, 2008]. Consequently, this allows to use SGM for real-time operations ([Gehrig et al., 2009; Haller and Nedevschi, 2010; Gehrig and Rabe, 2010]).

The approach of Gehrig et al. [2009] was the first real-time implementation of SGM which estimates disparity images $\mathcal{D}$ of the size $W(\text{idth}) \times H(\text{eight}) \in \mathbb{N}^2$ with a rate of 25 Hz up to a maximum size of 1400×400 pel using a field-programmable gate array (FPGA). The disparity values d are defined in the range of $[0...127]$ pel. Due to its efficiency and performance [Scharstein and Szeliski, 2002], this stereo matching approach is used during the thesis. We draw attention to the fact that the SGM algorithm does not provide any kind of precision information for the estimated disparity values.

Figure 2.4: Dense disparity image via semi-global matching (SGM). The color encodes the distance. Red stands for near, green for far away obstacles. For nearly each image point a disparity information is computed (excluding stereo shadow). Equations (2.4)-(2.6) are used to estimate the 3D scene points.

2.2.2.3 The Precision of Stereo Vision

Stereo Confidences. Following the idea of Pfeiffer et al. [2013], for all disparity values corresponding stereo confidence cues $\boldsymbol{c} = [..., c_{ij}, ...]$ with $i \in W$ and $j \in H$ are applied during this thesis. The estimation of the confidence cues is based on the metric described in [Wedel et al., 2009]. Wedel et al. [2009] use the slope of the disparity cost function as a confidence of the disparity measurements. The confidence value c is scaled to the interval [0...1] and is interpreted as follows: if the slope of the cost function is high ($c \to 1$), the sub-pixel position of the disparity is expected to be accurate which results in a precise disparity measurement. If the slope of the cost function is low ($c \to 0$), the estimation of the disparity value is not accurate in the sense that another disparity value could also be possible. An example of a stereo confidence image is shown in Fig. 2.5 where the confidence cues are presented in a gray scale color encoding. Black stands for inaccurate (or missing) disparity information and white for a very precise disparity measurement. Please take into account that the described confidence measurements are not probabilistic entities. As highlighted in [Pfeiffer et al., 2013], the confidence measurements are transformed into outlier probabilities. These probabilities are used to optimize the Stixel World which is described in Sec. 2.2.3.



(a) Disparity image (b) Confidence image

Figure 2.5: Dense disparity image via SGM 2.6(a) and the confidence image 2.6(c). The brighter a pixel is, the higher the confidence that the disparity measurement is correct. Image regions with disturbances, e.g. strong reflections on the street surface, result in inaccurate disparities and low confidence values. Good structured, vertical surfaces, like the rear of the right car, have high confidence cues.

Theoretical Precision of Triangulated 3D Points. In the following we describe the estimation of the precision of triangulated 3D points because it required in later sections (Sec. 6.2). With the help of the triangulation equations (2.4)-(2.6) and error propagation, the theoretical precision

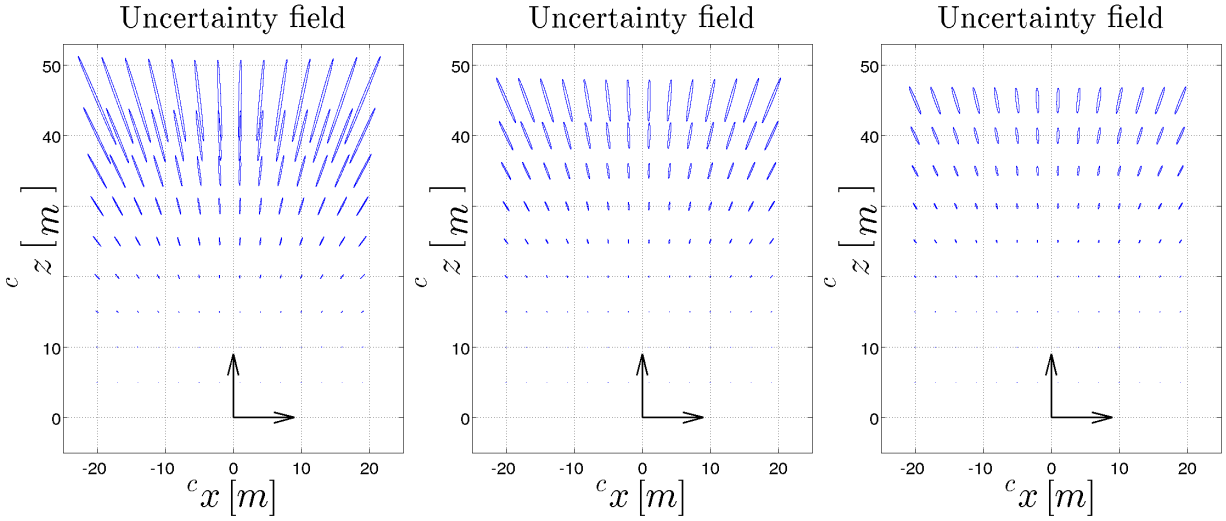
of a 3D point ${}^c\boldsymbol{X}$ is given by

$$\sigma_{c_x}^2 = \left(\frac{b}{d}\right)^2 \sigma_u^2 + \left(\frac{b(u^l - u_0)}{d^2}\right)^2 \sigma_d^2 \ , \tag{2.7}$$

$$\sigma_{c_y}^2 = \left(\frac{b}{d}\right)^2 \sigma_v^2 + \left(\frac{b(v^l - v_0)}{d^2}\right)^2 \sigma_d^2 \ , \text{ and} \tag{2.8}$$

$$\sigma_{c_z}^2 = \left(\frac{-bf}{d^2}\right)^2 \sigma_d^2 \ = \left(\frac{{}^cz^2}{fb}\right)^2 \sigma_d^2. \tag{2.9}$$

We take the assumption that only the disparity d and the image detection point $[u^l, v^l]^\mathsf{T}$ are uncertain. The precision of an image point σ_u^2 and σ_v^2 is in the range of $0.10\,\text{pel} < \sigma_v, \sigma_u < 0.25\,\text{pel}$. The precision of a disparity value σ_d^2 is in the range of $0.25\,\text{pel} < \sigma_d < 0.50\,\text{pel}$. This results from empirical studies of the Daimler research group of image understanding[1]. Equation 2.9 shows, that the standard deviation of the distance σ_{c_z} is proportional to the square of the distance and inversely proportional to the factor fb. This characteristic is typical for uncertain behavior in stereo vision. Figure 2.6 shows uncertainty fields of triangulated points in a 2D bird's eye view. As one can see, the uncertainty increases quadratically with increasing depth. The larger the baseline b and the focal length f, the better the precision of the 2D points.



(a) baseline $b = 25\,\text{cm}$ (b) baseline $b = 40\,\text{cm}$ (c) baseline $b = 55\,\text{cm}$

Figure 2.6: Uncertainty Fields of triangulated points in the ${}^cx$-${}^cz$-plane for different baselines and a fixed focal length. The uncertainties are set to $\sigma_d^2 = 0.5\,\text{pel}$ and $\sigma_u^2 = 0.25\,\text{pel}$. The uncertainty in the distance increases quadratically with increasing depth which is typical for stereo vision.

[1] http://www.6d-vision.com/

2.2.3 The Stixel World

In this section, we introduce the compact environment representation Stixel World [Pfeiffer and Franke, 2011] which is used during this thesis as input data. We describe the main idea and the generation of the Stixel World in the following.

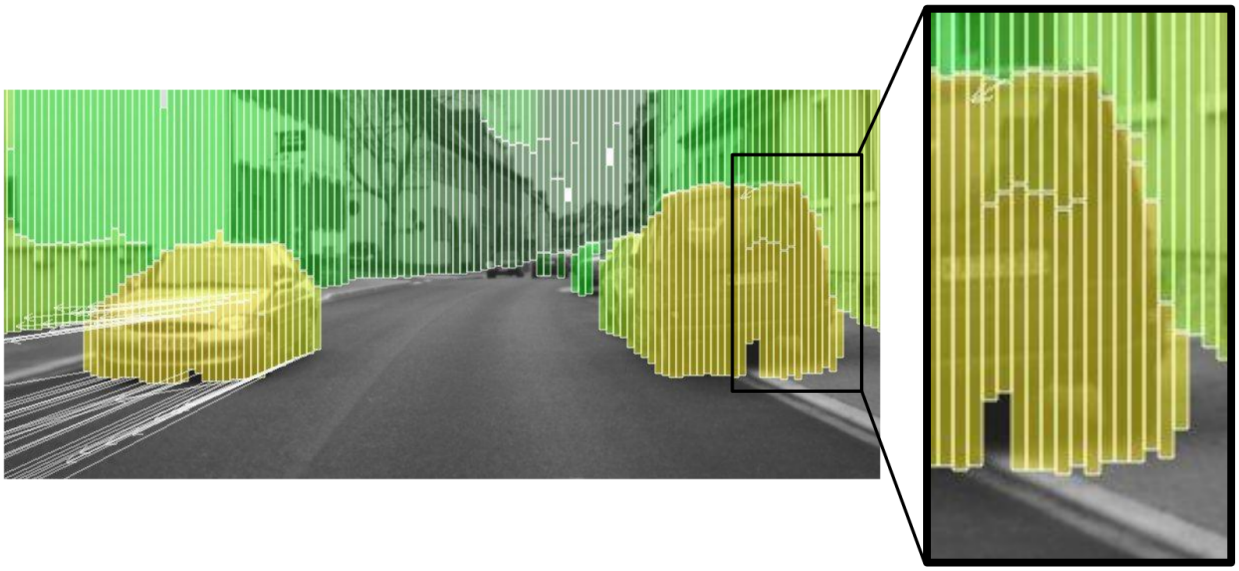
Key Idea of the Stixel World. In spite of the fact that stereo approaches produce dense and accurate 3D point clouds (see Sec. 2.2.2.2), the amount of data as output of this type of system is enormous. Consequently, stereo based object recognition systems [Erbs et al., 2012; Muffert et al., 2013] are required to process the large amount of data in real-time. As described in [Stein, 2012], automotive object recognition systems are limited by the CPU- and GPU-power and need low memory requirements. To handle this challenge and to produce a compact and robust scene representation, medium-level representations like described in [Felzenszwalb and Huttenlocher, 2004; Hornung et al., 2013; Veksler et al., 2010; Pfeiffer and Franke, 2011; Achanta et al., 2012] are used for the named tasks above.

The Stixel World [Badino et al., 2009; Pfeiffer and Franke, 2011] segments the current disparity image, and therefore the current 3D scene of the environment, into free space and object information. By exploiting the fact that the most man-made environments have either vertical or horizontal planar surfaces, objects are segmented into vertically oriented, adjacent rectangles which is shown in Fig. 2.7. Each rectangle, called a Stixel[2], has a fixed width in the image (e.g. 5 pel) and a variable height. The image region from the bottom of the image to the base point of a Stixel is defined as free space (see Fig. 2.7(a)). The Stixel World allows for an enormous reduction of the raw disparity input data. As an example, approximately 550 000 disparity measurements from a 1400×400 pel stereo image pair are reduced to only a few hundred Stixels. This data compression reduces the computational burden by a factor of roughly 1 000 without losing relevant information about the current scene. In addition, the Stixel World is robust against single stereo outliers. Subsequent driver assistance applications [Erbs et al., 2012; Enzweiler et al., 2012; Benenson et al., 2012; Muffert et al., 2013] benefit from this representation. In summary, the Stixel World gives access to relevant information such as free space and obstacles in a dynamic scene, and thus effectively bridges the gap between pixel-based and object-based vision.

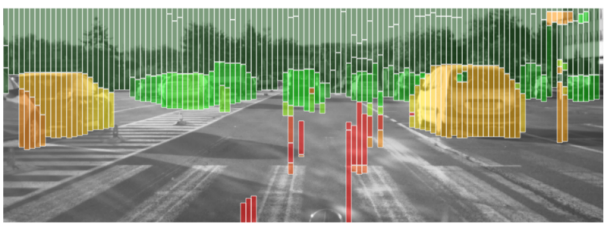
Generation and the Parametrization of the Static Stixel World. The generation of the Stixel World leads to a typical MAP estimation problem which is solved by the use of dynamic programming [Bellman, 1954]. It results in the most likely segmentation of the disparity map $\mathcal{D}$ into the classes $\mathcal{C}_{Stixel} \in \{\texttt{free space}, \texttt{obstacle}\}$ from the possible labeling set $\mathbb{S}$. The segmentation is regularized by a set of physically motivated world model priors which include gravity and ordering constraints. Follow the notation of Pfeiffer and Franke [2011], the Stixel labeling is equivalent to the most probable Stixel labeling $\mathcal{S}^*$ with

$$\mathcal{S}^* = \arg\max_{\mathcal{S}\in\mathbb{S}} \; p\big(\mathcal{S} \mid \mathcal{D}\big) \; . \tag{2.10}$$

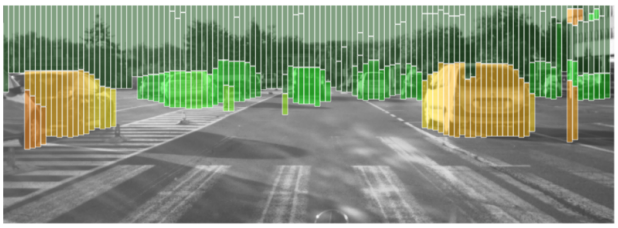
[2]The word creation Stixel combines stick and pixel



(a) The Stixel World



(b) The Stixel World without confidence measurements



(c) The Stixel World with confidence measurements

Figure 2.7: Examples of the Stixel World.

Because of reasons of efficiency, the segmentation step is decomposed into W column-wise labelings $\mathcal{S}_u$ where each column labeling at image position u is sub-classified into N_u layered segments $\boldsymbol{s}_{un}^{\text{map}}$:

$$\mathcal{S} = \{\mathcal{S}_u\}, \text{ with } 0 \leq u < W$$
$$\mathcal{S}_u = \{\boldsymbol{s}_{un}^{\text{map}}\}, \text{ with } 1 \leq n \leq N_u \leq H.$$

Each segment $\boldsymbol{s}_{un}^{\text{map}}$ is defined as a single Stixel with its width w. A Stixel has the composition

$$\boldsymbol{s}_{un}^{\text{map}} = [u, v_{un}^{\text{bt}}, v_{un}^{\text{tp}}, w, d_{un}, \tilde{\sigma}_{d_{un}}^2, c_{un}, p_{un}^{\text{out}}], \text{ with } 0 \leq v_{un}^{\text{bt}} \leq v_{un}^{\text{tp}} < H, c_{un} \in \mathcal{C}_{Stixel}. \tag{2.11}$$

The row coordinates v_{un}^{bt} and v_{un}^{tp} represent the Stixel bottom and the Stixel top point, respectively. The disparity of the Stixel d_{un} is estimated by an arbitrary function $f(\boldsymbol{d}_{un})$ which takes all disparity values $\boldsymbol{d}_{un} = [..., d_{ij}, ...]$ with $v_{un}^{\text{bt}} \leq i \leq v_{un}^{\text{tp}}$ and $u - \frac{1}{2}w \leq j \leq u + \frac{1}{2}w$ of the current Stixel segment into account. For obstacles with the same depth, a straight-forward definition of $f(\boldsymbol{d}_{un})$ is the

mean estimation over all disparities. The precision of each Stixel is represented by the empirical standard deviation

$$\tilde{\sigma}^2_{d_{un}} = \frac{1}{IJ-1}\sum_{i=1}^{I}\sum_{j=1}^{J}(d_{ij}-d_{un})^2. \tag{2.12}$$

To achieve the best probable Stixel results, (2.10) is solved by the use of dynamic programming. For further details we refer Pfeiffer [2011]. As described in Sec. 2.2.2.3, Pfeiffer et al. [2013] map the stereo confidence cues $\boldsymbol{c}$ to Stixel outlier probabilities p^{out}_{un} with $\boldsymbol{c}_n \rightarrow p^{\text{out}}_{un}$. These probabilities are taken into account during the Stixel segmentation which reduces the number of Stixel outliers significantly. Examples of the Stixel World with and without the use of p^{out}_{un} is shown in Fig. 2.8.

Generation of the Dynamic Stixel World. Up to this point, the Stixel World only describes the current, static world geometry. For the purpose of dynamic objects detection, the Stixel based tracking scheme proposed in [Pfeiffer and Franke, 2010] is chosen. An example of tracked Stixels is shown in Fig. 2.7(a). Besides the use of the disparity map, this scheme additionally requires optical flow information [Tomasi and Kanade, 1991] as well as the vehicle's motion which is computed by visual odometry [Badino, 2004] or IMUs. To estimate the motion properties of the Stixels, the 6D-vision principle suggested by Franke et al. [2005] is applied. Taking into account that all relevant dynamic Stixels are expected to move earthbound, only the longitudinal and lateral velocity ${}^c\dot{z}$ and ${}^c\dot{x}$ with reference to the camera system c are estimated. Therefore, we enrich the definition of a single Stixel by these two velocity components:

$$\boldsymbol{s}^{\text{dyn}}_{un} = \{\boldsymbol{s}^{\text{map}}_{un}, {}^c\dot{z}_{un}, {}^c\dot{x}_{un}\}. \tag{2.13}$$

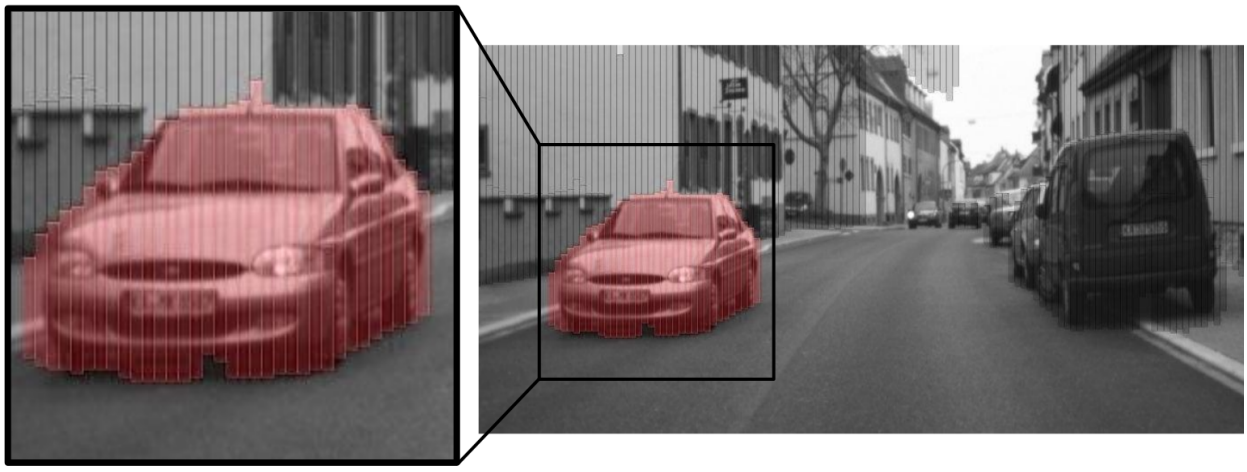
Segmentation of the Dynamic Stixel World. The separation into moving and stationary obstacles is achieved by a multi-class traffic scene segmentation introduced by Erbs et al. [2012]. The approach is based on the previously described dynamic Stixel World where each Stixel is assigned to a specific motion class or to static background. The approach is formulated in a CRF framework (see Sec. 2.3), is real-time capable, and yields highly accurate classification results in urban traffic scenarios [Erbs et al., 2012; Muffert et al., 2013]. The goal of the Stixel segmentation is to find the most probable labeling $\mathcal{L}^*_t$ at time step t, defined as

$$\mathcal{L}^*_t = \arg\max_{\mathcal{L}_t \in \mathbb{L}} p\big(\mathcal{L}_t \mid \mathcal{S}^{\text{dyn}}_t, \mathcal{L}_{t-1}\big)\ , \tag{2.14}$$

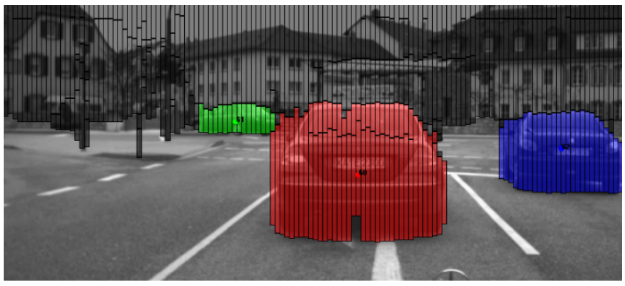
where $\mathbb{L}$ is the complete label space. The dynamic Stixel World $\mathcal{S}^{\text{dyn}}_t$ represents the observations in the maximization step. Equation (2.14) is inferred by the multi-class graph cuts optimization scheme described in [Boykov et al., 2001]. We add the labeling class $l_{un} \in \mathcal{L}$ with $\mathcal{L} \in \{\texttt{movingObj}, \texttt{staticObj}\}$ into the data structure of the dynamic Stixel World which results in the final Stixel definition:

$$\boldsymbol{s}^{\text{lab}}_{un} = \{\boldsymbol{s}^{\text{dyn}}_{un}, l_{un}\}. \tag{2.15}$$

The Stixel set $\mathcal{S}^{\text{lab}}$ is used as input data for our mapping approach. Figure 2.8 shows examples of the Stixel segmentation into moving objects, and into static environment information.



(a) 1st example of the dynamic Stixel World segmentation Erbs et al. [2012]. The vehicle on the left side is classified as an oncoming object. The corresponding Stixels are red. The static environment is colored in dark grey.



(b) 2nd example of the dynamic Stixel World segmentation.

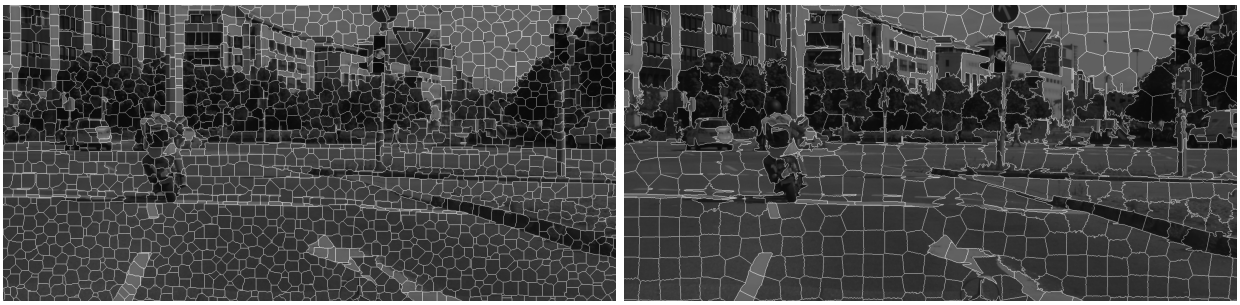
(c) 3rd example of the dynamic Stixel World segmentation.

Figure 2.8: Examples of object segmentation of the dynamic Stixel World [Erbs et al., 2012]. The examples (b) and (c) clearly show that the algorithm distinguish between different motion models which is represented by the different colors. In all cases, the static background information is classified in a very precise way. The images of the 2 nd and 3 rd example were directly taken from Erbs et al. [2012].

Alternative Super Pixel Representations. Next to the described Stixel World, there exists a lot of other super pixel representation which segment gray scale or disparity images into meaningful regions. Here, we give a small overview of three alternative techniques which are often cited in literature. Felzenszwalb and Huttenlocher [2004] introduced a graph based super pixel approach. The super pixels are created by minimizing cost functions which are defined by a graph structure (see also Sec. 2.3). The algorithm works well at image boundaries but the shape and the size of the super pixels is irregular. Veksler et al. [2010] also rely on a graph based algorithm. Super pixels are estimated by producing overlapping patches and stitching them together in an optimal way.

(a) Original gray scale image

(b) Felzenswalb and Huttenlocher [Felzenszwalb and Huttenlocher, 2004]



(c) Veksler et al. [Veksler et al., 2010]

(d) SLIC [Achanta et al., 2012]

Figure 2.9: Alternative super pixel approaches.

The formulated energy function is solved by using graph cuts [Boykov et al., 2001]. The super pixel representation SLIC [Achanta et al., 2012] stands for simple linear iterative clustering and is an adaption of the well-known k-means clustering algorithm [Mackay, 2003, chapter 20.1]. In contrast to the k-means clustering approach, the search space is limited by the maximum size of a super pixel which reduces the computation time. Furthermore, the distance measurement combines both color similarity and spatial proximity, where the size of the super pixel is controlled. The number of desired super pixels is an important control parameter in this approach. Examples of the three described super pixel approaches are illustrated in Fig. 2.9.

2.3 Probabilistic undirected Graphical Models

In this section we describe probabilistic graphical models which represent the conditional independence properties of random variables in a schematic representation. At the beginning, Markov random fields (MRFs) are introduced, followed by the description of the well-known inference estimation technique graph cuts (Sec. 2.3.2). In Sec. 2.3.3 we present a technique how to estimate uncertainties in graph cuts solutions. This includes the estimation of min-marginal energies and the use of a very efficient, iterative graph cuts scheme, also known as dynamic graph cuts.

2.3.1 Markov Random Fields

Markov random fields (MRFs) are graphical models which describe the probability of a set of correlated random variables $\boldsymbol{x} = [..., x_i, ...]$ with $i \in \mathcal{I}$ in an undirected graph. The random variables describe labels from the set $\mathcal{X}$ of all possible labellings. In the graph structure, the random variables represent the nodes. The dependencies between the nodes are defined by undirected edges. These edges represent the neighboring structure of the complete graph, which, in turn, reflects the conditional independence properties of the random variables. The *Markov assumption* implies that a random variable does not depends on all other variables, e.g. $p(x_k \mid \boldsymbol{x}_{\setminus x_k}) = p(x_k \mid N(x_k))$. Here, the expression $\boldsymbol{x}_{\setminus x_k}$ means that we exclude x_k from the complete random set $\boldsymbol{x}$ and $N(x_k)$ describes the neighborhood region of x_k. The Markov assumption allows us to make useful simplifications with regard to dependencies between the nodes. As an example, in image segmentation approaches the assumption is often made that not all pixels of an image are mutually dependent. With the definition of the Markov assumption the joint probability $p(\boldsymbol{x})$ could be factorized over cliques (complete subgraphs) $C \in \mathcal{C}$ of the graph by following the fundamental *Hammeresley-Clifford* theorem [Hammersley and Clifford, 1971; Clifford, 1990]

$$p(\boldsymbol{x}) = \frac{1}{\zeta} \prod_{C \in \mathcal{C}} \Psi_C(\boldsymbol{x}_C) \ , \tag{2.16}$$

where $\Psi_C(\boldsymbol{x}_C)$ are the potential function over the cliques $C \in \mathcal{C}$ with the associated random variables $\boldsymbol{x}_C$. The Denominator ζ describes the normalization constant and is also known as the partition function:

$$\zeta = \sum_{\boldsymbol{x} \in \mathcal{X}} \prod_{C \in \mathcal{C}} \Psi_C(\boldsymbol{x}_C) \ . \tag{2.17}$$

An example of a MRF is illustrated in Fig. 2.10. It shows a grid of random variables and the dependencies between them. Using the definition of maximal cliques, the joint probability is factorized into $p(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = \frac{1}{\zeta}\Psi_1(x_1, x_2, x_5, x_6)\Psi_1(x_2, x_3, x_6)\Psi_1(x_3, x_6, x_7)$
$\Psi_1(x_3, x_4)$ in this case.

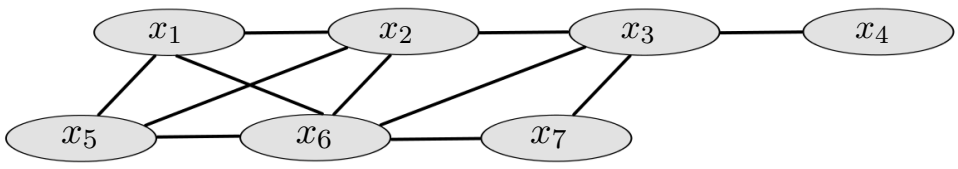


Figure 2.10: Example of a Markov random field (MRF). The black edges describe the dependencies between the random, gray variables. The joint distribution $p(x_1, x_2, x_3, x_4, x_5, x_6, x_7)$ is decomposed into 1×4-clique $\Psi_1(x_1, x_2, x_5, x_6)$, 2×3-cliques $\Psi_1(x_2, x_3, x_6)$, $\Psi_1(x_3, x_6, x_7)$ and 1×2-clique $\Psi_1(x_3, x_4)$.

A subclass of MRFs are Conditional random fields (CRFs) which model dependencies between the random variables $\boldsymbol{x}$ given the observations $\boldsymbol{z}$. CRFs were introduced by Lafferty et al. [2001] and are often used for scene segmentation approaches [Wojek and Schiele, 2008; Erbs et al., 2012; Scharwächter et al., 2013], or for object recognition [Wedel et al., 2009; Barth et al., 2010]. The overall conditional probability $p(\boldsymbol{x} \mid \boldsymbol{z})$ is defined by the product over the potential functions which model the dependencies between the random variables $\boldsymbol{x}_C$ given the observations $\boldsymbol{z}$:

$$p(\boldsymbol{x} \mid \boldsymbol{z}) = \frac{1}{\zeta} \prod_{C \in \mathcal{C}} \Psi(\boldsymbol{x}_C \mid \boldsymbol{z}) \ . \tag{2.18}$$

If we have a grid with a 4-neighborhood structure N_4 between the nodes, and consequently a maximum clique of two, the global joint distribution $p(\boldsymbol{x} \mid \boldsymbol{z})$ can be decomposed into the product of unary $\Psi(x_i \mid \boldsymbol{z})$ and binary terms $\Phi(x_i, x_j \mid \boldsymbol{z})$:

$$p(\boldsymbol{x} \mid \boldsymbol{z}) = \frac{1}{\zeta} \prod_{C \in \mathcal{I}} \Psi(x_i \mid \boldsymbol{z}) \prod_{(i,j) \in N_4} \Phi(x_i, x_j \mid \boldsymbol{z}) \ . \tag{2.19}$$

The unary terms model the individual state (label) decision for each node x_i individually given the data $\boldsymbol{z}$. The binary, or smoothness terms model the relation and the dependencies between neighbored nodes. As also mentioned in [Hammersley and Clifford, 1971], the defined configuration of the CRF of (2.19) is often expressed as an energy function:

$$E(\boldsymbol{x} \mid \boldsymbol{z}) = -\log p(\boldsymbol{x} \mid \boldsymbol{z}) \tag{2.20}$$

$$= \sum_{C \in \mathcal{I}} \underbrace{-\log \Psi(x_i \mid \boldsymbol{z})}_{:= E_i(x_i \mid \boldsymbol{z})} + \sum_{(i,j) \in N_4} \underbrace{-\log \Phi(x_i, x_j \mid \boldsymbol{z})}_{E_{ij}(x_i, x_j \mid \boldsymbol{z})} + \underbrace{\log(\zeta)}_{:= const.} \tag{2.21}$$

$$\propto \sum_{C \in \mathcal{I}} E_i(x_i \mid \boldsymbol{z}) + \sum_{(i,j) \in N_4} E_{ij}(x_i, x_j \mid \boldsymbol{z}) \ . \tag{2.22}$$

The two terms are called unary energies for $E_i(x_i \mid \boldsymbol{z})$, and binary energies for $E_{ij}(x_i, x_j \mid \boldsymbol{z})$. In the following, we describe a common technique to determine the optimum class assignments of each nodes in MRFs/CRFs. This task is also known as inference. Later on in Sec. 2.3.3, we also present an approach to estimate uncertainties for the label results.

2.3.2 Inference Estimation via Graph Cuts

The inference estimation problem in MRFs or CRFs can be cast as a MAP problem to find the optimum labellings or states $\hat{\boldsymbol{x}}$:

$$\hat{\boldsymbol{x}} = \underset{\boldsymbol{x}}{\operatorname{argmax}}\ p(\boldsymbol{x} \mid \boldsymbol{z}) \ . \tag{2.23}$$

This maximization step is also equivalent to minimizing the previous defined energy function (see (2.20))

$$\hat{\boldsymbol{x}} = \underset{\boldsymbol{x}}{\operatorname{argmin}}\ E(\boldsymbol{x} \mid \boldsymbol{z})\ . \tag{2.24}$$

The minimization of (2.24) is in general NP[3]-hard [Blake and Zisserman, 1987; Boykov et al., 2001]. This means, that there is no known algorithm or approach to solve (2.24) in polynomial time. Boykov et al. [2001] presented a technique to estimate the exact inference for undirected graphs for binary classification problems under the assumption that the binary energies must follow the sub-modularity condition [Kolmogorov and Zabih, 2004]: the labels or states must be $x_i \in \{0, 1\}$, and the binary energies must satisfy the sub-modularity condition

$$E_{ij}(0,1) + E_{ij}(1,0) \ \geq\ E_{ij}(1,1) + E_{ij}(0,0)\ . \tag{2.25}$$

If both conditions are applied, then the energy minimization problem from (2.24) is equivalent to finding the *minimum cut* of the constructed acyclic graph which is known by the term of *graph cuts* [Boykov et al., 2001]. Following the theorem of L. R. Ford and Fulkerson [1962], the task of finding the minimum cut in the acyclic graph is equivalent to compute the maximum possible flow from a terminal source node s through the graph to the sink node t. Therefore, the graph cuts problem is also known as the *st-mincut/max-flow* problem in literature. By adding the source node and the sink node to the original graph, the new flow graph has $N + 2$ nodes, where N is the number of nodes from the MRF/CRF. The edges from the nodes of the MRF/CRF to the s-node and to the k-node are estimated from the unary energies $E_i(x_i = 1)$ and $E_i(x_i = 0)$. The edges between the nodes of the MRF/CRF are calculated from the binary energies where the sub-modularity constraint from (2.25) is taken into account. For further details of the (flow) graph construction we refer to Kolmogorov and Zabih [2004]. The described idea of graph cuts is shown in Fig. 2.11 which illustrates an example of a binary segmentation for a 2×2 graph.

In contrast to other inference algorithms, e.g. belief propagation [Mackay, 2003, chapter 26.2], graph cuts do not provide any uncertainty values for the estimated states or label assignments. However, Kohli and Torr [2008] introduced a procedure to estimate uncertainty measurements of MAP-MRFs/CRFs estimation problems which is presented in the next section.

2.3.3 Uncertainties in Graph Cut Solutions

To the best of our knowledge, Kohli and Torr [2008] were the first who introduced a technique to measure uncertainties in graph cuts solutions. In the following we describe how these *marginal probabilities* can be estimated in an efficient way using the derivation and notation of Kohli and Torr [2008].

Estimation of Marginal Probabilities. First, the max-marginal probability $\nu_{i;j}$ is defined which represents the maximum probability over all possible configurations of the MRF/CRF in which the latent variable x_i is labeled to the class j. Mathematically, $\nu_{i;j}$ is defined as

$$\nu_{i;j} = \underset{\boldsymbol{x} \in \mathcal{X}, x_i = j}{\operatorname{argmax}}\ p(\boldsymbol{x} \mid \boldsymbol{z})\ . \tag{2.26}$$

[3] Non-deterministic Polynomial-time

To estimate the probabilities $p(\nu_{i;j})$ for any latent variable, the max-marginal probabilities are normalized with regard to the possible labellings:

$$p(\nu_{i;j}) = \frac{\nu_{i;j}}{\sum\limits_{k \in \mathcal{J}} \nu_{i;k}} \ . \tag{2.27}$$

Since we only regard binary labeling problems during this thesis, (2.27) is simplified to:

$$p(\nu_{i;0}) = \frac{\nu_{i;0}}{\nu_{i;0} + \nu_{i;1}} \ . \tag{2.28}$$

Kohli and Torr [2008] show that the max-marginal probabilities $\nu_{i;0}$ can be estimated from the min-marginal energies $\phi_{i;j}$ which results from graph cuts solutions:

$$\nu_{i;j} = \frac{1}{\zeta} \ \exp\Big(- \underbrace{\min_{\boldsymbol{x} \in \mathcal{X}, x_i = j} E(\boldsymbol{x} \mid \boldsymbol{z})}_{:= \phi_{i;j}} \Big) \ . \tag{2.29}$$

Combining (2.28) and (2.29) the probabilities $p(\nu_{i;j=0,1})$ can be expressed by the min-marginal energies:

$$p(\nu_{i;0}) = \frac{\frac{1}{\zeta} \ \exp\big(- \phi_{i;0} \big)}{\frac{1}{\zeta} \ \exp\big(- \phi_{i;0} \big) + \frac{1}{\zeta} \ \exp\big(- \phi_{i;1} \big)} \tag{2.30}$$

$$= \frac{\exp\big(- \phi_{i;0} \big)}{\exp\big(- \phi_{i;0} \big) + \exp\big(- \phi_{i;1} \big)} \ . \tag{2.31}$$

The "trick" of the probability estimation is to minimize step-by-step over a modified energy function where the value of the latent variable x_i is fixed to label 0 and to label 1, respectively. This is solved by setting the unary energy terms of $x_{i;j}$ in each case $j = \{0, 1\}$ to a huge number. In Fig. 2.12 the uncertainty images for a typical foreground-background classification are shown by using graph cuts and the previous described approach. The minimization of the modified energy function can solved via graph cuts. However, there is a big disadvantage of using common graph cuts techniques to estimate the probabilities $p(\nu_{i;j=0,1})$. To estimate the uncertainty values of a MAP-MRF/CRF solution, we have to compute for each single min-marginal energy $\phi_{i;j}$ a single graph cuts solution. Assuming a binary segmentation problem for an 600×500 pel image and a computational time for a single graph cuts of 15 ms, we have to compute $600 \times 500 \times 2 = 600\,000$ min-marginal energies which would results in an overall computational time of 2.5 hours. This computational burden is unacceptable for real-time applications. Thus, we present in the following section a highly efficient implementation of graph cuts, the *dynamic graph cuts* [Kohli and Torr, 2007].

Efficient Computation via Dynamic Graph Cuts. Kohli and Torr [2007] presented the dynamic graph cuts scheme to solve the st-mincut/max-flow problem in dynamically changing MRF models. Here, the new algorithm exploits only small changes between two graphs G_1 and

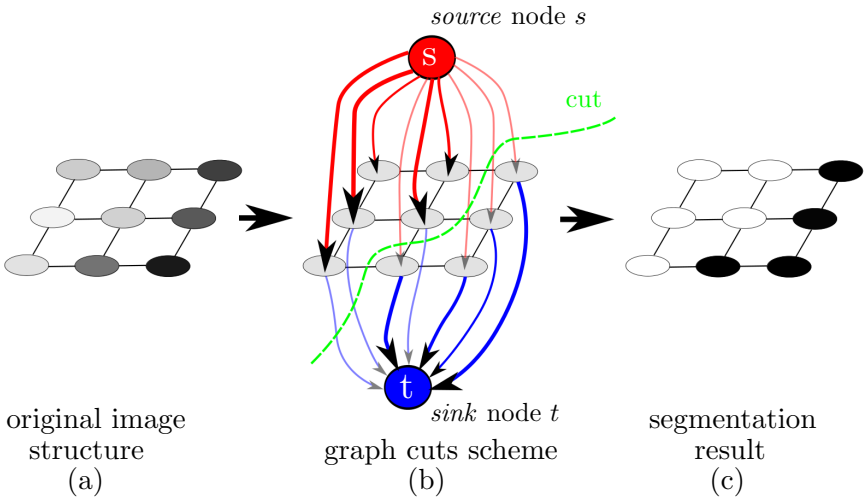


Figure 2.11: Concept of graph cuts for a 3×3 image. The left graph (a) represents the original image with gray scale values. The general graph cuts concept with the st-mincut is shown in (b): The random variables of the MRF structure are either assigned to the source node s or to the sink node t. The resulting minimum cuts shown in green minimizes (2.24). The binary MAP result is shown in (c). The illustration was inspired by Kolmogorov and Zabih [2004]

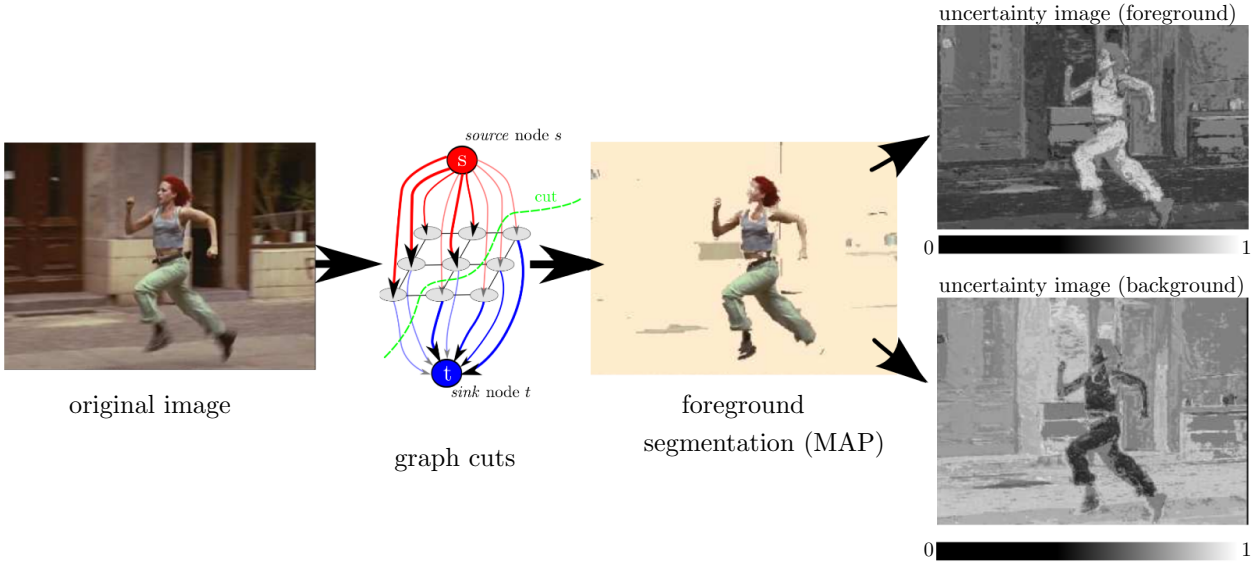


Figure 2.12: Example of foreground segmentation via graph cuts and the resulting probability images. The uncertainty images for both foreground (class label 0) and background (class label 1) are estimated from the min-marginal energies $\phi_{i;0}$ and $\phi_{i;1}$ respectively (see (2.31)). The example was taken from Kohli and Torr [2008] and shows a scene from the movie "run, Lola, run".

G_2 which represent similar MRFs. During the st-mincut/max-flow computation of graph G_1 the algorithm stores the flow through the graph and restructure the original graph G_1 into a residual graph G_1^r. The residual graph G_1^r is a re-parametrization of G_1 and differs only in the capacity of its edges. Under the assumption that G_2 differs only a little bit from G_1, and we know the changes between both graphs, the previous estimated flow and the residual graph G_1^r can be efficiently reused to estimate the st-mincut/max-flow of G_2. For detailed insights how the residual graph is estimated and how it is reused we refer to Kohli and Torr [2007].

In connection with the estimation of uncertainty values, the technique of dynamic graph cuts is optimal since only one unary energy term is changed between consecutive graphs. Kohli and Torr [2008] showed that the computational time for the estimation of uncertainty values is reduced by an immense factor of approximately 2.5×10^4 in a MFR with 10^5 nodes and a 4- neighborhood structure. Thus, the estimation of uncertainty values is possible in real-time since the dynamic graph cuts scheme is applied. An overall algorithm for the estimation of uncertainty values for MAP-solutions with dynamic graph cuts is shown in Algorithm 1. The implementation of the described approach is available as an open source C++ library [4].

Algorithm 1: Estimation of marginal probabilities with dynamic graph cuts

Input: Unary and binary energy terms of a two class segmentation problem
Output: MAP solution, uncertainty values (marginal probabilities)

Construct graph G with unary and binary energy terms.
Compute the st-mincut of G. It results in the MAP solution and in residual graph G^r.
Initialize uncertainty vector: $\boldsymbol{c} = \emptyset$
for *i = 1 to N (Nr. of nodes)* **do**
    for *j = 0 to 1 (Nr. of classes)* **do**
        Perform following steps to compute min-marginal energies ϕ_{ij}:
        - obtain energy E' where the value of the latent variable x_i is fixed to label j
        - construct graph G'
        - re-parameterization between G^r and G' to obtain G'^r using dynamic graph cuts update scheme
        - compute the st-mincut/max-flow of G'^r with dynamic graph cuts algorithm
    Estimate uncertainty values (marginal probabilities):
    $p(\nu_{i;0}) = \frac{\exp\left(-\phi_{i;0}\right)}{\exp\left(-\phi_{i;0}\right)+\exp\left(-\phi_{i;1}\right)}$
    Save: $\boldsymbol{c} = \boldsymbol{c} + [p(\nu_{i;0})]$;
    $G^r = G'^r$;
return $\boldsymbol{c}$

[4] http://research.microsoft.com/en-us/um/people/pkohli/code.html, (2015-12-07)

2.4 Probabilistic Recursive Existence Estimation

In this section we describe the estimation of states from sensor data in a time recursive way. Probabilistic recursive state algorithms are applied which means that we estimate belief distributions over possible states given a set of uncertain measurements [Thrun et al., 2005, chapter 2]. In comparison with undirected graphical models (see Sec. 2.3), recursive state algorithms can be expressed as directed graphs which is shown in Fig. 2.13. The recursive existence estimation problem is presented in Sec. 2.4.1. The general case of the recursive existence estimation problem can be reduced to the estimation of a discrete, binary state which is described in Sec. 2.4.2. Basic probabilistic concepts and rules are descibed in detail in Thrun et al. [2005]. In addition, Bergmann [1999] and Durrant-Whyte [2001] also derivated the recursive estimator in a probabilistic way.

2.4.1 Derivation of the General Time Recursive Bayesian Estimator

The key idea of the time recursive Bayesian estimator is to estimate the most likely state x_t of a process given a sorted list of accumulated, independent measurements with $\mathcal{Z}_{0:t} = (\boldsymbol{z}_0, \boldsymbol{z}_1, \boldsymbol{z}_2, ..., \boldsymbol{z}_t)$ up to time step t. Take into account that the state x_t is currently represented by a single, continuous scalar value. Given the observations $\mathcal{Z}_{0:t}$, the goal is to estimate the posterior distribution $p(x_t \mid \mathcal{Z}_{0:t})$. Using Bayes rule the distribution is expressed as

$$p(x_t \mid \mathcal{Z}_{0:t}) \overset{\text{Bayes rule}}{=} \frac{p(\mathcal{Z}_{0:t} \mid x_t)\ p(x_t)}{p(\mathcal{Z}_{0:t})}\ . \tag{2.32}$$

Under the assumption that consecutive observations are conditionally independent $p(\mathcal{Z}_{0:t} \mid x_t) = p(\boldsymbol{z}_t, \mathcal{Z}_{0:t-1} \mid x_t) = p(\boldsymbol{z}_t \mid x_t)\ p(\mathcal{Z}_{0:t-1} \mid x_t)$, and when applying Bayes rule again, we obtain

$$\begin{aligned} p(x_t \mid \mathcal{Z}_{0:t}) &= \frac{p(\boldsymbol{z}_t \mid x_t)\ p(\mathcal{Z}_{0:t-1} \mid x_t)\ p(x_t)}{p(\mathcal{Z}_{0:t})} \\ &= \frac{p(\boldsymbol{z}_t \mid x_t)\ p(x_t \mid \mathcal{Z}_{0:t-1})\ p(\mathcal{Z}_{0:t-1})\ p(x_t)}{p(\mathcal{Z}_{0:t})\ p(x_t)} \\ &= \frac{p(\boldsymbol{z}_t \mid x_t)\ p(x_t \mid \mathcal{Z}_{0:t-1})\ p(\mathcal{Z}_{0:t-1})}{p(\mathcal{Z}_{0:t})}\ . \end{aligned}$$

Using the rule of conditional probability $p(\mathcal{Z}_{0:t}) = p(\boldsymbol{z}_t, \mathcal{Z}_{0:t-1}) = p(\boldsymbol{z}_t \mid \mathcal{Z}_{0:t-1})p(\mathcal{Z}_{0:t-1})$, the general recursive estimation equation is defined as

$$p(x_t \mid \mathcal{Z}_{0:t}) = \frac{p(\boldsymbol{z}_t \mid x_t)\ p(x_t \mid \mathcal{Z}_{0:t-1})}{p(\boldsymbol{z}_t \mid \mathcal{Z}_{0:t-1})}\ . \tag{2.33}$$

Equation (2.33) describes the measurement update step of the time recursive estimation problem and provides the basis for standard estimators such as the Kalman filter or the particle filter. Following the notation of Thrun et al. [2005], the posterior is denoted as the belief of the state, with $bel(x_t) = p(x_t \mid \mathcal{Z}_{0:t})$. The term $p(\boldsymbol{z}_t \mid x_t)$ is called the likelihood function and represents the

probability that $\boldsymbol{z}_t$ is observed if the state x_t is given. The term $p(x_t \mid \mathcal{Z}_{0:t-1})$ describes the prior without incorporating any information about current observations at time step t. The denominator $\eta = p(\boldsymbol{z}_t \mid \mathcal{Z}_{0:t-1})$ is needed for normalization and is estimated using the law of total probability:

$$\eta = p(\boldsymbol{z}_t \mid \mathcal{Z}_{0:t-1}) = \int p(\boldsymbol{z}_t \mid \mathcal{Z}_{0:t-1}, x_t) p(x_t \mid \mathcal{Z}_{0:t-1}) dx_t \ . \tag{2.34}$$

Since it is assumed that the current observation $\boldsymbol{z}_t$ is independent of all previous measurements $\mathcal{Z}_{0:t-1}$ given the current state x_t, the expression is simplified to

$$= \int p(\boldsymbol{z}_t \mid x_t) p(x_t \mid \mathcal{Z}_{0:t-1}) dx_t \ . \tag{2.35}$$

This simplification makes sense, since the current state x_t completely contains the information of all previous observations. Finally, the prediction step, which is denoted as the belief $\overline{bel}(x_t)$, has to be determined by using the law of total probability again:

$$\begin{aligned} \overline{bel}(x_t) = p(x_t \mid \mathcal{Z}_{0:t-1}) &= \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid \mathcal{Z}_{0:t-1}) dx_{t-1} \\ &= \int p(x_t \mid x_{t-1}) bel(x_{t-1}) dx_{t-1}. \end{aligned} \tag{2.36}$$

Equation (2.36) implies the prediction of the state x_t according to the transition model $p(x_t \mid x_{t-1})$ without knowledge of any current observation information. Since we use the previous belief $bel(x_{t-1})$ during the prediction, the recursive structure of the filter is achieved. With this definition (2.33) is expressed by

$$bel(x_t) = \eta \ p(\boldsymbol{z}_t \mid x_t) \ \overline{bel}(x_t). \tag{2.37}$$

If the likelihood function $p(\boldsymbol{z}_t \mid x_t)$, the state transition model $p(x_t \mid x_{t-1})$ and the global prior $p(x_0 \mid \boldsymbol{z}_{-1}) = p(x_0)$ are known, the described time recursive estimation model is uniquely defined. The Basian network of the states and observations is shown in Fig. 2.13. In contrast to MRFs (see Sec. 2.3.1), a Bayesian network is a directed graphical model.

2.4.2 Bayes Estimation of a Binary Hypothesis

A special case of the general time recursive estimation problem is the binary state estimator. The state estimation is reduced to a discrete and binary hypothesis: the state x_t exists or not. The expression $\exists_t$ means that the state exists at time step t, while $\bar{\exists}_t$ denotes the counter hypothesis. With this simplification the prediction step in (2.36) is reduced to

$$P(\exists_t \mid \mathcal{Z}_{0:t-1}) = P(\exists_t \mid \exists_{t-1}) P(\exists_{t-1} \mid \mathcal{Z}_{0:t-1}) + P(\exists_t \mid \bar{\exists}_{t-1}) P(\bar{\exists}_{t-1} \mid \mathcal{Z}_{0:t-1}) \tag{2.38}$$

with $P(\bar{\exists}_{t-1} \mid \mathcal{Z}_{0:t-1}) = \big[1 - P(\exists_{t-1} \mid \mathcal{Z}_{0:t-1})\big]$. Observe, the notation changes from lower to upper case P since the state is discrete. The measurement update of (2.33) is defined by:

$$P(\exists_t \mid \mathcal{Z}_{0:t-1}) = \frac{P(\boldsymbol{z}_t \mid \exists_t) \ P(\exists_t \mid \mathcal{Z}_{0:t-1})}{P(\boldsymbol{z}_t \mid \exists_t) \ P(\exists_t \mid \mathcal{Z}_{0:t-1}) \quad + \quad P(\boldsymbol{z}_t \mid \bar{\exists}_t) \big[1 - P(\exists_t \mid \mathcal{Z}_{0:t-1})\big]} \ . \tag{2.39}$$

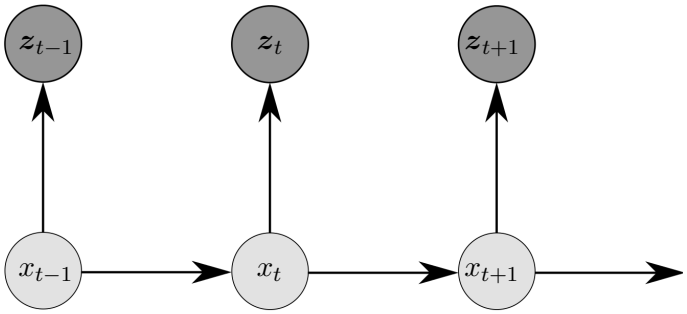


Figure 2.13: Basian network with unknown states x and observations $\boldsymbol{z}$. The unknown states are light gray and the observations are dark gray. The dependencies between them are modeled with *directed* edges. The assumption is made that the observations are independent of each other and that the current state depends only of its preprocessor.

Both (2.38) and (2.39) represent the binary state estimator for each time step. These formulas can also be found in [Altendorfer and Matzka, 2010; Scharwächter, 2013; Muffert et al., 2014]. Due to the fact that the state is discrete, binary, and follows the Markov assumption [Thrun et al., 2005, chapter 2.4.4], the transition probabilities $P(\exists_t \mid \exists_{t-1})$ and $P(\exists_t \mid \bar{\exists}_{t-1})$ are modeled as a two state Markov chain which is represented in Fig. 2.14. The transition probabilities $P\big(\exists_t \mid \exists_{t-1}\big)$ and $P\big(\bar{\exists}_t \mid \bar{\exists}_{t-1}\big)$ influence the inertia of the recursive time filter in an important way: the larger the probability $P\big(\exists_t \mid \exists_{t-1}\big)$, the smaller the probability that the state is changing. To provide a temporally stable state estimation, these probabilities are usually chosen to be large, e.g. 0.95. This means that the transition probabilities from one state to the other, $P\big(\exists_t \mid \bar{\exists}_{t-1}\big)$ and $P\big(\bar{\exists}_t \mid \exists_{t-1}\big)$, should be small, e.g. 0.05.

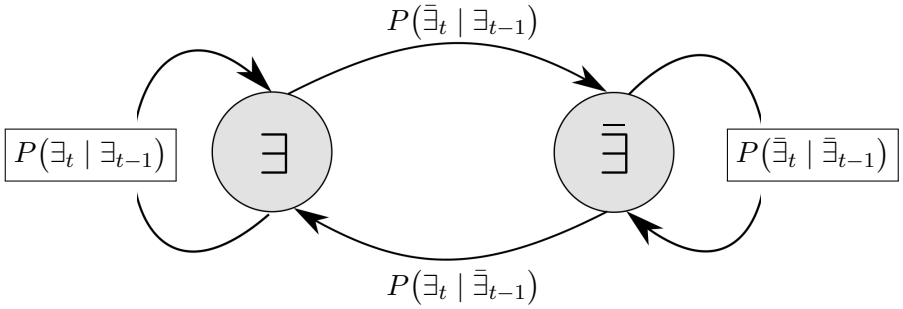


Figure 2.14: The binary Markovian two-state transition model. It is used for the prediction step during recursive existence estimation. For temporal stability the transition probabilities from one state to the other are usually chosen much smaller than transition probabilities back into the same state.

2.5 Occupancy Grid Mapping

In this section the general idea of occupancy grid mapping is presented. Most of the definitions used in this section can also be found in detail in Moravec and Elfes [1985] and Thrun et al. [2005, Chapter 9]. An efficient and widespread representation when describing the continuous space is the projection of the environment into a discrete, regular, 2D grid structure under the planar world assumption. The single grid cells ${}^{a}m_i$ are listed in the final grid map ${}^{a}\mathcal{M} = (..., {}^{a}m_i, ...)$, with $i \in I$ which is in reference to an arbitrary coordinate system S_a. Due to the fact that the order of the grid cells is unique, each element is assigned to coordinates of system S_a. The map elements ${}^{a}m_i$ are frequently described by binary variables, namely if the cells represent free space (0) or occupied areas (1). Following the notation of Thrun et al. [2005], an occupied grid cell is represented by its probability $p(\underline{m}_i = 1) \in [0, 1]$. This map representation is called an occupancy grid map which was introduced by Moravec and Elfes [1985] at the first time (see also Sec. 1.2.4). Depending on the mapping assignment, the grid cell resolution and dimension have to be defined.

The occupancy grid map generation follows the idea of "mapping with known poses" which means to estimate the map posterior $p({}^{a}\mathcal{M}|{}^{a}\mathcal{X}_{0:t}, {}^{s}\mathcal{Z}_{0:t})$ given the pose ${}^{a}\mathcal{X}_{0:t}$ and the sensor readings ${}^{s}\mathcal{Z}_{0:t}$ up to time step t in reference to the sensor system S_s over the complete Map in a probabilistic way. Thrun et al. [2005, chapter 9] give a detailed overview about probabilistic methods for 2D occupancy grid mapping techniques. The advantage of this representation is that the complete captured environment of a moving platform is represented in a probabilistic manner. Thus, a classification into occupied, free, and unknown ($p(m_i) = 0.5$) areas is possible.

To solve the mapping problem in an efficient way, Moravec and Elfes [1985] assume that all grid cells are independent. Because of this assumption, the map posterior $p({}^{a}\mathcal{M}|{}^{a}\mathcal{X}_{0:t}, {}^{s}\mathcal{Z}_{0:t})$ is approximated by the product of the marginals of each individual grid cell:

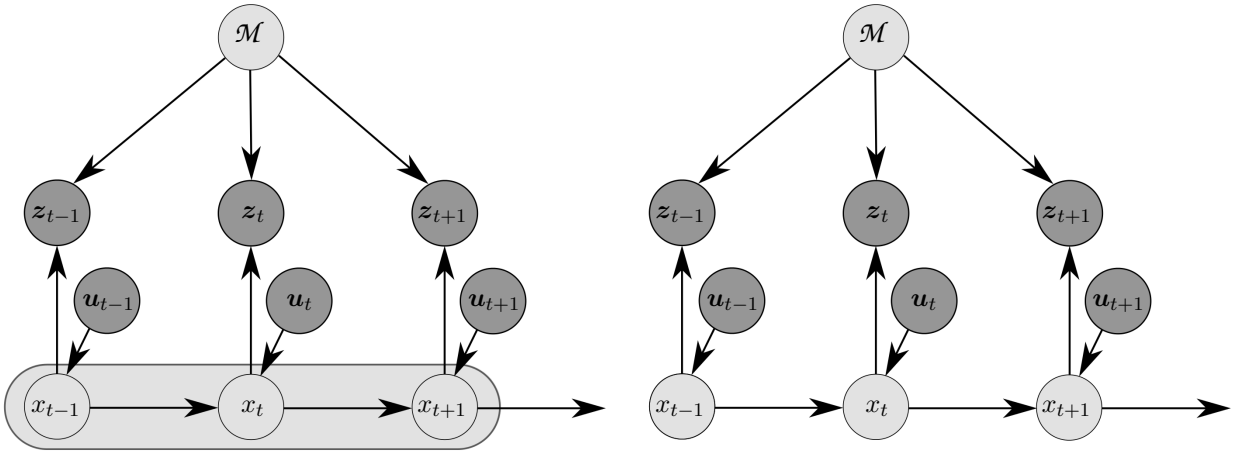
$$p({}^{a}\mathcal{M}|{}^{a}\mathcal{X}_{0:t}, {}^{s}\mathcal{Z}_{0:t}) = \prod_{i=1}^{I} p({}^{a}m_i|{}^{a}\mathcal{X}_{0:t}, {}^{s}\mathcal{Z}_{0:t}) \ . \tag{2.40}$$

This factorization allows the estimation of the occupancy probability $p({}^{a}m_i|{}^{a}\mathcal{X}_{0:t}, {}^{s}\mathcal{Z}_{0:t})$ for each grid cell ${}^{a}m_i$ individually over the total number of grid cells I in the map ${}^{a}\mathcal{M}$. There exist a lot of approaches to model $p({}^{a}m_i|{}^{a}\mathcal{X}_{0:t}, {}^{s}\mathcal{Z}_{0:t})$, but in the most cases an inverse sensor model [Thrun et al., 2005, p. 288] is used which leads to a recursive binary Bayes filter similar to the described approaches in Sec. 2.4. From now, we neglect the indices of the coordinate systems which is a frequent practice in robotics and define the map posterior by $p(\mathcal{M} \mid \mathcal{X}_{0:t}, \mathcal{Z}_{0:t})$.

2.6 The general SLAM Problem

The SLAM problem deals with the simultaneous estimation of a map $\mathcal{M}$ and the estimation of the best trajectory of a mobile platform $\mathcal{X}_{0:t}$ given only the control information $\mathcal{U}_{0:t-1}$ and sensor readings $\mathcal{Z}_{0:t}$. The trajectory is defined by the set $\mathcal{X}_{0:t} = \{\boldsymbol{x}_t\}$ where each vector $\boldsymbol{x}_t$ represents a pose in 2D or 3D. In general, the representation of the map $\mathcal{M}$ is arbitrary, but, in most cases, landmark based or grid based maps (see Sec. 2.5) are used [Grisetti et al., 2007; Kaess et al., 2008]. SLAM is one of the most fundamental problems in robotic applications [Thrun et al., 2005],

since neither a consistent map is given to solve the localization problem nor a certain trajectory is available to build correct and precise maps. Thus, the SLAM problem can be understood as a "chicken and egg problem" because mutual dependencies between the map and the pose estimation are given. As described in [Thrun et al., 2005, chapter 10], the SLAM problem can be divided into two different forms, namely in the full SLAM problem and the on-line SLAM problem. The full SLAM problem deals with the estimation of the joint posterior over all poses of the trajectory $p(\mathcal{X}_{0:t}, \mathcal{M} \mid \mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1})$. The on-line SLAM problem estimates the posterior only over the current pose $\boldsymbol{x}_t$ along with the map estimation $p(\boldsymbol{x}_t, \mathcal{M} \mid \mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1})$. Figure 2.15 shows the graphical models of both the full and the on-line SLAM problem. In the following section we introduce the Rao-Blackwellized particle filter (RBPF) which is used in this thesis to solve the on-line SLAM problem for grid maps.



(a) The full SLAM problem

(b) The on-line SLAM problem

Figure 2.15: Graphical models of (a) the full SLAM problem and (b) the on-line SLAM problem, referred to Thrun et al. [2005]. In case (a) the posterior over the complete path of the trajectory is estimated. It is illustrated by the bright gray box around the pose nodes $\boldsymbol{x}$. As well as the pose nodes the map $\mathcal{M}$ is unknown, too. In case (b) only the posterior of the current pose node is estimated. This decomposition leads to an incremental, on-line approach. In both cases, the full and on-line SLAM problem, the sensor readings $\mathcal{Z}_{0:t}$ and the control elements $\mathcal{U}_{0:t-1}$ of the robot are given.

2.7 Grid based SLAM with Rao-Blackwellized Particle Filters

In this section we describe the Rao-Blackwellized particle filters (RBPFs) for grid based maps to solve the SLAM problem. At the beginning, we introduce particle filters (see Sec. 2.7.1). In Sec. 2.7.2 we introduce the RBPF which is a specific realization of the SIR particle filter (see Sec. 2.7.1.2).

2.7.1 Particle Filters

Particle filters approximate an arbitrary posterior distribution by a finite number of samples in a recursive way. Particle filters allow us to represent multi modal distributions which is an advantage to other filters like Kalman filters [Welch and Bishop, 1995]. We refer to [Stachniss, 2006, chapter 2.1] and [Thrun et al., 2005, chapter 4.3] for more detailed information.

2.7.1.1 Idea and General Description of Particle Filters

In particle filter approaches the posterior distribution of a dynamic state given a set of data information is described by a finite set of possible state hypotheses called *particles* $\boldsymbol{p}_t^{(i)}$. The set $\mathcal{P}_t$ of all I particles is denoted by

$$\mathcal{P}_t := \{\boldsymbol{p}_t^{(i)}\}, \text{ with } \boldsymbol{p}_t^{(i)} = (x_t^{(i)}, w_t^{(i)}),\ 1 \leq i \leq I\ . \tag{2.41}$$

Each particle $\boldsymbol{p}_t^{(i)}$ represents a concrete realization $x_t^{(i)}$ of the true state $\tilde{x}_t$ at the time step t with its weight $w_t^{(i)}$. To bridge the gap between the definition of Sec. 2.4.1 and the current section, the particle filter set should be proportional to the posterior distribution:

$$x_t^{(i)} \sim p(x_t | \mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1}). \tag{2.42}$$

In addition to (2.33) we introduce the control set $\mathcal{U}_{0:t-1} = (\boldsymbol{u}_0, \boldsymbol{u}_1, \boldsymbol{u}_2, ..., \boldsymbol{u}_{t-1})$ up to the previous time step $t-1$ to the posterior distribution. The controls represent e.g. the relative orientation and speed of a moving platform or a robot (see Sec. 2.1). Along with the measurements $\mathcal{Z}_{0:t}$, the controls are assumed to be given.

Just like the described recursive Bayesian estimator (see Sec. 2.4.1), the particle filter generates the current particle set $\mathcal{P}_t$ recursively from the previous set $\mathcal{P}_{t-1}$. This approach is known as the *Sampling Importance Resampling* (SIR) particle filter [Thrun et al., 2005, chapter 4.3] which is described in the following. The general derivation of the particle filter algorithm is discussed in the Appendix A. An implementation of the described SIR particle filter is listed in Algorithm 2.

2.7.1.2 The Sampling Importance Resampling (SIR) Particle Filter

Sampling. Sampling is the first step of the SIR particle filter which means that the next particle set $\bar{\mathcal{P}}_t$ is generated from the previous set $\mathcal{P}_{t-1}$ (see also Appendix A). This step corresponds to the prediction step of the general Bayesian estimator which is discussed in Sec. 2.4.1: the new generated particle set is the filter's representation of the $\overline{bel}(x_t)$ [Thrun et al., 2005, p. 99]. By following the definitions of Appendix A, we use the proposal distribution $\pi(.)$ for sampling which is in this case defined by the prediction distribution. Thus, we get the next states by sampling from this distribution:

$$\bar{x}_t^{(i)} \sim p(x_t | x_{0:t-1}^{(i)}, \mathcal{Z}_{0:t-1}, \mathcal{U}_{0:t-1}). \tag{2.43}$$

Observe, that, similar to (2.36), current observations are excluded in this step. In the initialization step $t = 0$ we assume prior world assumptions of the state to generate the first particle set. For instance, for the task of localization, particles can be drawn proportional to a uniform distribution over the 2D or 3D space of a given map. But in this context, a coarse GPS position is used more often.

Importance Weighting. Importance weighting is the next step in which for each particle of $\bar{\mathcal{P}}_t$ an importance weight $\bar{w}_t^{(i)}$ is computed. Let $\wp(x_{0:t}^{(i)}|\mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1})$ be the desired distribution of all states $x_{0:t}^{(i)}$ over the complete time period. This distribution is called the *target distribution* and we cannot sample from this function directly, since the target distribution can not be represented by a parametric form (see Appendix A). Therefore, we draw samples from the *proposal distribution* $\pi(x_{0:t}^{(i)}|\mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1})$ and weight each particle individually to approximate the target distribution $\wp(.)$. By following the definition of Stachniss [2006, p. 31 ff] and the derivation in Appendix A, each weight $\bar{w}_t^{(i)}$ is estimated by

$$\bar{w}_t^{(i)} = \eta \, \frac{\wp(x_{0:t}^{(i)}|\mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1})}{\pi(x_{0:t}^{(i)}|\mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1})} \, . \tag{2.44}$$

For the estimation of the weights it is indispensable that the target distribution can be analyzed point-wise. Equation (2.44) is normalized by η to guarantee that the sum over all weights is 1. Weight estimation incorporates current measurements in the particle filtering process. Figure 2.16 shows an example of the sampling and weighting step to approximate an arbitrary target distribution.

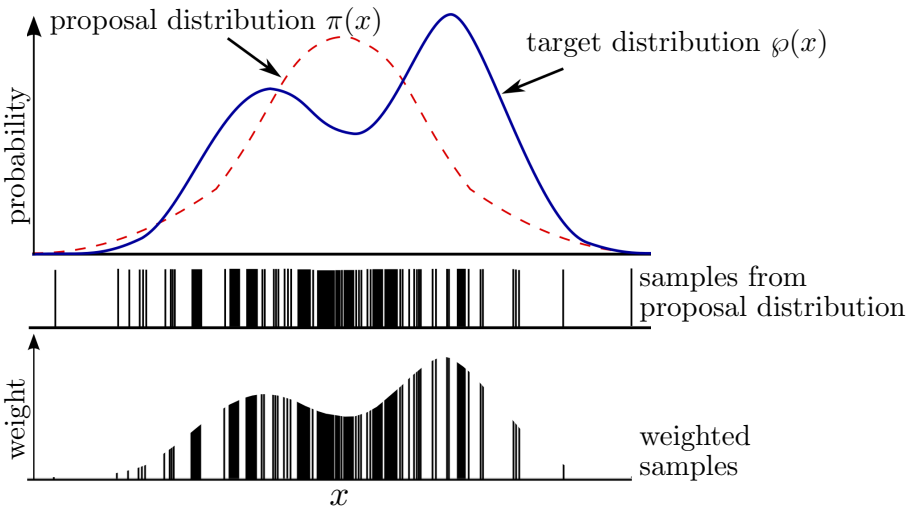


Figure 2.16: Approximation of a arbitrary target distribution $\wp(x)$ (blue curve) by weighted samples. Samples are drawn from a proposal distribution $\pi(x)$ (red dotted curve). The weighted samples are represented by vertical lines at the bottom. The visualization of importance resampling was taken from [Thrun et al., 2005, p. 447].

Resampling. Resampling is the final step of the SIR particle filter which transforms the temporary particle set $\bar{\mathcal{P}}_t$ into the new set $\mathcal{P}_t$. The probability of drawing samples from $\bar{\mathcal{P}}_t$ is proportional to the importance weights $\bar{w}_t^{(i)}$. This means, that the distribution (but not the number) of particles is changing by incorporating the weights. Analogous to the measurement update from Sec. 2.4.1,

the resampling step represents, approximately, the posterior $bel(x_t)$. After the resampling step, the weights correspond to the uniform distribution with $\bar{w}_t^{(i)} = 1/M$.

In more general words and as mentioned in [Thrun et al., 2005, p. 100], the resampling step is a probabilistic implementation of Darvin's *survival of the fittest*. Particles with higher weights have a high chance to "survive" and "breed", whereas particles with lower weights are "in danger of extinction". This strategy is, in general, applicable since the filter prefer "good" particles which represents "good" state hypotheses. On the other hand, it can be problematic to replace apparently "bad" samples too early. In the literature [van der Merwe et al., 2000; Doucet et al., 2001; Grisetti et al., 2007], this phenomenon is called the particle depletion or particle deprivation problem. Thrun et al. [2005, chapter 4.3.4] give detailed information and strategies to reduce the risk of particle depletion by applying low-variance resampling.

Algorithm 2: Algorithm of the SIR particle filter.

Input: The previous sample Set $\mathcal{P}_{t-1}$ and the data information $\mathcal{Z}_{0:t}$ and $\mathcal{U}_{0:t-1}$. In the initialization step, prior world assumptions over the state are assumed.

Output: The next sample Set $\mathcal{P}_t$.

$\mathcal{P}_t = \bar{\mathcal{P}}_t = \emptyset$ // Initialize the new particle set.

for *i = 1 to M* **do**

  sample:

  $\bar{x}_t^{(i)} \sim p(x_t | x_{0:t-1}^{(i)}, \mathcal{Z}_{0:t-1}, \mathcal{U}_{0:t-1})$ // The sampling step:draw next generation of particles(prediction).

  weighting:

  $\bar{w}_t^{(i)} = \eta \frac{p(x_{0:t}^{(i)} | \mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1})}{\pi(x_{0:t}^{(i)} | \mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1})}$ // Importance weighting:incorporating current measurements.

  $\bar{\mathcal{P}}_t = \bar{\mathcal{P}}_t + [\bar{x}_t^{(i)}, \bar{w}_t^{(i)}]^T$ // Update temporary particle filter set.

for *i = 1 to M* **do**

  resample:

  draw $x_t^{(i)}$ from $\bar{\mathcal{P}}_t$ with probability $\propto \bar{w}_t^{(i)}$ // Resampling:change the particle distribution.

  $\mathcal{P}_t = \mathcal{P}_t + (x_t^{(i)}, \frac{1}{M})$ // Update final particle filter set.

return $\mathcal{P}_t$

2.7.2 The Rao-Blackwellized Particle Filter

Rao-Blackwellized particle filters (RBPFs) [Murphy, 1999; Doucet et al., 2000] are presented in this section which are used to solve the on-line SLAM problem for grid maps in this thesis. The "trick" of this approach is to separate the estimation of the trajectory from the estimation of the map:

$$p(\mathcal{X}_{0:t}, \mathcal{M} | \mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1}) \overset{\text{Product rule.}}{=} \underbrace{p(\mathcal{X}_{0:t} | \mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1})}_{\text{pose posterior}} \underbrace{p(\mathcal{M} | \mathcal{X}_{0:t}, \mathcal{Z}_{0:t})}_{\text{map posterior}} . \tag{2.45}$$

The splitting of the overall posterior into a pose posterior and a map posterior results in a sequential estimation of the trajectory and the map. Here, take into account that in (2.45) the assumption was made that the map is independent of the control information. The estimation of the map posterior is computed in a closed form by using "mapping with known poses" strategies. Since the map posterior can be estimated analytically, only the estimation of the pose posterior is solved by a sample procedure. Otherwise, the sampling would be carried out over the whole possible map space which would be intractable.

We follow the idea of Doucet et al. [2001] who used the recursive SIR particle filter approach (see Sec. 2.7.1.2) to determine the best possible trajectory. This means that each particle has its own trajectory, and, consequently, each particle has its own individual map. Each map is built separately by using the particle's individual trajectory and the observations which are the same for all particles. In mathematical terms, the sample set for the RBPF is defined by:

$$\mathcal{P}_t := \{(\boldsymbol{x}_t^{(i)}, \mathcal{M}_t^{(i)}, w_t^{(i)})\}, \text{ with } 1 \leq i \leq I. \tag{2.46}$$

Sampling. Using RBPFs, it is common practice to draw the samples from the probabilistic odometry motion model $p(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}^{(i)}, \mathcal{U}_{0:t-1})$. Compared to (2.43), no observations $\mathcal{Z}_{0:t-1}$ are taken into account which leads to:

$$\boldsymbol{x}_t^{(i)} \sim p(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}^{(i)}, \mathcal{U}_{0:t-1}) . \tag{2.47}$$

Importance Weighting. If one analyzes (2.44), the weight estimation would be very inefficient since the complete trajectory is analyzed at each time step. To achieve a recursive weight estimation and follow up the on-line SLAM idea, Doucet et al. [2001] derived the recursive weight estimation which is used in many particle filter applications [Dellaert et al., 1999; Montemerlo et al., 2002; Grisetti et al., 2007]. By following Doucet et al. [2001], it can be shown that the weight estimation in (2.44) is proportional to the product of the observation model $p(\boldsymbol{z}_t | \mathcal{M}_{t-1}^{(i)}, \boldsymbol{x}_t^{(i)})$ with the previous weights $\bar{w}_{t-1}^{(i)}$ under the assumption, that the proposal distribution $\pi(\boldsymbol{x}_t^{(i)} \mid \boldsymbol{x}_{1:t-1}^{(i)}, \mathcal{Z}_{0:t}, \mathcal{U}_{0:t-1})$ is replaced by the probabilistic odometry motion model $p(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}^{(i)}, \boldsymbol{u}_{t-1})$. Then, the weights are estimated by:

$$\bar{w}_t^{(i)} \propto p(\boldsymbol{z}_t | \mathcal{M}_{t-1}^{(i)}, \boldsymbol{x}_t^{(i)}) \ \bar{w}_{t-1}^{(i)} . \tag{2.48}$$

The observation model $p(\boldsymbol{z}_t|\mathcal{M}_{t-1}^{(i)}, \boldsymbol{x}_t^{(i)})$ represents the posterior of the observations given the previous map $\mathcal{M}_{t-1}^{(i)}$ and the current pose $\boldsymbol{x}_t^{(i)}$ of the individual particles i. For the computation of the observation model, e.g. beam sensor models [Thrun et al., 2005, chapter 6.3], map matching models [Schröter et al., 2007] or iterative closest point (ICP) algorithms [Besl and McKay, 1992] often are used.

The described recursive sampling and weighting procedure is equivalent to the Monte Carlo localization (MCL) problem. In contrast to the MCL, where one map is already given, RBPFs produce I different maps on-line. In connection with the described general SLAM problem (see Sec. 2.6) it makes sense that the RBPF contains a localization procedure.

In conclusion, weight estimation answers the question, how well does the current observation vector match to the individual global occupancy grid maps. The better the observations fit to the map, the larger the weights. Figure 2.17 shows examples of different particles with their individual maps and trajectories using RBPFs. These examples were taken from [Stachniss, 2006, p. 122].

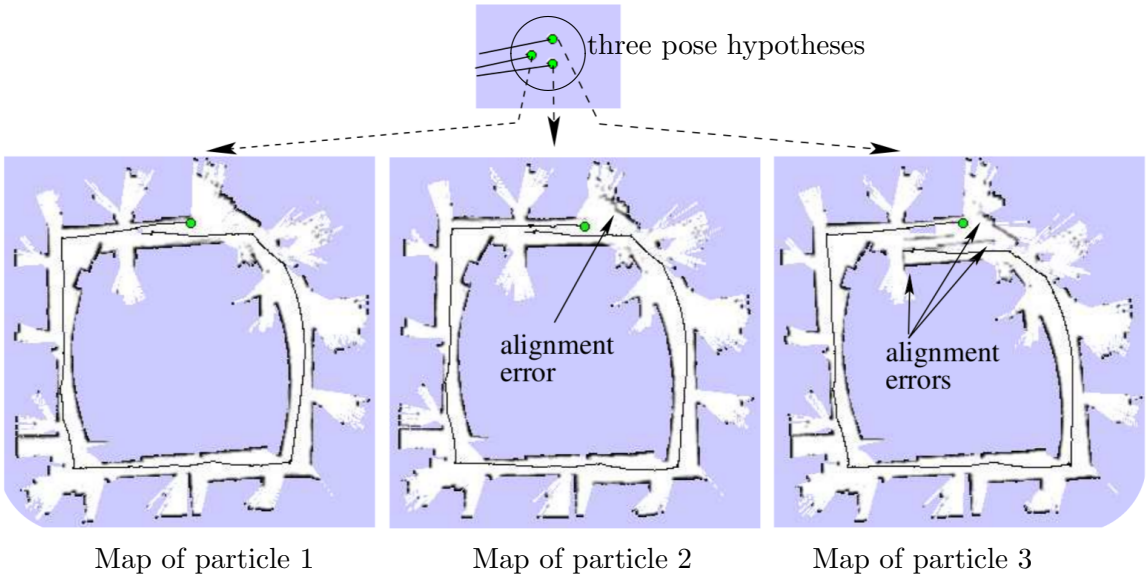


Figure 2.17: The idea of the RBPF for grid based maps. Each particle produces its own trajectory and, consequently, its own map. The green point represents the current position of the robot for three different particles. In contrast to the map of particle 1, the maps of the other two particles include alignment errors. This effect should be reflected in the particle weights: from particle 1 to particle 3, the weights decrease significantly. In the resampling step particle 3 can be replaced by particle 1. The images were taken from [Stachniss, 2006, p. 122].

Resampling. As described in Sec. 2.7.1.2, the resampling is the last step of any particle filters to transform the previous distribution of the particles into the new particle filter set by incorporating the estimated weights $\hat{w}^i$. By following the basic idea of the SIR particle filter, this step is carried out at each time step.

However, this results in a very inefficient implementation and can lead to the previous mentioned particle deprivation problem [van der Merwe et al., 2000; Doucet et al., 2001; Grisetti et al., 2007]. Thus, it is indispensable to find a criterion when a resampling step is carried out. Lui [1996] introduced the effective sample size number N_{eff} for particle filters which gives feedback about how well the current particle distribution fits to the target distribution. With regard to RBPFs this quantity is estimated by

$$N_{\text{eff}} = \left[\sum_{i=1}^{N} \left(\hat{w}^i \right)^2 \right]^{-1} , \tag{2.49}$$

which allows us to control the resampling step. The following statements makes this clear. If N_{eff} is high (the maximum value is N), the particle distribution represents a good approximation of the target (true) distribution and the weights are nearly uniformly distributed. A resampling step is not necessary. If N_{eff} is low (the minimum value is 1), the approximation of the target distribution is bad. The variance of the importance weights is high. A resampling step is unavoidable. Thus, the resampling step is carried out only if the effective sample size number drops below a threshold, e.g. $N_{\text{eff}} < N/2$.

Map Generation. On the basis of (2.45) the map posterior $p({}^a\mathcal{M}_{0:t} \mid {}^a\mathcal{X}_{0:t}, {}^s\mathcal{Z}_{0:t})$ is estimated using "mapping with known poses" for grid maps. The background is previously described in Sec. 2.5. A novel mapping approach is presented in Sec. 3.

2.8 Feature based SLAM with Graphs

Next to recursive, grid based SLAM approaches like RBPFs, *en bloc* SLAM approaches based on graphs are very common in the field of robotics [Thrun and Montemerlo, 2006; Ziegler et al., 2014; Lategahn and Stiller, 2014]. Graph SLAM approaches follow the idea of the full SLAM problem (see Sec. 2.6) where the complete pose trajectory and the map, represented as a set of landmarks, is estimated *en bloc* via one optimization step. These approaches revise estimates over the entire history which results in general in more accurate and consistent solutions. Nevertheless, the approach keeps by definition the whole observation and control history alive which results in high processing time and computing power in large scale environments. Figure 2.18 shows the idea of the graph SLAM approach which represents poses and observed landmarks as vertices and constraints as edges. In the following, we define the optimization problem and utilize the notation of [Thrun and Montemerlo, 2006].

Successive poses $\boldsymbol{x}_{t-1}$ and $\boldsymbol{x}_t$ are connected via edges which represent the information constraint between the unknowns. The information constraint is defined by a non-linear (probabilistic) motion model $g(\boldsymbol{u}_{t-1}, \boldsymbol{x}_t - 1)$ governed by the control vector $\boldsymbol{u}_{t-1}$ with its inverse covariance matrix Ω^m_{t-1} of the motion noise. Edges between poses $\boldsymbol{x}_t$ and landmarks $m_{t,i}$ are defined by the observation model $h(\boldsymbol{x}_t, m_{t,i})$ with the inverse covariance matrix $\Omega^o_{t,i}$ of the observation noise. We also have to consider the anchoring constraint $\boldsymbol{x}_0^\mathsf{T} \Omega_0 \boldsymbol{x}_0$ to anchor the first position $\boldsymbol{x}_0$ of the vehicle in a global map.

Over the whole graph $G(\mathcal{X}_{0:t}, \mathcal{M})$, the sum of all non-linear constraints can be formulated as a non-linear least squares (NLS) problem

$$\begin{aligned} G(\mathcal{X}_{0:t}, \mathcal{M}) = \boldsymbol{x}_0^{\mathsf{T}} \Omega_0 \boldsymbol{x}_0 \\ + \sum_{t=1}^{T} [\boldsymbol{x}_t - g(\boldsymbol{u}_{t-1}, \boldsymbol{x}_{t-1})]^T \Omega_{t-1}^m [\boldsymbol{x}_t - g(\boldsymbol{u}_{t-1}, \boldsymbol{x}_{t-1})] \\ + \sum_{t=1}^{T} \sum_{i=1}^{I} [\boldsymbol{z}_{t,i} - h(\boldsymbol{x}_t, m_{t,i})]^T \Omega_{t,i}^o [\boldsymbol{z}_{t,i} - h(\boldsymbol{x}_t, m_{t,i})] \ , \end{aligned} \tag{2.50}$$

to find the minimum

$$\mathcal{X}_{0:t}^*, \mathcal{M}^* = \underset{\mathcal{X}, \mathcal{M}}{\operatorname{argmin}} \ G(\mathcal{X}, \mathcal{M}) \ , \tag{2.51}$$

where $\mathcal{X}_{0:t}^*$ and $\mathcal{M}^*$ are the most likely trajectory and most likely map, respectively. NLS problems like (2.50) are usually solved using Gauss-Newton (GN) or Levenberg-Marquardt (LM) solvers, as already stated in Sec. 1.2.6. Frameworks like *g2o* [Kümmerle et al., 2011] exploits the sparse connectivity structure of the graphs to solve (2.50) efficiently. In Sec. 6.2 the graph SLAM idea is used to estimate reference data for evaluation purposes.

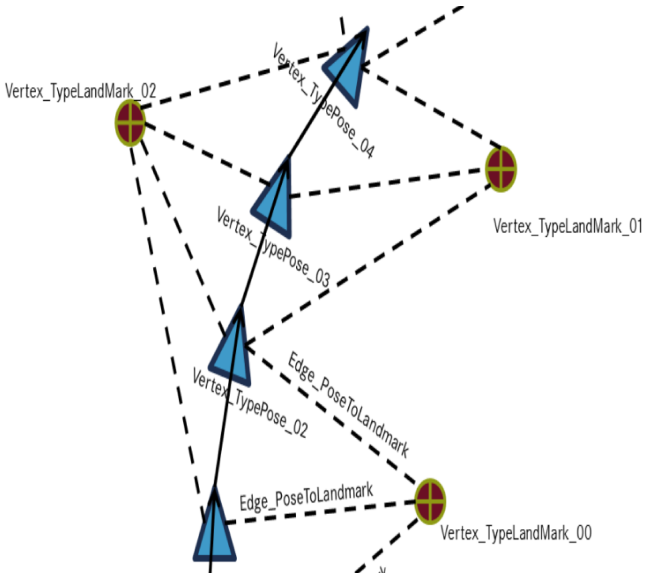


Figure 2.18: The SLAM problem as a graph representation [Thrun and Montemerlo, 2006]. The blue triangles represent the different poses of the ego vehicle and the red-green circles represent the observed landmarks. Both, landmarks and poses, represent the unknowns. The connections between the poses represent the odometry constraints, the links between the pose and landmarks are the observations. The namespace of the edges and nodes was taken from Kümmerle et al. [2011].

2.9 Evaluation Criteria

In this section evaluation criteria are described which are applied in Sec. 5 and Sec. 6 of this thesis. We present the *empirical accuracy* which result from the comparison of estimated test samples and ground truth (GT) data. The term *classification accuracy* is also defined which is often used in machine learning or computer vision.

2.9.1 Empirical Accuracy

The term empirical accuracy means the comparison of estimated test samples and their precision with ground truth or reference data. We distinguish between ground truth and reference data since ground truth has a nearly infinitely high precision and reference data is "only" several orders of magnitude more precise than the estimated results. In this context, precision means that the covariance information of both the test samples and ground truth/reference data are available. We assume that we have access to ground truth or reference data and define these values by the vector $\boldsymbol{s}_r$ with their covariance matrices $\Sigma_{s_r s_r}$. The samples $\boldsymbol{s}_t$ with their covariance matrices $\Sigma_{s_t s_t}$ must be consistent with $\boldsymbol{s}_r$. Under these conditions, the error vector $\boldsymbol{e}$ can estimated easily:

$$\boldsymbol{e} = \boldsymbol{s}_t - \boldsymbol{s}_r \ . \tag{2.52}$$

These differences can be analyzed, e.g. by plotting a histogram or their max-min-values. The mean absolute error (MAE)

$$\text{MAE} = \frac{1}{I} \sum_{i=1}^{I} \mid e_i \mid \tag{2.53}$$

is a common measure to give an overall accuracy for the test set. In order to use the information of ground truth and estimated test samples, the precision is partially taking into account. Based on the variances $\sigma^2_{s_{t_i}}$ and $\sigma^2_{s_{r_i}}$ the weights

$$w_i = \frac{1}{\sigma^2_{e_i}} \tag{2.54}$$

are estimated with

$$\sigma^2_{e_i} = \sigma^2_{s_{t_i}} \ + \ \sigma^2_{s_{r_i}}. \tag{2.55}$$

The weights can be used in a second measure, the weighted mean absolute error (WMAE) which is defined by:

$$\text{WMAE} = \frac{1}{\sum_{i=1}^{I} w_i} \sum_{i=1}^{I} w_i \mid e_i \mid \ . \tag{2.56}$$

Observe, the covariances are not taken into account in this measure, why the WMAE is also suboptimal.

2.9.2 Classification Accuracy

The performance of classification or segmentation results are often assessed on their detection rates or labeling errors. Since we estimate these rates with ground truth (GT) or reference data, we also address these detection rates as accuracies. The labeling errors are represented by the confusion matrix. In this thesis, we are also interested in evaluating maps with the help of this accuracy assessment. Therefore, we estimate the detection rates for both occupied areas (obstacles) and free space areas with the help of generated ground truth maps. Tab. 2.1 shows the pattern of the confusion matrix which is used during this thesis. The table contains the four elements of the 2×2 confusion matrices of three experiments as rows. In our case false negatives (FN) are defined as areas which are falsely predicted as free space, but are GT obstacles. Therefore, the false positives (FP) are areas which are falsely predicted as obstacles, but are actual free space areas. The true negatives (TN) are areas where estimated free space and GT free space is aligned, and the true positives (TP) are areas where estimated obstacles and GT obstacles are the same. Based on these definitions we estimate the detection rates in percent. The formulas are also included in Tab. 2.1.

Table 2.1: Pattern of the used confusion matrix during this thesis. It represents the estimated classification accuracies for estimated obstacles and free space (Est.) compared to ground truth (GT) obstacles and ground truth free space for different configurations. Based on the true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN) rates in percent can be estimated.

	GT obstacles [%]		GT free space [%]	
	Est. obstacles	Est. free space	Est. obstacles	Est. free space
Configuration	$\frac{TP}{TP+FN}$	$\frac{FN}{TP+FN}$	$\frac{TN}{TN+FP}$	$\frac{FP}{TN+FP}$

Chapter 3

Concept of an Incremental Mapping Approach with Markov Random Fields using Known Poses

In this chapter we present a novel 2D incremental occupancy grid mapping approach which is based on MRFs to model the dependencies between neighboring grid cells. We use the super pixel representation Stixel World as input data. This representation is based on disparity images which are captured from a stereo camera system mounted on a moving earth-bounded platform. The following section gives an overview of the entire chapter which includes the overall system overview of the approach, as well as the organization of this chapter.

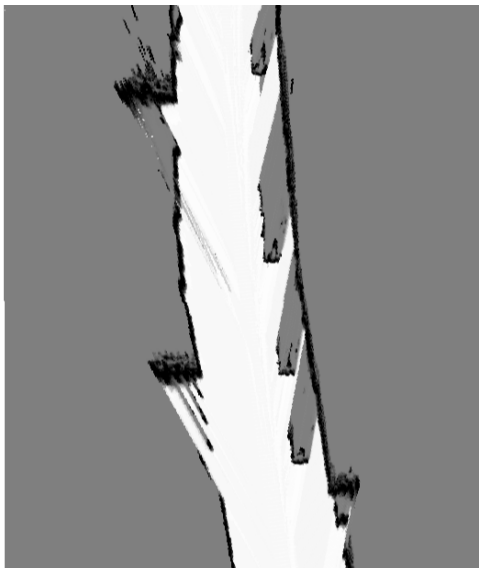
3.1 Overview

In this section we give a brief overview of the complete concept. At a glance, we present the input data, the objective of the mapping approach, the system overview, and the organization of this chapter.

Input Data. As input data we rely on the super pixel representation Stixel World (see Sec. 2.2.3) which is generated from dense disparity images recorded by our test vehicle. In combination with optical flow information and tracking algorithms each Stixel is segmented into a static or dynamic obstacle. We also use an IMU which delivers steering and velocity information to estimate the global pose of the vehicle. In this chapter, we assume the vehicle odometry is certain.

Objective. The key objective of the new approach is to learn an occupancy grid map of the captured environment in an incremental way, and also model the dependencies between neighboring grid cells explicitly. To the best of our knowledge, this is the first time that MRFs are used in combination with occupancy grid maps to model dependencies between neighboring grid cells. The challenge is to develop an incremental mapping approach which handles these dependencies and to estimate marginal probabilities for each different grid cell in an efficient way.

Furthermore, the super pixel representation Stixel World is used as input data which is also a key aspect of this thesis. Using Stixels for occupancy grid mapping approaches was applied in Muffert et al. [2013] and Muffert et al. [2014] for the first time and allows us to neglect dynamic objects during mapping. Both, the modeling of dependencies of neighboring grid cells and the use of the Stixel World will result in more robust and more precise occupancy grid maps. In the context of the next generation of driver assistance systems the novel approach must be real-time capable. Figure 3.1 shows a sample map using the new mapping approach which is presented in the following sections. We also assume that the pose of the vehicle is known at this point.



(a) Occupancy grid map example

(b) original gray scale image, $t = 222$

(c) original gray scale image, $t = 270$

Figure 3.1: Example of our novel occupancy grid mapping approach using MRFs (a). The images (b) and (c) were taken to estimate the disparity images. They also help to understand the resulting grid map. As one can see, parking cars, gates, buildings, and also poles are mapped precisely.

System Overview and Organization of the Chapter. The system overview of the novel grid mapping approach is shown in Fig. 3.2. We describe the preprocessing steps in Sec. 3.2 which includes the sensor setup, data acquisition (Sec. 3.2.1), the definition of the coordinate systems (Sec. 3.2.2), and the definition of our input data (Sec. 3.2.3). Sec. 3.3 presents the optimization formulation of our proposal in a probabilistic fashion. Sec. 3.4 deals with the definition of the unary terms which includes the derivation of the time recursive structure (Sec. 3.4.1), the definition of the measurement model (Sec. 3.4.2), as well as a detailed description of the prediction step (Sec. 3.4.3). In Sec. 3.5 the binary terms are introduced which model the dependencies between neighboring grid cells. The incremental map generation is presented in Sec. 3.6, including the definition of the used graph structure (Sec. 3.6.1), as well as the estimation of marginal probabilities via dynamic graph cuts (Sec. 3.6.2). At the end of this chapter, we present the pseudo code and the runtime behavior of the overall mapping approach. We finalize the chapter by presenting first results.

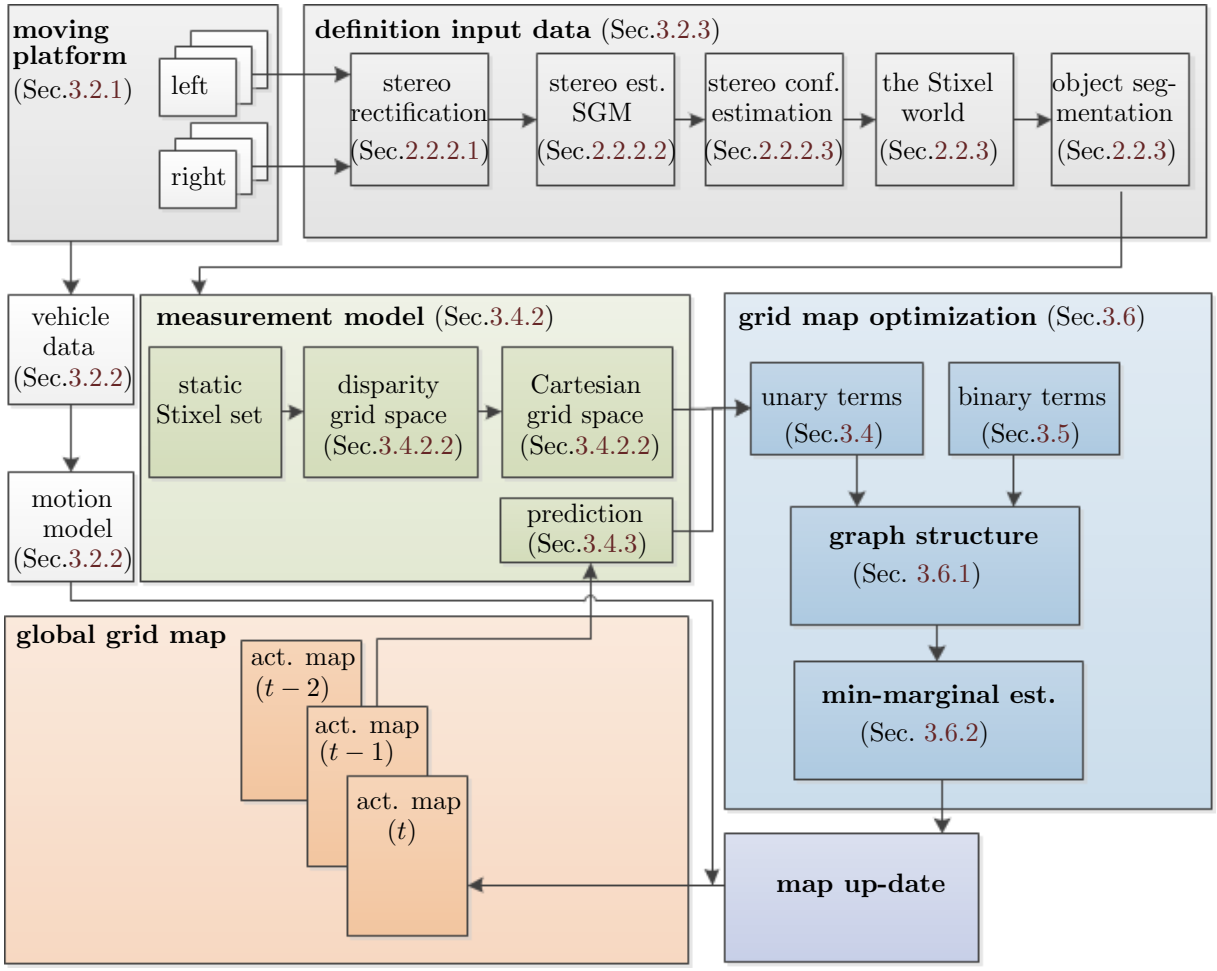


Figure 3.2: Overview of the realized mapping approach. The figure shows the main processing steps in different colors. Based on data acquisition of the moving platform, rectified stereo image sequences are estimated at each time step t. With the help of SGM, disparity images are generated to be used for the estimation of the Stixel World. The Stixel World is segmented into static and dynamic environment information. We define our measurement model based on the static Stixel set. In the grid mapping optimization process the unary and binary terms are used to model the graph structure of the MRF. After the estimation of the min-marginals, the global grid map update is carried out using a corresponding vehicle model. It is assumed that the pose of the vehicle is correct in this chapter.

3.2 Preprocessing Steps

In this section the preprocessing steps are described which include the sensor setup (Sec. 3.2.1), data acquisition (Sec. 3.2.2), the Stixel World generation, and the formal definition of the observations which are used in the mapping approach (Sec. 3.2.3).

3.2.1 Sensor Setup and Data Acquisition

We assume that the environment in front of the moving platform is captured by stereo vision. From now on, we define the moving platform as our vehicle. The stereo camera rig is mounted behind the windshield and points to the forward driving direction shown in Fig. 3.3. We assume that the stereo system fulfills the normal stereo case which is previously described in Sec. 2.2.2.1.

It is required that the intrinsic calibration parameters and the orientation of the stereo camera system in reference to the vehicle coordinate system are stable during data acquisition. The stereo vision system shown in Fig. 3.3 produces synchronous image sequences with the resolution of 1024×440 pel and runs with a constant frame rate of 25 Hz.

(a) Stereo camera rig (b) Test vehicle *S 500 Intelligent Drive*

Figure 3.3: Stereo camera rig (a) and test vehicle (b). Fig (a) shows the stereo camera system behind the windshield. The cameras are running synchronously with a constant frame rate of 25 Hz. Some of the image sequences used in this thesis were captured with the *S 500 Intelligent Drive* [Franke et al., 2013].

3.2.2 Coordinate Systems and Control Information

Coordinate Systems. Figure 3.4 shows the relevant coordinate systems applied in this concept. The 3D camera coordinate system S_c has its origin ${}^c\boldsymbol{O}$ in the projection center of the left camera as previous described in Sec. 2.2.2.1. In contrast to the projective camera model introduced in Sec. 2.2.1, the positive ${}^cz$-axis points in the direction of the image plane. Hence, it is a left handled system. The origin ${}^r\boldsymbol{O}$ of the vehicle coordinate system S_r is in the middle of the rear axle which in turn defines the ${}^ry$-axis. The positive ${}^rx$-axis points in the driving direction, the positive ${}^rz$-axis is aligned to the sky. We use the index r since the test vehicle can also be defined as a robot.

In consideration of the fact that the desired occupancy grid map is represented in 2D space, we have to take the projection from the 3D vehicle coordinate system into the 2D global map coordinate system S_w into account. The definition of its origin and orientation is, in general, arbitrary.

Control Information. In this chapter, the relative odometry information is derived from IMUs (see Sec. 2.1). The odometry information describes the movement of the vehicle between consecutive

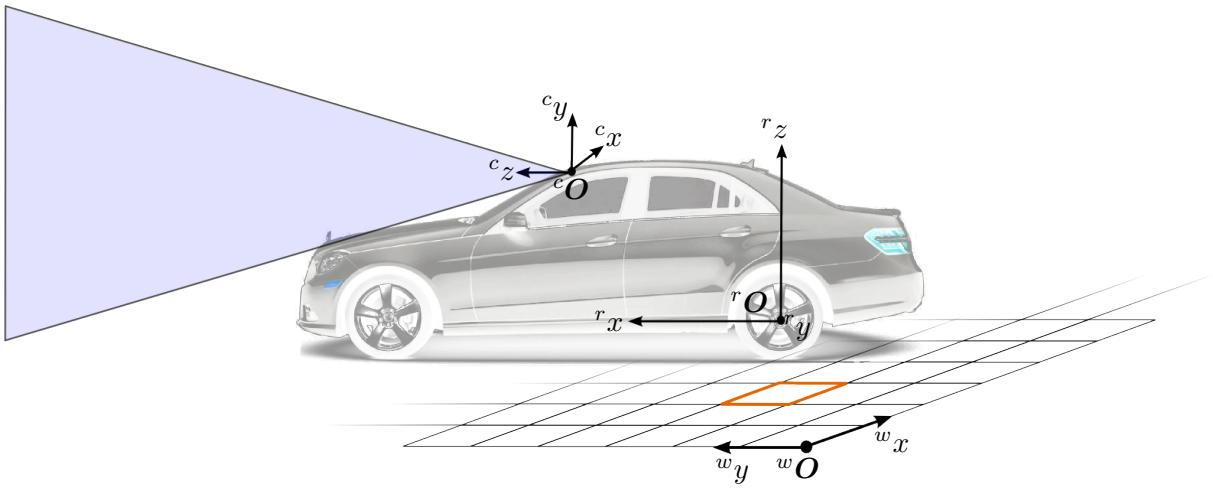


Figure 3.4: Overview of the used coordinate systems. S_c is the 3D camera coordinate system, S_r is the coordinate system of the test vehicle, and S_w is the 2D coordinate system of the global grid map.

time steps $t-1$ and t. Since a 2D mapping approach is developed we also define the motion of the vehicle in 2D which is defined by the forward velocity v along the ${}^{r}x$-axis and the yaw rate $\dot{\varphi}$ around the ${}^{r}z$-axis measured at time step $t-1$. The discrete time interval Δt depends on the rate of the IMU. This relative 2D odometry information is represented by the control vector $\boldsymbol{u} = [v, \dot{\varphi}, \Delta t]^{\mathsf{T}}$. As already mentioned in Sec. 2.6, we concatenate all odometry information up to time $t-1$ in $\mathcal{U}_{0:t-1} = \{\boldsymbol{u}_0, ..., \boldsymbol{u}_{t-1}\}$. The relative homogeneous motion matrix ${}^{r}\Delta\mathsf{M}_{t-1}^{t}(\boldsymbol{u}_{t-1})$, in reference to the vehicle frame S_r, is defined by:

$$ {}^{r}\Delta\mathsf{M}_{t-1}^{t}(\boldsymbol{u}_{t-1}) = {}^{r}\begin{bmatrix} \Delta\boldsymbol{R}(\dot{\varphi}_{t-1}, \Delta t) & \Delta\boldsymbol{T}(v_{t-1}, \dot{\varphi}_{t-1}, \Delta t) \\ \mathbf{0}_3^T & 1 \end{bmatrix}_{t-1}^{t} . \tag{3.1} $$

The relative 2D rotation matrix $\Delta\boldsymbol{R}(\dot{\varphi}_{t-1}, \Delta t)$ and the 2D translation vector $\Delta\boldsymbol{T}(v_{t-1}, \dot{\varphi}_{t-1}, \Delta t)$ are estimated using a desired vehicle motion model. Here, we apply a motion model with constant velocity and yaw rate which is described e.g. in [Barth, 2010] and [Badino et al., 2013] in detail. The multiplication of all relative motion matrices of (3.1) up to time step $t-1$ results in the absolute motion matrix ${}^{r}\mathsf{M}_t({}^{r}\boldsymbol{x}_t)$:

$$ {}^{r}\mathsf{M}_t({}^{r}\boldsymbol{x}_t) = {}^{r}\Delta\mathsf{M}_0^1(\boldsymbol{u}_0)\ {}^{r}\Delta\mathsf{M}_1^2(\boldsymbol{u}_1)\ ...\ {}^{r}\Delta\mathsf{M}_{t-1}^{t}(\boldsymbol{u}_{t-1}) . \tag{3.2} $$

Here, the vector ${}^{r}\boldsymbol{x}_t$ includes the absolute pose parameters with ${}^{r}\boldsymbol{x}_t = [{}^{r}x_t, {}^{r}y_t, {}^{r}\varphi_t]^{\mathsf{T}}$. Thus, the homogeneous matrix ${}^{r}\mathsf{M}_t({}^{r}\boldsymbol{x}_t)$ is defined by the absolute 2D rotation $\boldsymbol{R}({}^{r}\varphi_t)$ and the translation vector $[{}^{r}x_t, {}^{r}y_t]^{\mathsf{T}}$:

$$ {}^{r}\mathsf{M}_t({}^{r}\boldsymbol{x}_t) = {}^{r}\begin{bmatrix} \boldsymbol{R}({}^{r}\varphi_t) & [{}^{r}x_t, {}^{r}y_t]^{\mathsf{T}} \\ \mathbf{0}_3^T & 1 \end{bmatrix}_{t} . \tag{3.3} $$

Under the assumption that we know the transformation between the map coordinate system S_w and the vehicle coordinate S_r at $t = 0$, defined by ${}^{w}_{r}\mathsf{M}({}^{w}\boldsymbol{x}_0)$, the motion of the vehicle at t with reference to S_w is estimated by:

$$ {}^{w}\mathsf{M}_t({}^{w}\boldsymbol{x}_t) \;=\; {}^{w}_{r}\mathsf{M}({}^{w}\boldsymbol{x}_0)\; {}^{r}\mathsf{M}_t({}^{r}\boldsymbol{x}_t)\;. \tag{3.4} $$

All pose vectors are concatenated to ${}^{w}\mathcal{X}_{0:t} = \{{}^{w}\boldsymbol{x}_0, ..., {}^{w}\,\boldsymbol{x}_t\}$. We neglect the index w for convenience only, and write $\mathcal{X}_{0:t} = \{\boldsymbol{x}_0, ..., \boldsymbol{x}_t\}$ which was already introduced in Sec. 2.6.

3.2.3 An efficient Scene Representation as Input Data

This section defines the input data for the novel mapping approach based on stereo vision. In contrast to general mapping techniques where raw sensor readings are used, e.g. raw LIDAR point clouds, we post-process the original sensor data first. The work flow of these steps are shown in Fig. 3.5.

From Disparities to Stixels. Based on the rectified image sequences of the stereo camera rig, dense disparity images and its confidence maps are estimated via the SGM algorithm at each acquisition time step. Details of these algorithms are described in Sec. 2.2.2. We use the Stixel World (see Sec. 2.2.3) which represents the relevant information of the current scene in terms of free space and obstacles with only a few hundred Stixels only. To achieve a dynamic scene representation, the Stixels are tracked over consecutive time steps with the help of Kalman filters. This allows us to segment the dynamic Stixel World into moving and stationary obstacles by means of a multi-class traffic scene segmentation which is based on graph cuts. Details are further described in Sec. 2.2.3.

The Input Data. We use the static Stixel sets $\mathcal{S}^{\text{lab}} = \{\mathcal{S}^{\text{lab}}_u\}$ with $\mathcal{S}^{\text{lab}}_u = \{\boldsymbol{s}^{\text{lab}}_{un}\}$ as our input data. The Stixel sets are collected over time into the observation set $\mathcal{S}^{\text{lab}}_{0:t} = \{\mathcal{S}^{\text{lab}}_0, ..., \mathcal{S}^{\text{lab}}_t\}$. A detailed overview of the definition and notation of the Stixel sets is described in Sec. 2.2.3. The label "lab" tells us, that we use the segmented Stixel World. To make the following equations and derivation steps more readable, we neglect the label type of the Stixel sets and define $\mathcal{S}_{0:t} = \mathcal{S}^{\text{lab}}_{0:t}$.

The main advantage of the Stixel segmentation is that we neglect dynamic obstacles during the mapping step which improves the quality of our generated maps. Otherwise, we would need to detect dynamic objects in the grid maps which is, in general, more difficult. Furthermore, the Stixels gives us direct information about free space and obstacles which help during the definition of the observation model later on. Following the definitions in Sec. 2.2.3 a single Stixel is defined by

$$ \boldsymbol{s}_{un} = [u, v^{\text{bt}}_{un}, v^{\text{tp}}_{un}, w, d_{un}, \tilde{\sigma}^2_{d_{un}}, c_{un}, p^{\text{out}}_{un}, {}^{c}\,\dot{z}_{un}, {}^{c}\,\dot{x}_{un}, l_{un}]. \tag{3.5} $$

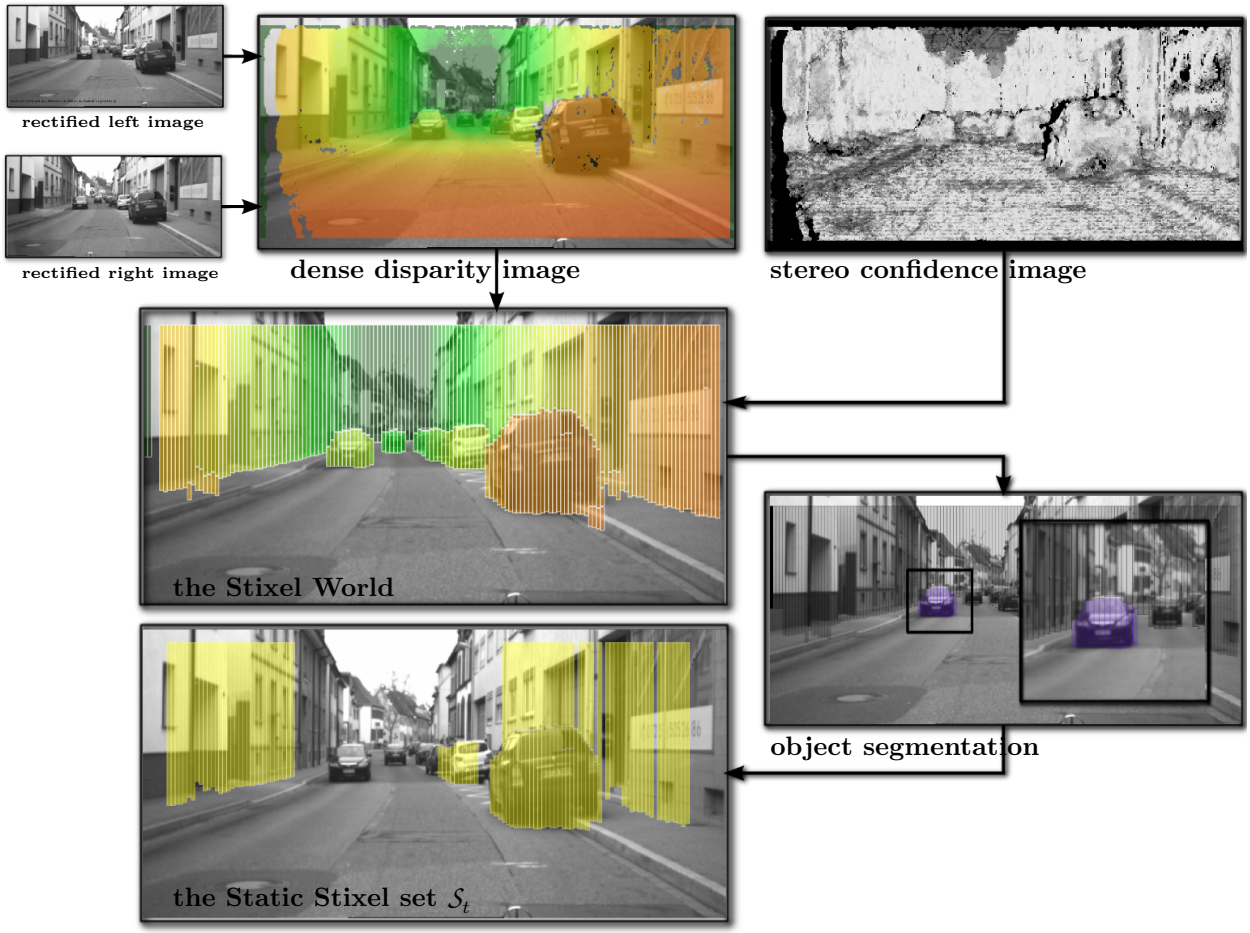


Figure 3.5: Data preprocessing steps. Based on rectified image sequences, dense disparity images are estimated via SGM. Red color stands for near by and green for far away objects. The *Stixel World* is generated using the disparity images and stereo confidence values. The color encoding is the same as for the SGM result. To distinguish between static and dynamic objects, the Stixels are segmented via Graph Cuts; the oncoming car is classified and shown in purple. All other Stixels are labeled as background. By using Stixels only with the class types c_{object} and l_{static} we only consider static environment information for the novel mapping approach. The Stixel set $\mathcal{S}$ is represented by yellow marked Stixels and represents the input data. Because of the uncertain behavior of stereo vision we only consider Stixels up to a distance of 40 m.

For the meaning of the parameters of a single Stixel see Tab. 3.1. We only consider Stixels up to a distance of 40 m since the precision of stereo decreases quadratically with the measured distance which is discussed in Sec. 2.2.2.3. Based on this fact we only use the first two Stixel segments ($n = 1, 2$) per image column u. The following key points summarize the main benefits (+) and downsides (−) of this representation and, therefore, complete this section.

- The Stixel World classifies the disparity information into free space and obstacles (+) .
- The segmentation of the Stixel World allows us to neglect dynamic obstacles during the mapping process (+) .
- The enormous data reduction (compared to the raw disparity images) allows an efficient use in further real-time capable processing steps (+) .
- During the Stixel generation the assumption is made that all obstacles in the environment are only represented by vertically structured rectangles (−).
- The abstraction of the environment into rectangles increases discretization errors (−).

Table 3.1: The parameters of a single Stixel element $\boldsymbol{s}_{un}$. Each Stixel includes information about its geometry (image position, disparity value, width and velocity), its uncertainty (disparity uncertainty and outlier probability), and its label (free space vs. obstacles, moving vs. static obstacles).

Stixel element	Description	
u	column position	
v_{un}^{bt}	row position (bottom)	
v_{un}^{tp}	row position (top)	
w	width	**geometry**
d_{un}	disparity value	
${}^{c}\dot{z}_{un}$	longitudinal velocity	
${}^{c}\dot{x}_{un}$	lateral velocity	
$\tilde{\sigma}_{d_{un}}^2$	emp. standard deviation of the disparity	**uncertainty**
p_{un}^{out}	outliers probability	
c_{un}	class definition, with $c \in \{\texttt{free space}, \texttt{obstacle}\}$	**labels**
l_{un}	dyn. class definition, with $l \in \{\texttt{movingObj}, \texttt{staticObj}\}$	

3.3 The Optimization Formulation

In this section we formulate the previously stated objective (see Sec. 3.1) as an optimization problem in a probabilistic fashion. The goal is to estimate the global posterior distribution of a 2D grid map $\mathcal{M}_t$ given the input data $\mathcal{S}_{0:t}$ and the pose information $\mathcal{X}_{0:t}$ up to time step t. The Map $\mathcal{M}_t$ is represented as a regular, 2D occupancy grid map with its characteristics stated in Sec. 2.5. The individual grid cells m_i with $i \in I$ are binary variables which reflect the occupancy probability. Because of their generic structure, these maps are suitable for on-line applications, e.g. localization or path planning approaches (see Sec. 1.2.4 and Sec. 1.2.5). The representation is also independent of the used sensor type which makes the following optimization formulation generic with regard to the input data. In a probabilistic way, we want to estimate the posterior of the map $\mathcal{M}_t$

$$p(\mathcal{M}_t|\mathcal{S}_{0:t}, \mathcal{X}_{0:t}) \ .$$

In our new concept we postulate that neighboring grid cells are *dependent* which is the main contribution of this thesis. Therefore, we do not easily factorize over the single grid cells as stated in Sec. 2.5, (2.40).

In contrast to most common mapping techniques, the modeling of dependent grid cells has received little attention in literature as already observed in Sec. 1.2.5. This motivates us to formulate the posterior of the map as a MRF which considers dependencies between neighboring grid cells. The map is interpreted as a probabilistic undirected graph (see Sec. 2.3) where the nodes represent the single grid cells and the dependencies between them are modeled as undirected edges. Similar to Sec. 2.3.1, (2.19), the global joint distribution $p(\mathcal{M}_t|\mathcal{S}_{0:t}, \mathcal{X}_{0:t})$ is factorized into potential functions which model the dependencies between the Map $\mathcal{M}_t$ and the observations $\mathcal{S}_{0:t}$ and $\mathcal{X}_{0:t}$. The overall posterior $p(\mathcal{M}_t|\mathcal{S}_{0:t}, \mathcal{X}_{0:t})$ is decomposed into the product of unary and binary terms with a maximum clique size of two:

$$p(\mathcal{M}_t|\mathcal{S}_{0:t}, \mathcal{X}_{0:t}) \ = \ \frac{1}{\zeta} \prod_{C \in \mathcal{I}} \underbrace{\Psi(m_{i,t}|\mathcal{S}_{0:t}, \mathcal{X}_{0:t})}_{\text{unary terms}} \prod_{(i,j) \in N_4} \underbrace{\Phi(m_{i,t}, m_{j,t}|\mathcal{S}_{0:t}, \mathcal{X}_{0:t})}_{\text{binary terms}} \ . \tag{3.6}$$

The unary terms $\Psi(m_{i,t}|\mathcal{S}_{0:t}, \mathcal{X}_{0:t})$ specify how each individual grid cell is influenced by the observations and the pose, whereas the binary terms $\Phi(m_{i,t}, m_{j,t}|\mathcal{S}_{0:t}, \mathcal{X}_{0:t})$ model the dependencies between neighboring grid cells. We assume a 4-neighborhood rule which is stated by the symbol N_4 with the index tuples (i, j). For the normalization we have to consider the partition function ζ which was also introduced in Sec. 2.3.1, (2.17). In the next sections we define the unary and binary terms in detail.

3.4 Definition of the Unary Terms

We describe the definition of the unary terms which includes the derivation to a time recursive structure (Sec. 3.4.1), the definition of our measurement model (Sec. 3.4.2) and how to formulate the prediction terms (Sec. 3.4.3) in the following. This section also include parts which were already published in [Muffert et al., 2014].

3.4.1 Derivation of a Time Recursive Structure

To achieve an on-line capable mapping algorithm we derive a recursive estimator using the approach equal to the one described in Sec. 2.4 and in Muffert et al. [2014]. The potentials of the unary terms $\Psi(m_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t})$ represent the influence of the Stixel set $\mathcal{S}_{0:t}$ and the pose set $\mathcal{X}_{0:t}$ on a single grid cell. The unary potential terms are defined as the conditional distribution

$$p(m_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t}) = \Psi(m_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t}) \ . \tag{3.7}$$

To obtain a recursive structure, the first assumption is that all observed Stixel sets between consecutive time steps $\mathcal{S}_t$ and $\mathcal{S}_{t-1}$ are completely independent. In general, this statement is true since Stixels are generated at each time step individually from disparity images. However, Stixel tracking and Stixel segmentation (see Sec. 2.2.3) lead to a time dependency between the Stixel sets $\mathcal{S}_t$ and $\mathcal{S}_{t-1}$. Therefore, consecutive Stixel sets are highly correlated. This correlation is not taken into account in this thesis to allow a time recursive structure.

By applying Bayes rule, we observe:

$$p(m_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t}) \overset{\text{Bayes rule}}{=} \frac{p(\mathcal{S}_{0:t} \mid m_{i,t}, \mathcal{X}_{0:t}) \ p(m_{i,t}|\mathcal{X}_{0:t})}{p(\mathcal{S}_{0:t}|\mathcal{X}_{0:t})} \ . \tag{3.8}$$

Separating the Stixel set $\mathcal{S}_t$ at time t from the whole history $\mathcal{S}_{0:t-1}$ and take the independence of consecutive Stixel sets into account, we obtain

$$= \frac{p(\mathcal{S}_t, \mathcal{S}_{0:t-1} \mid m_{i,t}, \mathcal{X}_{0:t}) \ p(m_{i,t}|\mathcal{X}_{0:t})}{p(\mathcal{S}_{0:t}|\mathcal{X}_{0:t})} \tag{3.9}$$

$$= \frac{p(\mathcal{S}_t \mid m_{i,t}, \mathcal{X}_{0:t}) \ p(\mathcal{S}_{0:t-1} \mid m_{i,t}, \mathcal{X}_{0:t}) \ p(m_{i,t}|\mathcal{X}_{0:t})}{p(\mathcal{S}_{0:t}|\mathcal{X}_{0:t})} \ . \tag{3.10}$$

By using the Bayes rule for the term $p(\mathcal{S}_{0:t-1} \mid m_{i,t}, \mathcal{X}_{0:t})$, we observe:

$$= \frac{p(\mathcal{S}_t \mid m_{i,t}, \mathcal{X}_{0:t}) \ p(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t}) \ p(\mathcal{S}_{0:t-1} \mid \mathcal{X}_{0:t}) \ \cancel{p(m_{i,t}|\mathcal{X}_{0:t})}}{\cancel{p(m_{i,t} \mid \mathcal{X}_{0:t})} \ p(\mathcal{S}_{0:t}|\mathcal{X}_{0:t})}$$

$$= \frac{p(\mathcal{S}_t \mid m_{i,t}, \mathcal{X}_{0:t}) \ p(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t}) \ p(\mathcal{S}_{0:t-1} \mid \mathcal{X}_{0:t})}{p(\mathcal{S}_{0:t}|\mathcal{X}_{0:t})} \ . \tag{3.11}$$

With the definition of conditional probability the denominator $p(\mathcal{S}_{0:t}|\mathcal{X}_{0:t})$ is expressed as:

$$p(\mathcal{S}_{0:t}|\mathcal{X}_{0:t}) = p(\mathcal{S}_t, \mathcal{S}_{0:t-1} \mid \mathcal{X}_{0:t}) = \frac{p(\mathcal{S}_t, \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})}{p(\mathcal{X}_{0:t})} \tag{3.12}$$

$$= \frac{p(\mathcal{S}_t, \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})}{p(\mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})} \ \frac{p(\mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})}{p(\mathcal{X}_{0:t})} \tag{3.13}$$

$$= p(\mathcal{S}_t \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t}) \ p(\mathcal{S}_{0:t-1} \mid \mathcal{X}_{0:t}) \ . \tag{3.14}$$

Next, we substitute (3.14) in (3.11) and obtain:

$$p(m_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t}) = \frac{p(\mathcal{S}_t \mid m_{i,t}, \mathcal{X}_{0:t})\; p(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})\; \cancel{p(\mathcal{S}_{0:t-1} \mid \mathcal{X}_{0:t})}}{p(\mathcal{S}_t|\mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})\cancel{p(\mathcal{S}_{0:t-1}|\mathcal{X}_{0:t})}} \tag{3.15}$$

$$= \frac{p(\mathcal{S}_t \mid m_{i,t}, \mathcal{X}_{0:t})\; p(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})}{p(\mathcal{S}_t|\mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})} \tag{3.16}$$

$$\propto \underbrace{p(\mathcal{S}_t \mid m_{i,t}, \mathcal{X}_{0:t})}_{\text{measurement model}}\; \underbrace{p(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})}_{\text{prediction term}}. \tag{3.17}$$

The term $p(\mathcal{S}_t \mid m_{i,t}, \mathcal{X}_{0:t})$ represents our measurement model which describes the process of modeling the environment based on uncertain sensor readings at time step t.. The second term $p(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})$ is called the prediction term since the state of a grid cell is estimated without the Stixel set $\mathcal{S}_t$. Equations (3.16) and (3.17), respectively, have the same form as the general recursive filter estimator in Sec. 2.4.1, (2.33). The measurement model and the prediction term are defined in the following sections.

3.4.2 The Measurement Model

The measurement model $p(\mathcal{S}_t \mid m_{i,t}, \mathcal{X}_{0:t})$ describes the conditional probability of the complete Stixel set at time step t given a specific grid cell and the trajectory of the ego vehicle. To achieve real-time capability and define the measurement model in an efficient way, we assume that all single Stixels $\boldsymbol{s}_{un,t}$ in the current Stixel $\mathcal{S}_t$ set are conditionally independent. Independence between observations is a common assumption in robotics which is also state in Thrun et al. [2005, p. 152]. Therefore, we factorize over the image columns u first:

$$p(\mathcal{S}_t \mid m_{i,t}, \mathcal{X}_{0:t}) = \prod_u p(\mathcal{S}_{u,t} \mid m_{i,t}, \mathcal{X}_{0:t})\ . \tag{3.18}$$

Since multiple Stixels per column u exist and we also assume that these Stixels are independent, we factorize over the number of segments per coloum n:

$$= \prod_u \prod_n p(\boldsymbol{s}_{un,t} \mid m_{i,t}, \mathcal{X}_{0:t}) \tag{3.19}$$

Because of the fact that the complete trajectory $\mathcal{X}_{0:t}$ is given w.r.t. the global map system S_w, we consider only the current global pose information ${}^w\boldsymbol{x}_t$ in the measurement model:

$$= \prod_u \prod_n p(\boldsymbol{s}_{un,t} \mid m_{i,t}, {}^w\boldsymbol{x}_t). \tag{3.20}$$

Observe, that we add the index w again to explicitly state that the current pose is w.r.t. the map system S_w.

3.4.2.1 Definition of the Measurement Model in the Column-Disparity Space

In this subsection we define the measurement model in the column-disparity space. By exploring the term $p(\boldsymbol{s}_{un,t} \mid m_{i,t}, {}^{w}\boldsymbol{x}_t)$ in detail, one challenge arrives which was also stated in Muffert et al. [2014]. Each single Stixel is originally computed in the local two dimensional column(u)-disparity(d) space of the camera system whereas the grid cells and the pose information are given w.r.t. the global Cartesian system S_w. Therefore, it is necessary to transform either the Stixel information into the global Cartesian grid map space or to project the cell position into the local column-disparity space. Due to the fact that the measurement model should describe the nature of sensor behavior in a very precise way, and the fact that the Stixels are estimated from disparity images, we define the measurement model $p(\boldsymbol{s}_{un,t} \mid m_{i,t}, {}^{w}\boldsymbol{x}_t)$ in the column-disparity space. This requires the transformation of $m_{i,t}$ into the column-disparity space.

Transformation in the Column-Disparity Space. Using the index function f_i, the coordinate components of each cell are given by $f_i : m_{i,t} \mapsto [{}^{w}x_{m_{i,t}}, {}^{w}y_{m_{i,t}}]^{\mathsf{T}}$. Using the inverse motion matrix ${}^{w}\mathsf{M}_t^{-1}({}^{w}\boldsymbol{x}_t)$, the grid cell coordinates ${}^{r}x_{m_{i,t}}$ and ${}^{r}y_{m_{i,t}}$ in the vehicle coordinate system S_r are obtained by:

$$\begin{bmatrix} {}^{r}x_{m_{i,t}} \\ {}^{r}y_{m_{i,t}} \\ 1 \end{bmatrix} = {}^{w}\Delta\mathsf{M}_t^{-1}({}^{w}\boldsymbol{x}_t) \begin{bmatrix} {}^{w}x_{m_{i,t}} \\ {}^{w}y_{m_{i,t}} \\ 1 \end{bmatrix} . \tag{3.21}$$

Taking the focal length f, the basis b, and the column component of the principal point u_0 of the current stereo system into account (see also Sec. 2.2.2.1), the estimated disparity $d_{m_{i,t}}$ and the image column $u_{m_{i,t}}$ of the corresponding grid cell $m_{i,t}$ are defined by:

$$d_{m_{i,t}} = \frac{f\ b}{{}^{r}x_{m_{i,t}}} \tag{3.22}$$

and

$$u_{m_{i,t}} = u_0 - \frac{f\ {}^{r}y_{m_{i,t}}}{{}^{r}x_{m_{i,t}}} . \tag{3.23}$$

The Density Functions. Similar to the range finder model proposed in [Thrun et al., 2005, Chapter 6.3], our measurement model is defined by a mixture distribution which handles noisy depth measurements and outliers. We make a reasonable assumption that the Stixels are only uncertain in the disparity $d_{un,t}$. This allows us to model the measurement noise of the Stixel with a 1D Gaussian distribution $\mathcal{N}\big(d_{m_{i,t}}, \sigma^2_{d_{m_{i,t}}}\big)$ with the mean $d_{m_{i,t}}$ and variance $\sigma^2_{d_{m_{i,t}}}$. The measurement probability is given by

$$p_{\text{meas}}\big(d_{un,t} \mid d_{m_{i,t}}, \sigma^2_{d_{m_{i,t}}}\big) =$$

$$\begin{cases} \eta_{\text{meas}}\ \mathcal{N}\big(d_{un,t}; d_{m_{i,t}}, \sigma^2_{d_{m_{i,t}}}\big), & \text{if: } d_{\max} > d_{un,t} > d_{\min} \text{ and } u - \frac{1}{2}w < u_{m_{i,t}} < u + \frac{1}{2}w \\ 0 \qquad\qquad\qquad\qquad\quad , & \text{else .} \end{cases} \tag{3.24}$$

The expression $\mathcal{N}\big(d_{un,t}; d_{m_{i,t}}, \sigma^2_{d_{m_{i,t}}}\big)$ is the normal distribution which is defined by the mean $d_{m_{i,t}}$ and standard deviation $\sigma_{d_{m_{i,t}}}$, and evaluated at the Stixel's disparity value $d_{un,t}$. For practical reasons, the function is only valid if $d_{un,t}$ is in the disparity interval $[d_{\max}; d_{\min}]$ and the estimated column index $u_{m_{i,t}}$ is equal to the Stixel's column index u. The normalizer η_{meas} is defined by the integral over the disparity interval

$$\eta_{\text{meas}} = \Big(\int_{d_{\min}}^{d_{\max}} \mathcal{N}\big(d_{un,t}; d_{m_{i,t}}, \sigma^2_{d_{m_{i,t}}}\big)\ dd_{un,t} \Big)^{-1}. \tag{3.25}$$

To handle possible Stixel outliers, we model a uniform distribution over the valid disparity space:

$$p_{\text{out}}\big(d_{un,t} \mid d_{m_{i,t}}\big) = \begin{cases} \frac{1}{(d_{\max} - d_{\min})}, & \text{if: } d_{\max} > d_{un,t} > d_{\min} \text{ and } u - \frac{1}{2}w < u_{m_{i,t}} < u + \frac{1}{2}w \\ 0 \qquad\qquad , & \text{else .} \end{cases} \tag{3.26}$$

We formulate the overall density function for occupied grid cells by combining both distributions in (3.24) and (3.26):

$$p(\boldsymbol{s}_{un,t} \mid m_{i,t}, {}^{w}\boldsymbol{x}_t) = \alpha\ p_{\text{meas}}\big(d_{un,t} \mid d_{m_{i,t}}, \sigma^2_{d_{m_{i,t}}}\big)\ +\ \big(1-\alpha\big)\ p_{\text{out}}(d_{un,t} \mid d_{m_{i,t}})\ . \tag{3.27}$$

For the definition of the unary potential functions in the MRF (see Sec. 3.6.1), we also have to define the density functions $p(\boldsymbol{s}_{un,t} \mid \neg m_{i,t}, {}^{w}\boldsymbol{x}_t)$, since

$$p(\boldsymbol{s}_{un,t} \mid \neg m_{i,t}, {}^{w}\boldsymbol{x}_t) \neq 1 - p(\boldsymbol{s}_{un,t} \mid m_{i,t}, {}^{w}\boldsymbol{x}_t)\ . \tag{3.28}$$

Here, $\neg m_{i,t}$ represents a free grid cell. The density functions of free space are also realized by a mixture distribution:

$$p(\boldsymbol{s}_{un,t} \mid \neg m_{i,t}, {}^{w}\boldsymbol{x}_t) = \alpha\ p_{\text{free}}\big(d_{un,t} \mid d_{m_{i,t}}, \sigma^2_{d_{m_{i,t}}}\big)\ +\ \big(1-\alpha\big)\ p_{\text{out}}(d_{un,t} \mid d_{m_{i,t}})\ . \tag{3.29}$$

The probability $p_{\text{free}}\big(d_{un,t} \mid d_{m_{i,t}}, \sigma^2_{d_{m_{i,t}}}\big)$ represents valid free space and is given by:

$$p_{\text{free}}\big(d_{un,t} \mid d_{m_{i,t}}, \sigma^2_{d_{m_{i,t}}}\big) =$$

$$\begin{cases} \eta_{\text{free}}\ \Big(1 - \mathcal{N}\big(d_{un,t}; d_{m_{i,t}}, \sigma^2_{d_{m_{i,t}}}\big)\Big), & \text{if: } d_{\max} > d_{un,t} > d_{\min} \text{ and } u = u_{m_{i,t}} \\ 0 \qquad\qquad\qquad\qquad\qquad\qquad , & \text{else .} \end{cases} \tag{3.30}$$

The normalizer η_{free} for free space is estimated similar to the normalizer in (3.25). Equations (3.29) and (3.20) allow us to define the overall posterior

$$p(\mathcal{S}_t \mid \neg m_{i,t}, \mathcal{X}_{0:t}) = \prod_u \prod_n p(\boldsymbol{s}_{un,t} \mid \neg m_{i,t}, {}^{w}\boldsymbol{x}_t). \tag{3.31}$$

Two examples of the density functions are shown in Fig. 3.6. The variance $\sigma^2_{d_{m_{i,t}}}$ and the weighting factor α are the model parameters $\Theta = [\sigma^2_{d_{m_{i,t}}}, \alpha]$ and have to be learned in general. This topic is discussed in the following paragraph.

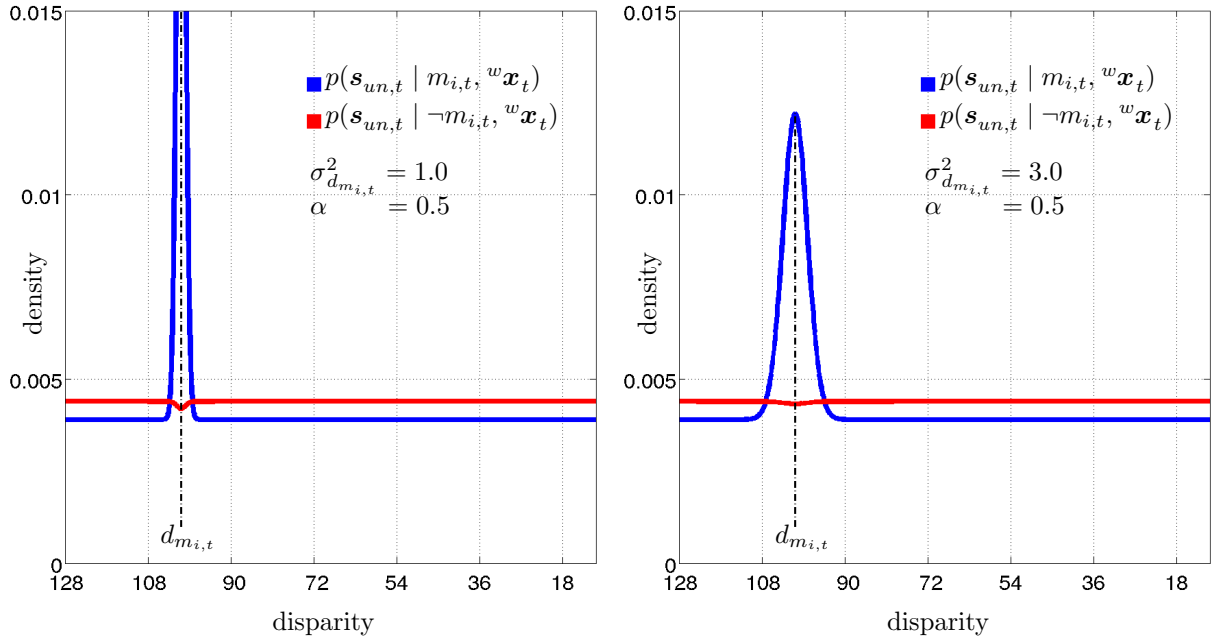


Figure 3.6: Examples of density functions which are used for defining the measurement model. In the left example the model parameters are set to $\sigma^2_{d_{m_{i,t}}} = 1.0$ and $\alpha = 0.5$, in the example on the right the parameters are $\sigma^2_{d_{m_{i,t}}} = 3.0$ and $\alpha = 0.5$. In both cases, the disparity space is limited by the continuous interval $[128, 0]$. Therefore, $d_{\max}$ is 128 and $d_{\min}$ is 0.

Estimation of the Model Parameters Θ. The determination of the model parameters Θ has an important influence on the defined measurement model. Different possibilities exist to estimate these parameters. As mentioned in [Thrun et al., 2005, Chapter 6.3.2] the model parameters can be determined by using measurement data in combination with ground truth information. Based on this idea, Pfeiffer et al. [2010] developed an approach to analyze the precision of the Stixels using a high performance laser scanner as a reference sensor. This allows the use of statistical analysis to estimate the variance $\sigma^2_{d_{m_{i,t}}}$ and make assumptions about the outlier probability which control the weighting factor α. Another possibility is to learn the complete parameter set in the mapping process with actual measurement data. Here, we have to consider the model parameters Θ in the measurement model and adjust the likelihood in (3.20) to $p(\mathcal{S}_t \mid m_{i,t}, \mathcal{X}_{0:t}, \Theta)$. This leads to a maximum likelihood estimator which is used to iteratively estimate the parameters. Details of this estimator are also discussed in [Thrun et al., 2005, Chapter 6.3.2].

In this thesis, the model parameters are not learned in the mapping approach. We instead rely on Stixel information which also includes the precision information $\tilde{\sigma}^2_{d_{un}}$ and outlier probabilities p^{out}_{un}. Therefore, the precision of the Gaussian is set by the empirical standard deviation of each Stixel $\sigma^2_{d_{m_{i,t}}} = \tilde{\sigma}^2_d$ and the weighting factor is defined by the outlier probability $\alpha = p_{\text{out}}$.

3.4.2.2 Realization and Practical Considerations

In this section we discuss the realization and practical consideration in the implementation of the measurement model. This includes the definition of the disparity intervals for different Stixel types, the use of the discrete disparity space, and the inverse mapping into the Cartesian space.

The Disparity Intervals. As shown in Fig. 3.7, we define three major different Stixel scenarios. The figure shows our world assumptions for (1) static Stixels in the first row ($n = 1$), (2) static Stixels in the second row ($n = 2$), and, (3) Stixels which are labeled as dynamic obstacles ($l_{un} =$ `movingObj`). As seen in (3.24) - (3.30), the density functions are only valid in defined disparity ranges. Without a limitation of these disparity ranges for the three different scenarios we are not able to fulfill the named world assumptions. Therefore, we do the following definitions by limiting the disparity ranges for $d_{\min}$ and $d_{\max}$. Please consider that the disparity values decrease by increasing the distance.

- For the first, static Stixel in a column with $n = 1$, $c_{un} =$ `obstacle`, and $l_{un} =$ `staticObj`, we assume that the space up to the foot point of the Stixel is free space. Around the foot point of the Stixel we assume a static obstacle which occupies the surrounding grid cells. The space behind the closest Stixel is, in general, unknown area. For this situation, the disparity range is defined by $d_{\max} = 128$ and $d_{\min} = d_{un,t} - 2\sigma^2_{d_{m_{i,t}}}$.

- Stixel obstacles with $n = 2$, $c_{un} =$ `obstacle`, and $l_{un} =$ `staticObj` give us information only about the existence of that obstacle. Since we are working in a 2D grid, we project these 2 nd row Stixels into the plane of the grid map. The reasonable assumption is made that no information about free space in front of and behind these Stixels is available. The maximum value is changed and is defined by $d_{\max} = d_{un,t} + 2\sigma^2_{d_{m_{i,t}}}$. The minimum value is the same than in the situation where $n = 1$.

- For Stixels labeled as dynamic obstacles with $n = 1$, $c_{un} =$ `obstacle`, and $l_{un} =$ `movingObj`, we assume that the space between the vehicle and the other moving Stixel obstacle is modeled as free space. Therefore, we set the disparity interval limits for these kind of Stixels to $d_{\max} = 128$ and $d_{\min} = d_{un,t} + 2\sigma^2_{d_{m_{i,t}}}$.

Using the Discrete Disparity Grid Space. Referring to (3.20), the major challenge is that we have to iterate over all individual disparity values $d_{m_{i,t}}$ for the grid cells of the complete map which is not very efficient for on-line applications. To overcome this burden, we directly define a local grid map in the column (u^*)-disparity(d^*)space. Since the stereo camera geometry is assumed to be stable and the disparity space is limited in the range of $[128, 0]$, the column-disparity space is constant at any time step t. The column space is in the range of $u^* \in [0, W]$ where W is the image width. The disparity space has to be quantized in the range of $d^* \in [128, 0]$ with a defined sampling interval $f_s = \frac{1}{d_s}$ where d_s is the disparity sampling rate. The bigger the disparity sampling rate, the higher the resolution of the local disparity space. The parameter d_s has to be defined and is an important factor in our evaluation.

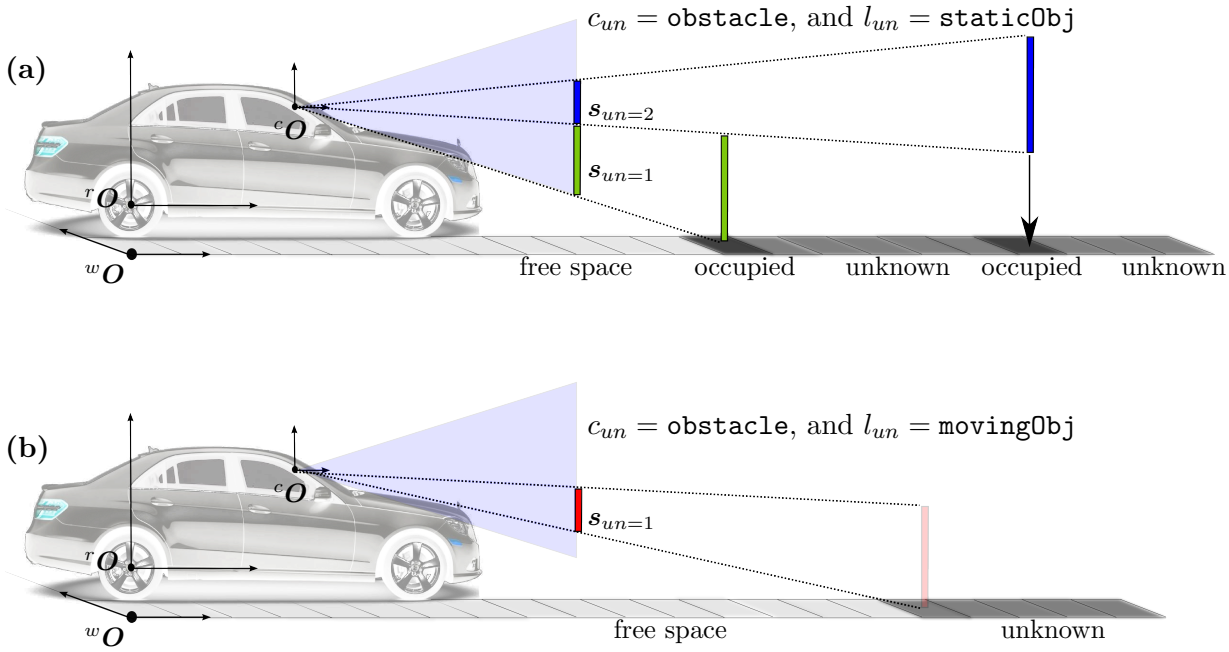
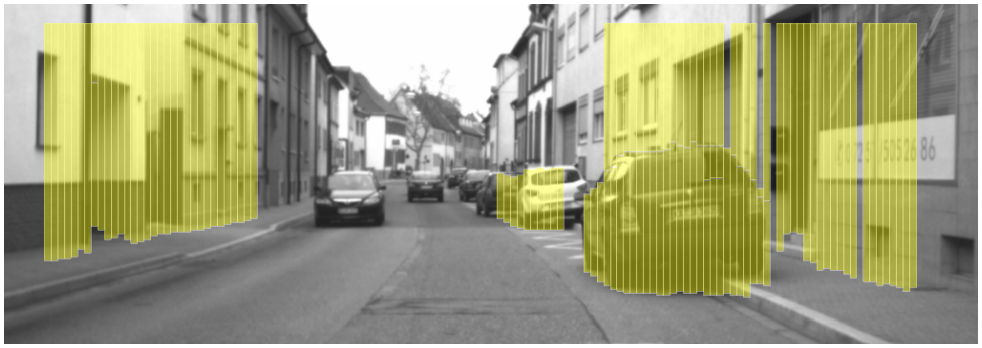


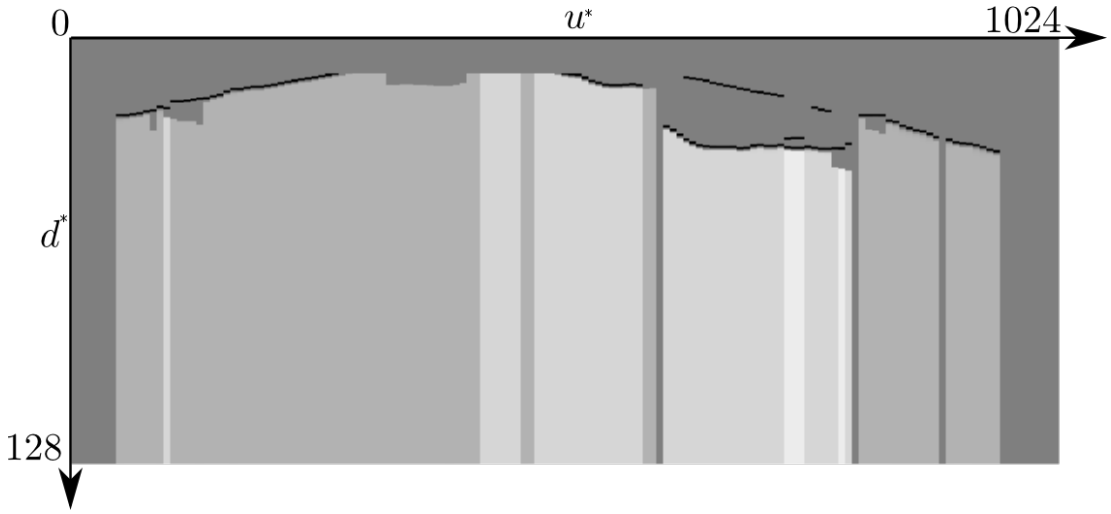
Figure 3.7: Model assumptions with regard to the used Stixel types. In (a) the model assumptions for static Stixels with c_{un} = `obstacle`, and l_{un} = `staticObj` are represented. Stixels with $n = 1$ include information about free space and occupied areas. Stixels with $n > 1$ give us information only about occupied areas since we project each Stixel into the 2D space of the grid map. Thus, we define that no obstacle or free space information between the two Stixel types is available. In (b) we present the assumption for Stixels labeled as dynamic obstacles with c_{un} = `obstacle`, and l_{un} = `movingObj`. We only model the free space area up to the foot point of these Stixel types.

Since a discrete disparity space is chosen, the continuous density functions in (3.27) and (3.29) have to be replaced by the discrete functions $P(\boldsymbol{s}_{un,t} \mid m_{i,t}, {}^{w}\boldsymbol{x}_t)$ and $P(\boldsymbol{s}_{un,t} \mid \neg m_{i,t}, {}^{w}\boldsymbol{x}_t)$. Furthermore, the normalizers η_{meas} and η_{free} are defined by the sum over the corresponding discrete interval. The defined column-disparity (u^*-d^*) space is in the range of the complete Stixel set $\mathcal{S}_{u,t}$ which allows us to generate a dense column-disparity occupancy grid map. An example is shown in Fig. 3.8 which represents our desired measurement model in the local column-disparity space at time step t.

Inverse Mapping into the Cartesian Space. In this paragraph the transformation of the dense disparity occupancy grid in the Cartesian space is explained. Equations (3.22) and (3.23) are inverted to calculate the Cartesian grid cells which are influenced from the column (u^*)-disparity(d^*) occupancy grid. Because of the characteristic of the disparity space the warping into the Cartesian space leads to ambiguities which is also mentioned in [Badino et al., 2007; Perrollaz et al., 2010] and shown in Fig. 3.9. We observe, that Cartesian grid cells near the origin are influenced by several column-disparity cells and, vice versa, several Cartesian grid cells far away from the origin



(a) The static Stixel set $\mathcal{S}_t$ limited up to 40 m.

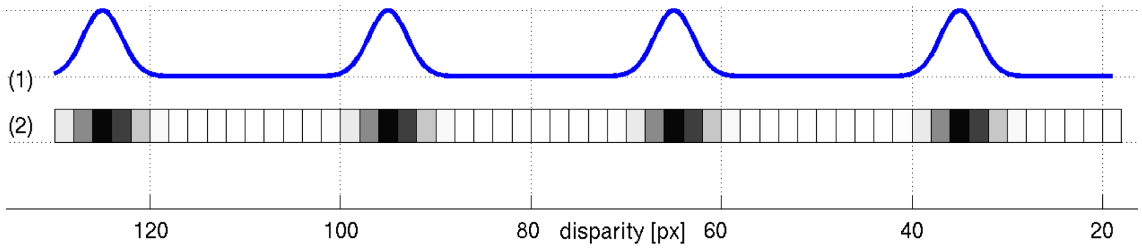


(b) The resulting column (u^*)-disparity(d^*) occupancy grid map.

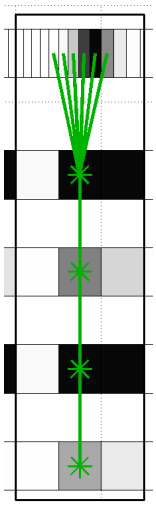
Figure 3.8: The static Stixel set $\mathcal{S}_t$ (a) and the resulting column (u^*)-disparity(d^*) occupancy grid map (b). The grid map represents the implemented measurement model at time step t. Because of the well-known uncertain behavior of stereo results only Stixels up to 40 m are considered in our approach. Because of stereo shadows and calibration errors we also neglect Stixels at the left and right border. Since we only want to map static obstacles, labeled dynamic obstacles are excluded from the mapping approach, but, nevertheless, we exploit the free space area up to the foot point of these obstacles.

are influenced by only one column-disparity cell. To solve these transformation issues, we rely on a heuristic, but easy-to-implement and real time suitable solution similar to the work of Perrollaz et al. [2010]. A combination of a maximum filter, which solves the first issue, and a bilinear interpolation, which solves the second issue, is applied. The advantage of this combination is also shown in Fig. 3.9, example (4). It can be assumed that the transformation between the column (u^*)-disparity(d^*) space and the Cartesian space is stable for a specific stereo camera calibration. This leads to the fact, that the warping procedure can be stored as a look up table.

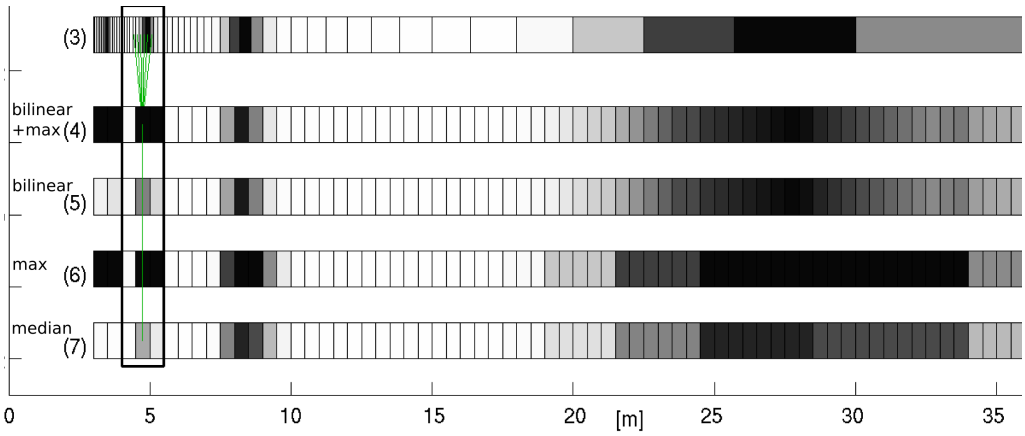
An example of the transformation of the local column-disparity grid map into the local Cartesian grid is shown in Fig. 3.10. Figure 3.11 shows the comparison between the local 2D occupancy grid map and the original triangulated 3D points which were estimated with the help of the disparity image. The figure also includes the original Stixel World. The transformation of the local Cartesian grid map into the global grid map is straight forward. We have to invert the transformation in (3.21). This step is highly parallel and, therefore, we use the power of the GPU.



(a) A continuous signal in the disparity space (1) with its discretization (2). To avoid visualization issues, the disparity sampling rate is 2 px.



(b) close-up



(c) The transformation of the discrete disparity signal (2) into a non-equidistant Cartesian space (3). Different transformations into the equidistant Cartesian grid structure are shown in (4)-(7). Method (4) shows a combination of a bilinear interpolation (5) in combination with a maximum filter (6). In method (7) a median filter is applied. Take into account that by using only the bilinear interpolation (5) or the median filter (7) obstacles within 5 m of the origin disappear which is clearly shown in the close-up in (b). The close-up in Fig. (b) also shows which grid information from (3) is used to estimate the different filter results.

Figure 3.9: Transformation of a continuous disparity signal (a) into an equidistant, discrete Cartesian space (b-c). Figure (c) shows different strategies of transforming the signal (1) into an equidistant Cartesian grid structure. Figure (b) shows the issue that occurs when several column-disparity cells “fall” into one single grid cell.

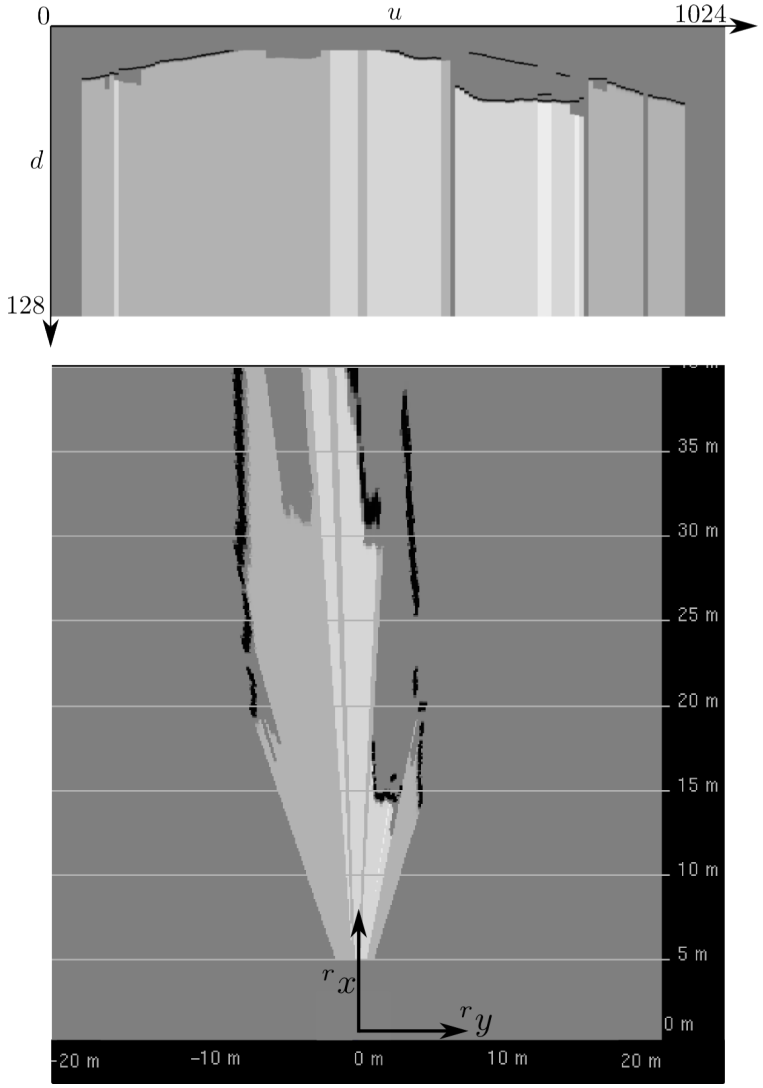
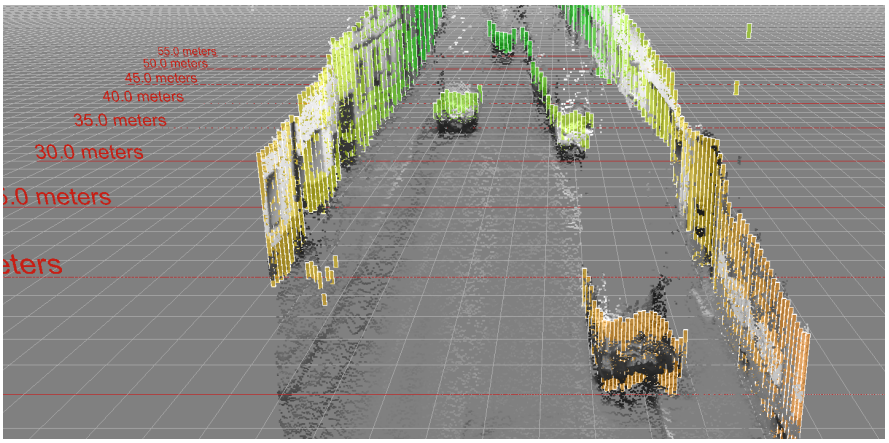


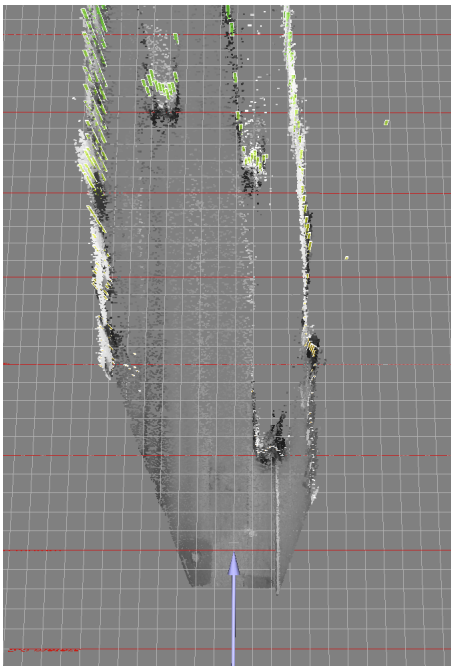
Figure 3.10: Comparison between the column (u^*)-disparity(d^*) occupancy grid map and the resulting Cartesian grid map. The Cartesian grid map is calculated in reference to the local 2D coordinate system of the ego vehicle. The dimension of the grid map is limited from $-20\,\mathrm{m}$ to $20\,\mathrm{m}$ for the ${}^r y$-axis and from $0\,\mathrm{m}$ to $40\,\mathrm{m}$ for the ${}^r x$-axis. The contour of the parked car on the right side is mapped precisely. By using the presented measurement model, it is possible to map obstacles behind other obstacles which can also be seen on the right side of the grid map. Moving obstacles are not represented in the map which increases the quality of free space.

3.4.3 The Prediction Step

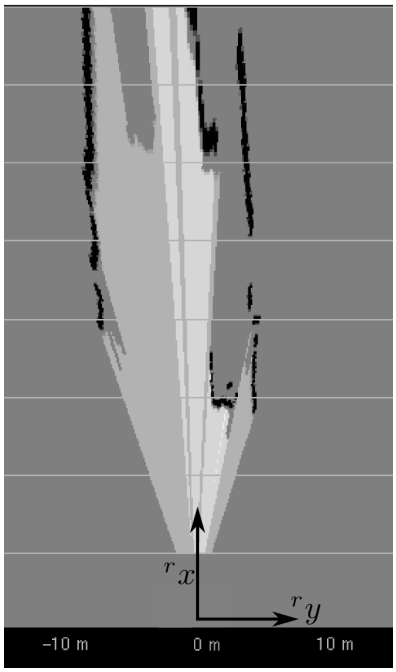
After we defined the measurement model for the recursive mapping approach, the prediction term $p(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})$ is described which represents the conditional probability of a single grid cell without taken current measurements $\mathcal{S}_t$ into account.



(a) Original 3D point cloud derived from dense disparity images and the resulting Stixels. The color of the Stixels represents the distance to the vehicle. Red stands for near by and green for far away obstacles. The perspective is about 3 m above ground level and shows the driving direction up to 50 m.



(b) Top view of Fig. (a).



(c) The resulting occupancy Cartesian grid map with reference to the relative vehicle coordinate system.

Figure 3.11: Original 3D point cloud derived from dense disparity values (a), its 2D top view (b) and the resulting occupancy Cartesian grid map (c).

3.4.3.1 Derivation

For the derivation of the prediction term we follow the steps of Sec. 2.4.1. By introducing the new variable $m_{i,t-1}$, and the use of the law of total probability, we define the prediction term $p(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})$ as:

$$p(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t}) = \int p(m_{i,t} \mid m_{i,t-1}) p(m_{i,t-1} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t}) dm_{i,t-1} \tag{3.32}$$

$$= \int p(m_{i,t} \mid m_{i,t-1}) p(m_{i,t-1} \mid \mathcal{S}_{0:t-1}, {}^{w}\boldsymbol{x}_t, \mathcal{X}_{0:t-1}) dm_{i,t-1} \tag{3.33}$$

$$= \int p(m_{i,t} \mid m_{i,t-1}) p(m_{i,t-1} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t-1}) dm_{i,t-1} \ . \tag{3.34}$$

In (3.33) we separate the current position vector ${}^{w}\boldsymbol{x}_t$ from $\mathcal{X}_{0:t}$ and make the reasonable assumption, that the position ${}^{w}\boldsymbol{x}_t$ tells us nothing about the previous state of the grid cell $m_{i,t-1}$ (see (3.34)). With the help of this assumption, the recursive structure of the estimator is derived since $p(m_{i,t-1} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t-1})$ represents the desired posterior distribution of (3.8) one time step earlier. Since we defined a single grid cell as a binary, discrete variable, (3.34) becomes:

$$\begin{aligned} P(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t}) = \ & P(m_{i,t} \mid m_{i,t-1}) \ P(m_{i,t-1} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t-1}) + \\ & P(m_{i,t} \mid \neg m_{i,t-1}) \ P(\neg m_{i,t-1} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t-1}), \end{aligned} \tag{3.35}$$

with

$$P(\neg m_{i,t-1} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t-1}) = \big(\ 1 - P(m_{i,t-1} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t-1}) \ \big) \ .$$

Equation (3.35) has the same form as (2.38) which describes the general structure of the binary prediction model of Sec. 2.4.2. In the following, we discuss the transition model as well as the desired conditional probability at time step $t-1$.

3.4.3.2 The Transition Model

The terms $p(m_{i,t} \mid m_{i,t-1})$ and $p(m_{i,t} \mid \neg m_{i,t-1})$ represent the state transition probabilities. Because of the fact that a grid cell state is discrete, binary, and follows the Markov assumption, these terms represent a two state Markov chain. This definition is previously mentioned in Sec. 2.4.2. The Markov chain is illustrated in Fig. 3.12. The transition probabilities describe the inertia of the recursive time filter. If $p(m_{i,t} \mid m_{i,t-1})$ approaches to 1, the assumption is made that the state of the grid cell does not change immediately and is temporally stable.

This statement has a direct impact on our mapping approach. If we detect static obstacles, we postulate that the state of a grid cell is stable in a time recursive way. Therefore, the transition probability $p(m_{i,t} \mid m_{i,t-1})$ should be large, e.g. 0.95. If we detect dynamic obstacles, the assumption is made that the state of a grid cell is not stable over consecutive time steps. This means that the transition probability should be small, e.g. $p(m_{i,t} \mid m_{i,t-1}) = 0.05$.

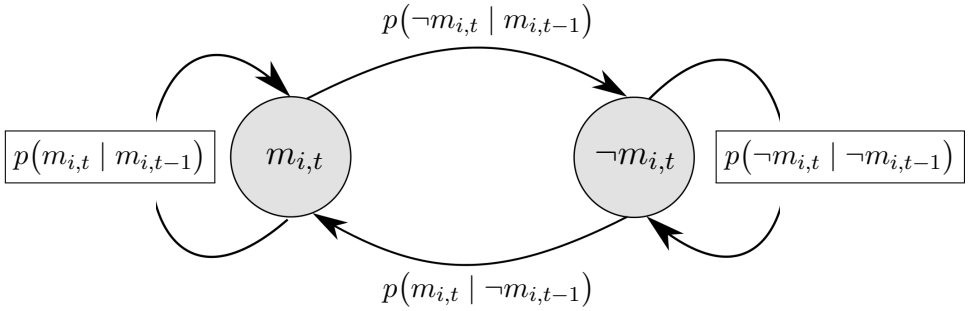
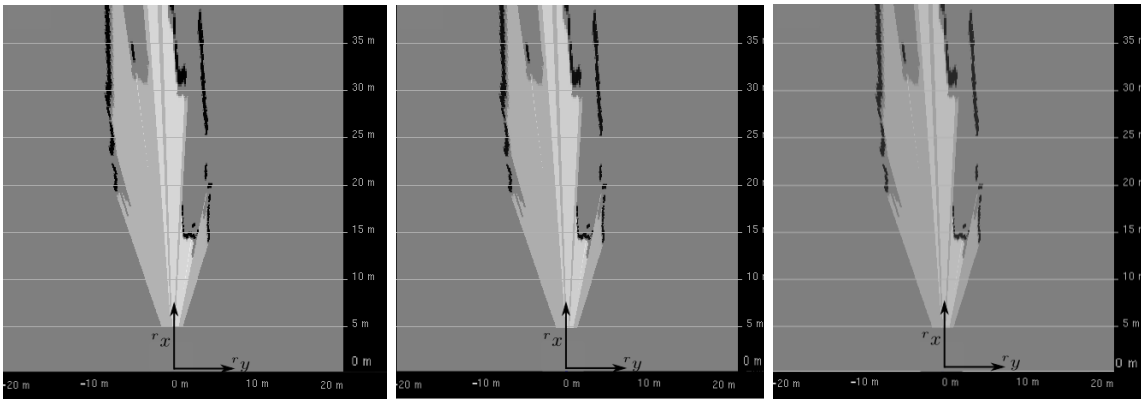
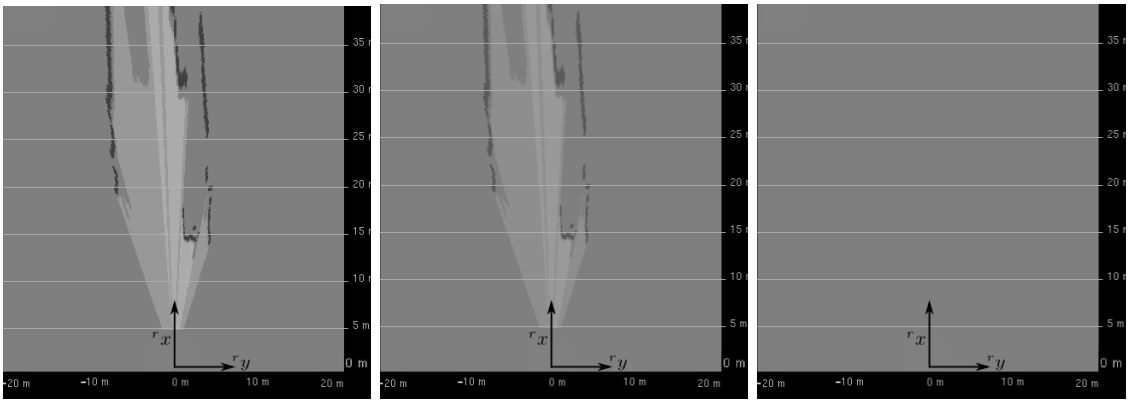


Figure 3.12: The transition model for the mapping approach for occupied grid cells $m_{i,t}$ and free grid cells $\neg m_{i,t}$. For the map prediction step, we use exactly the same model as introduced in Sec. 2.4.2 which describes the general Markovian two state model.



(a) $p(m_{i,t} \mid m_{i,t-1}) = 1$ (b) $p(m_{i,t} \mid m_{i,t-1}) = 0.95$ (c) $p(m_{i,t} \mid m_{i,t-1}) = 0.85$



(d) $p(m_{i,t} \mid m_{i,t-1}) = 0.75$ (e) $p(m_{i,t} \mid m_{i,t-1}) = 0.65$ (f) $p(m_{i,t} \mid m_{i,t-1}) = 0.50$

Figure 3.13: Example results of the prediction step using different transition probabilities $p(m_{i,t} \mid m_{i,t-1})$. In these examples as well as in the mapping approach the transition probabilities $p(\neg m_{i,t} \mid m_{i,t-1})$ are defined by $1 - p(m_{i,t} \mid m_{i,t-1})$.

In this thesis, the focus is on the first statement since we neglect dynamic obstacles in general. Different methods can be used to define the transition probabilities. In, e.g. Siegemund [2013], transition probabilities between different class types are learned from reference data which was synchronized with observations. In, e.g. Muffert et al. [2014], we evaluated different transition probabilities $p(m_{i,t} \mid m_{i,t-1})$ to find the best performance of the presented Stixel mapping approach. Here, the focus was on a robust free space estimation and a reliable obstacle detection even during difficult weather conditions. It turned out, that the best results were achieved by $p(m_{i,t} \mid m_{i,t-1}) = 0.99$. Example results of the prediction step using different transition probabilities are shown in Figure 3.13. The figure also shows the extreme cases. If a transition probability of 1 is chosen, the prediciton step has no influence. On the contrary, a transition probability of 0.5 makes the influence of former observations obsolete.

3.4.3.3 The Desired Posterior Distribution at Time Step $t-1$

The term $p(m_{i,t-1} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t-1})$ describes the posterior distribution of a single grid cell given all observations and poses at time step $t-1$. Hereby, the recursive structure of the mapping approach is achieved which implies two major aspects we have to consider: First, the term $p(m_{i,0} \mid \mathcal{S}_0, \mathcal{X}_0) = p(m_{i,0})$ at the initialization time step $t = 0$ has to be defined which represents the global prior of the grid map. This term is easy to define and is $p(m_{i,0}) = 0.5$ in general. The much more complex point is that it is necessary to estimate marginal probabilities for each single grid cell to achieve a meaningful recursive structure. This means that we cannot regard our overall optimization of (3.6) as a MAP problem because the terms $p(m_{i,t-1} \mid \mathcal{S}_{t-1}, \mathcal{X}_{0:t-1})$ and $p(\neg m_{i,t-1} \mid \mathcal{S}_{t-1}, \mathcal{X}_{0:t-1})$ would be equal to 1 or 0. Because of this fact, we introduced in Sec. 2.3.3 a general technique to estimate marginal probabilities for MRF/CRF MAP approaches. With the definition of the measurement model in (3.20) and definition of the prediction term (3.35), we define the *unary energies* $E_i(m_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t})$ by following the definition in (2.22):

$$E_i(m_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t}) = -\log\big(\ \Psi(m_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t})\ \big) \tag{3.36}$$

$$= -\log\big(\ p(m_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t})\ \big) \tag{3.37}$$

$$\propto -\log\big(\ p(\mathcal{S}_t \mid m_{i,t}, \mathcal{X}_{0:t})\ \big) - \log\big(\ p(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})\ \big). \tag{3.38}$$

The unary energies for non-existing grid cells $E_i(\neg m_{i,t} \mid \mathcal{S}_t, \mathcal{X}_{0:t})$ are defined by:

$$E_i(\neg m_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t}) \propto -\log\big(\ p(\mathcal{S}_t \mid \neg m_{i,t}, \mathcal{X}_{0:t})\ \big) - \log\big(\ 1 - p(m_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})\ \big). \tag{3.39}$$

3.5 Definition of the Binary Terms

We describe the definition of the binary terms $\Phi(m_{i,t}, m_{j,t} | \mathcal{S}_{0:t}, \mathcal{X}_{0:t})$ of the overall optimization problem in (3.6) in the following. The binary terms contain prior knowledge of the relationship between neighbored grid cells under the condition of given data. However, we formulate the binary terms as a generalized *data independent* Potts model ([Potts, 1952]) which has proven in many

computer vision applications [Kohli and Torr, 2007; Erbs et al., 2012]. Therefore, the general formulation of the binary terms can be simplified to

$$\Phi(m_{i,t}, m_{j,t}) \approx \Phi(m_{i,t}, m_{j,t} | \mathcal{S}_{0:t}, \mathcal{X}_{0:t}). \tag{3.40}$$

We postulate, that neighboring grid cells belong to the same class type. Therefore, we punish inhomogeneous regions with $m_{i,t} \neq m_{j,t}$ in the 4-neighborhood region N_4 with the index tuples (i, j) and support homogeneous regions with $m_{i,t} = m_{j,t}$. The punishing term for inhomogeneous regions is defined by the expression $-\lambda_b \ \log(1 - k_{ij})$ with $k_{ij} < 1$. The parameters k_{ij} and λ_b control the binary term which are tuned manually in this thesis. For homogeneous regions we define the punishing term as $-\lambda_b \ \log(k_{ij})$. Finally, the binary terms are defined by the following energies:

$$E_{i,j}(m_{i,t}, m_{j,t}) = -\lambda_b \ \log\big(\ \Phi(m_{i,t}, m_{j,t})\ \big) = \begin{cases} -\lambda_b \ \log(k_{ij}), & \text{if: } m_{i,t} = m_{j,t}\ , \\ -\lambda_b \ \log(1 - k_{ij}) \quad , & \text{if: } m_{i,t} \neq m_{j,t}\ . \end{cases} \tag{3.41}$$

To figure out if the statement $m_{i,t} \neq m_{j,t}$ or $m_{i,t} = m_{j,t}$ is valid, we take the probabilities of the grid cells of time step $t - 1$ into account. If $p(m_{i,t-1}) > 0.5$, we assume that the current grid cell is occupied which means $m_{i,t} = 1$. If $p(m_{i,t-1}) < 0.5$, we assume $m_{i,t} = 0$.

3.6 Incremental Map Generation via dynamic Graph Cuts and Marginal Probability Estimation

In this section we describe the incremental map generation based on dynamic graph cuts and min-marginal estimation to solve the optimization formulation of (3.6). The unary and binary energy terms $E_i(m_{i,t} \mid \mathcal{S}_t, \mathcal{X}_{0:t})$ and $E_{i,j}(m_{i,t}, m_{j,t})$ are used in this context which were previously described in Sec. 3.4 and Sec. 3.5. In general, common inference techniques like graph cuts (see Sec. 2.3.2) are used to solve our optimization problem in (3.3). Nevertheless, two major challenges occur when we want to fulfill our defined conditions in Sec. 3.1.

- **Real-time capability must be guaranteed**. Solving the optimization problem via dynamic graph cuts means that we have to create an undirected graph in which the nodes represent the grid cells. This implies that the size of the map defines the size of the graph. Building a graph over the complete map would be inefficient and, therefore, not real-time applicable. This topic is later discussed in Sec. 3.6.1.
- **An incremental mapping process must be maintained**. As mentioned above, general inference algorithms only provide MAP estimation solutions. In regard to our mapping approach, this would result in a binary classification, namely if a grid cell is occupied or free. To achieve an incremental mapping process we need marginals $p(m_{i,t-1} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t-1})$ during the prediction and update step (see Sec. 3.4.3). The estimation of marginal probabilities is solved by estimating uncertainties in graph cut solutions and is presented in Sec. 3.6.2. The basic idea of marginal probability estimation in graph cuts solutions is also described in Sec. 2.3.3.

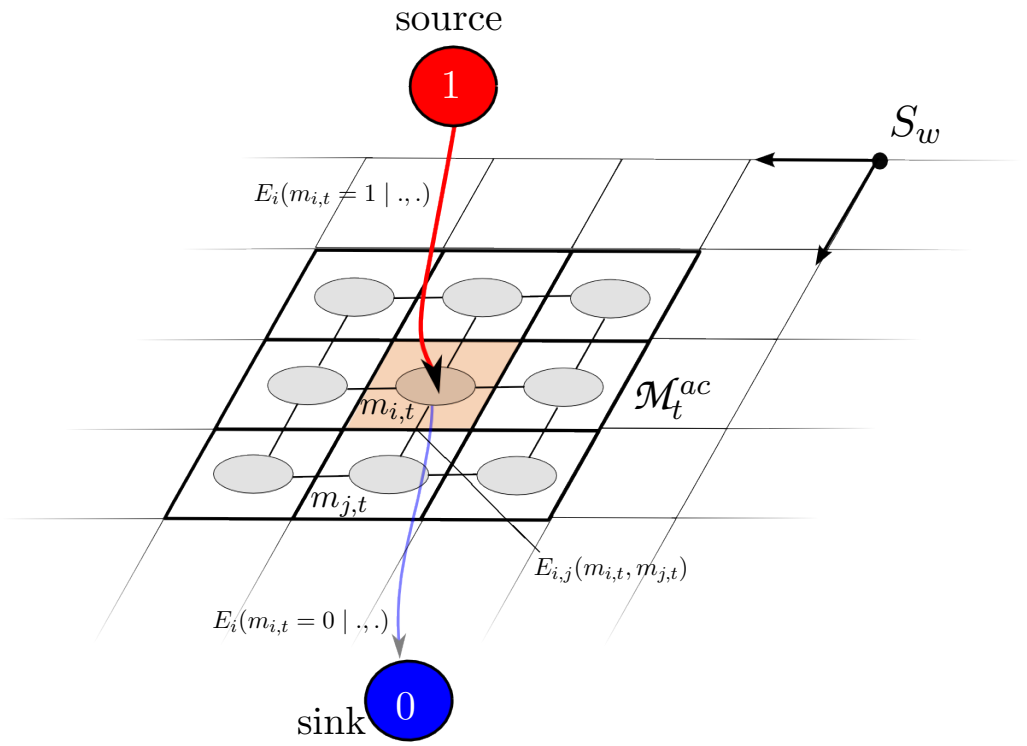


Figure 3.14: Graph structure of the *active map area* $\mathcal{M}_t^{ac}$. The active map area $\mathcal{M}_t^{ac}$ is represented by nine dark bold grid cells. The energies $E_i(m_{i,t} = 1 \mid .,.)$ respectively $E_i(m_{i,t} = 0 \mid .,.)$ represent the unary terms. The energy $E_{i,j}(m_{i,t}, m_{j,t})$ stands for the smoothness or binary term between the two grid cells $m_{i,t}$ and $m_{j,t}$.

3.6.1 Definition of the Graph Structure and its Size

Synchronously to Sec. 2.3, the conditional properties of our unknown grid cell states are described with probabilistic graphical models. Since the state of each grid cell is binary and the sub-modularity constraint is fulfilled, graph cuts are used to estimate the exact MAP solutions for our optimization problem.

Although we are not interested in the MAP solution in general, we still have to set up an undirected graph structure since we require this structure for the upcoming min marginal estimation (see Sec. 3.6.2). The nodes of the undirected graph are the single grid cells $m_{i,}$. The source node s and the sink node t represent the binary state behavior which describes whether a grid cell is occupied or free. The unary terms define the edges which connect the nodes to source or sink. To set up the graph, the unary energies defined in (3.38) and (3.39) are used. The links between the grid cells describe the mutual grid dependency and are represented by the defined binary terms of (3.41). Figure 3.14 shows the structure of the resulting graph.

Since real-time capability is an important factor, we have to consider the dimension of the previous defined graph. From our point of view, it is inefficient and unnecessary to represent the whole grid map $\mathcal{M}_t$ in a graph, since the whole size of the grid map is unknown during on-line mapping and, second, major map areas are unaffected by current measurements. This includes

covered map areas which are no longer influenced by observations as well as unknown areas which were never observed by the robot. Because of this, the graph size is based on the map area which is influenced by the current observations $\mathcal{S}_t$ in the surroundings of our robot's position.

From now, this map area is designated as the *active map area* $\mathcal{M}_t^{ac} \subseteq \mathcal{M}_t$. As an example, $\mathcal{M}_t^{ac}$ can cover an area of $40\,\mathrm{m} \times 40\,\mathrm{m}$ around the current robot position which would result in a 400×400 grid structure based on a grid cell resolution of $0.1\,\mathrm{m}$. In regard to image segmentation algorithms (e.g. [Kohli and Torr, 2007; Erbs et al., 2012]), handling optimization problems with this graph size is real-time feasible. The corresponding graph for $\mathcal{M}_t^{ac}$ is defined by $G_{M_t^{ac}}$.

3.6.2 Marginal Probability Estimation in the Graph Structure

The general estimation of marginal probabilities is described in Sec. 2.3.3. As a short reminder, the computation of marginal probabilities is essential for the prediction step described in Sec. 3.4.3 and, consequentially, essential for the time recursive map estimation. We follow (2.26)-(2.31) and Algorithm 1 in Sec. 2.3.3 for marginal probabilities estimation. This leads to a meaningful prediction step and, consequentially, to correct occupancy grid map results.

In the first step, we compute the st-mincut/max-flow of $G_{M_t^{ac}}$ which results in the residual graph $G^r_{M_t^{ac}}$ and the most probable assignment of all active grid cells in $\widehat{\mathcal{M}_t^{ac}}$:

$$\widehat{\mathcal{M}_t^{ac}} = \underset{\mathcal{M}_t^{ac}}{\operatorname{argmax}}\; p(\mathcal{M}_t^{ac}|\mathcal{S}_{0:t}, \mathcal{X}_{0:t}) \;. \tag{3.42}$$

After achieving the MAP solution, we estimate for each grid cell $m^{ac}_{i,t;j}$ with labels $j \in \{1, 0\}$ the min-marginal energies

$$\phi^{ac}_{i,t;j} = \underset{\mathcal{M}_t^{ac}, m^{ac}_{i,t}=j}{\operatorname{argmin}}\; E(\mathcal{M}_t^{ac}|\mathcal{S}_{0:t}, \mathcal{X}_{0:t}) \;, \tag{3.43}$$

and the max-marginal probabilities

$$\nu^{ac}_{i,t;j} = \frac{1}{\zeta}\; \exp\big(-\;\phi^{ac}_{i,t;j}\big) \;. \tag{3.44}$$

Here, ζ is the partition function. For the estimation of the min-marginal energies/max-marginal probabilities, we modify step-by-step the energy function $E(\mathcal{M}_t^{ac}|\mathcal{S}_{0:t}, \mathcal{X}_{0:t})$. Each active grid cell $m^{ac}_{i,t}$ is fixed to the label $j = 1$ and respectively to $j = 0$ by setting the unary energy terms of $m^{ac}_{i,t} = j$ to large values (close to infinity). This step is computationally expensive since we have to estimate a single st-mincut/max-flow solution for each active grid cell $m^{ac}_{i,t}$ and each labeling j.

To overcome this computational burden, we use *dynamic graph cuts* which was introduced by Kohli and Torr [2007]. Sec. 2.3.3 also contains in-depth information on this specific approach. Instead of creating a new graph for each single min-marginal energy estimation from scratch, dynamic graph cuts recycle the solution from the previous step. For initialization, the algorithm uses the residual graph $G^r_{M_t^{ac}}$.This type of "recursion" allows for a real-time computation of the max-marginal probabilities which was stated in [Kohli and Torr, 2007].

Finally, we are interested in the marginal probabilities $p(\nu^{ac}_{i,t;1})$ which result from the normalization of the max-marginal probabilities in regard to a specific labeling:

$$p(\nu^{ac}_{i,t;1}) = \frac{\frac{1}{\zeta}\ \exp\big(-\phi^{ac}_{i,t;1}\big)}{\frac{1}{\zeta}\ \exp\big(-\phi^{ac}_{i,t;0}\big) + \frac{1}{\zeta}\ \exp\big(-\phi^{ac}_{i,t;1}\big)} \tag{3.45}$$

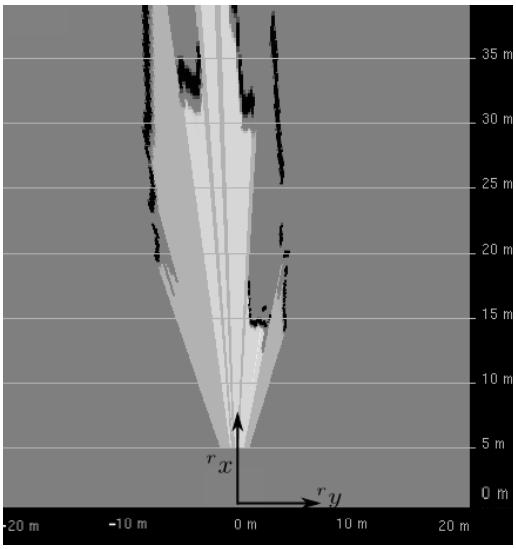
$$= \frac{\exp\big(-\phi^{ac}_{i,t;1}\big)}{\exp\big(-\phi^{ac}_{i,t;0}\big) + \exp\big(-\phi^{ac}_{i,t;1}\big)}\ . \tag{3.46}$$

In this case, the labeling $j = 1$ means we estimate the marginal probabilities for occupied grid cells. The estimation of marginal probabilities solves our previously mentioned challenge, namely the estimation of "true" probabilities for $p(m^{ac}_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t})$. Therefore, we define in the final step:

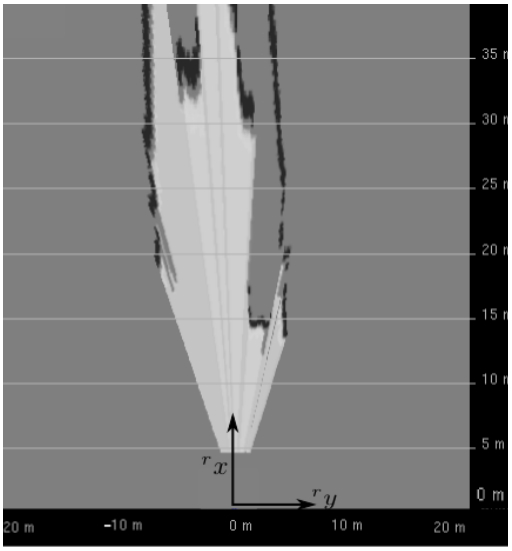
$$p(m_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t}) := p(\nu^{ac}_{i,t;1})\ . \tag{3.47}$$

Based on this definition it is possible to achieve correct results during the update and the prediction steps while we model the dependencies between grid cells explicitly. The estimation of marginal probabilities allows the realization of an incremental mapping approach based on probabilistic graphical models (MRFs) which is the core topic of this thesis. An example of these marginal probabilities $p(\nu^{ac}_{i,t;1})$ is shown in Figure 3.15. The figure also illustrates the measurement model $p(\mathcal{S}_t \mid m^{ac}_{i,t}, \mathcal{X}_{0:t})$, and the results of the prediction step $p(m^{ac}_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})$ which define the unary terms for the optimization step.

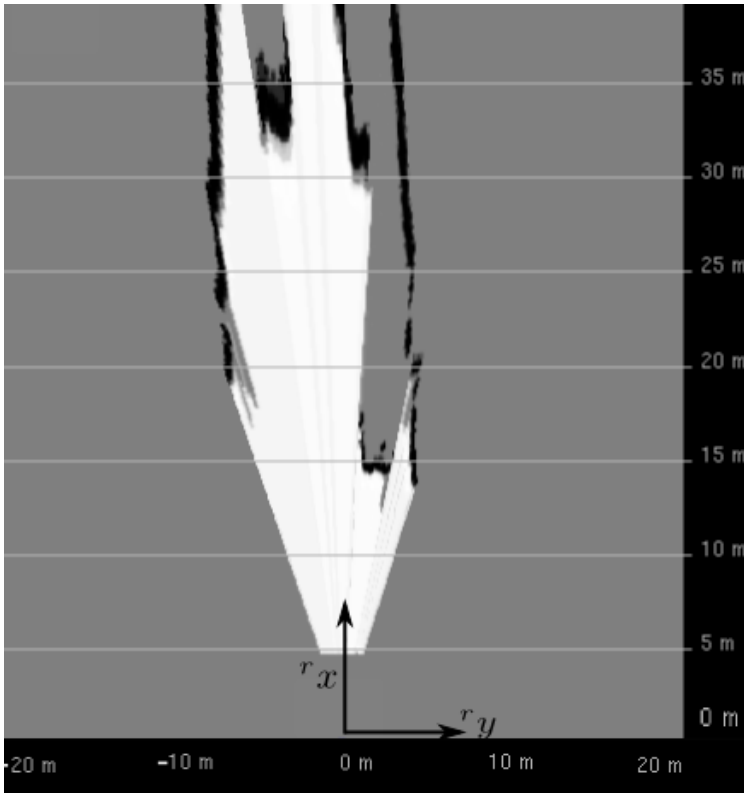
As discussed in Sec. 3.5, the binary terms are controlled by the parameters k_{ij} and λ_b. Figure 3.16 shows the results of the marginal estimation using different values of λ_b where the k_{ij} is fixed to 0.08. In this figure, we also compare the results of the marginal probability estimation with the MAP solution $\widehat{\mathcal{M}^{ac}_t}$. If we chose a $\lambda_b = 0$, the binary terms have no influence. The larger the value λ_b, the larger is the influence of the smoothing effect.



(a) Measurement model $p(\mathcal{S}_t \mid m^{ac}_{i,t}, \mathcal{X}_{0:t})$.

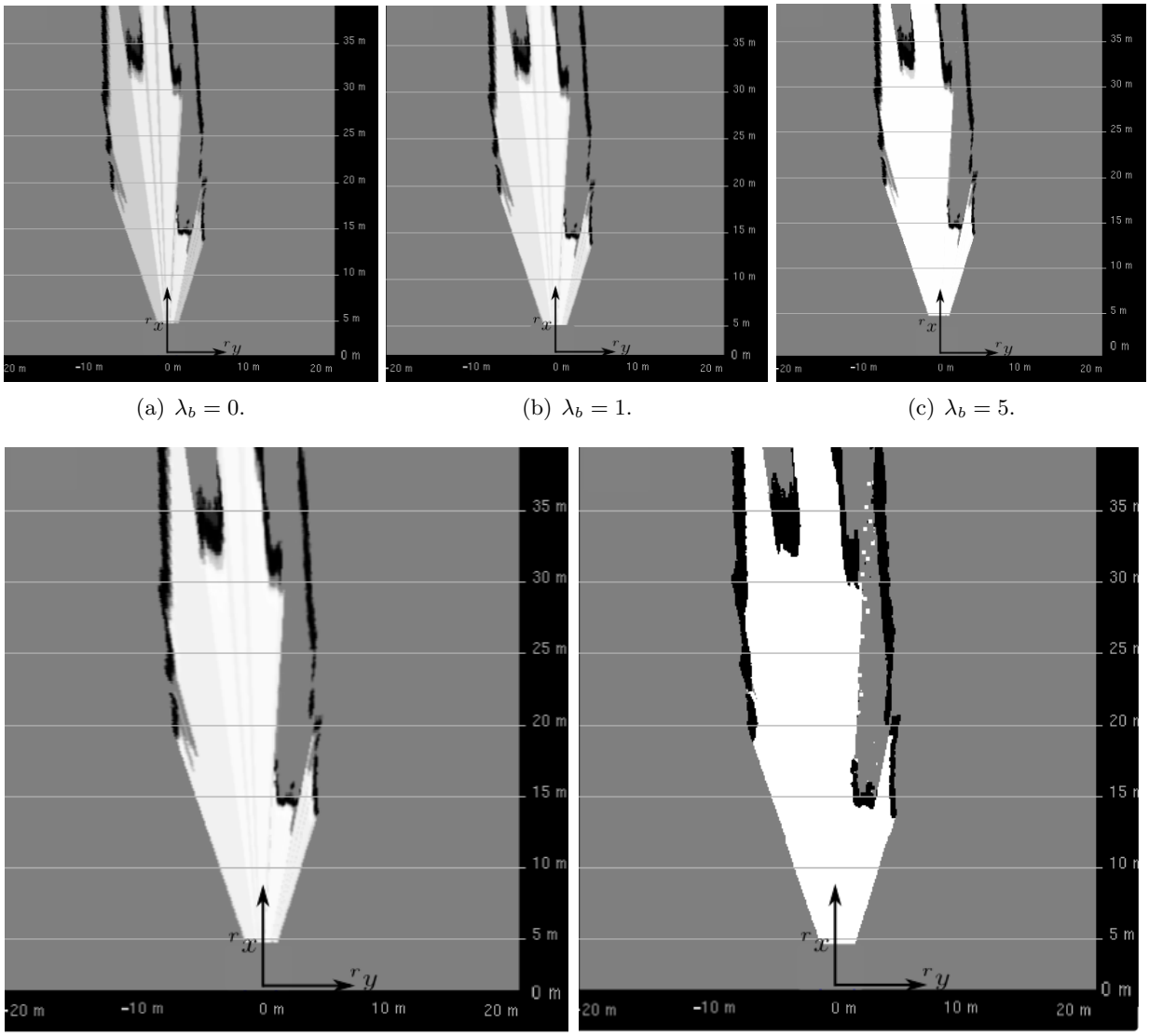


(b) Prediction $p(m^{ac}_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})$.



(c) Marginal probabilities $p(\nu^{ac}_{i,t;1})$.

Figure 3.15: The results of the measurement model $p(\mathcal{S}_t \mid m^{ac}_{i,t}, \mathcal{X}_{0:t})$ (a), the prediction step $p(m^{ac}_{i,t} \mid \mathcal{S}_{0:t-1}, \mathcal{X}_{0:t})$ (b), and the resulting marginal probabilities $p(\nu^{ac}_{i,t;1})$ (c) which are used to define the terms $p(m^{ac}_{i,t} \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t})$.



(a) $\lambda_b = 0$. (b) $\lambda_b = 1$. (c) $\lambda_b = 5$.

(d) $k_{ij} = 0.08$, $\lambda_b = 2$. (e) $\widehat{\mathcal{M}_t^{ac}}$.

Figure 3.16: Results of marginal probability estimation with different control parameters λ_b and a fixed value for $k_{ij} = 0.08$ (sub-figures (a)-(d)) in comparison to the MAP solution $\widehat{\mathcal{M}_t^{ac}}$ (sub-figure (e)). If we chose a $\lambda_b = 0$, the binary terms have no influence. The larger the value λ_b, the larger is the influence of the smoothing effect.

3.7 Implementation Details of the Overall Mapping Algorithm

Hardware and Software Details. The novel occupancy grid mapping approach is fully implemented in C++ and embedded in the software framework of the image understanding group of the Daimler Benz AG in Sindelfingen, Germany. The algorithm has only one major external dependency, namely the C++ library[1] of Kohli and Torr [2007]. This library includes the realization of dynamic graph cuts.

The presented approach is running in the test vehicle using the GPU (NVIDIA GeForce GTX 480) and CPU (Intel Core i7-980X 3.33Ghz). We also use 2 FPGA platforms for the estimation of dense disparity images via SGM and for the estimation of the Stixel World. The overall incremental mapping approach based on MRFs is presented in Algorithm 3. The global map $\mathcal{M}_{t-1}$ at time step $t-1$, the Stixel set $\mathcal{S}_t^{lab}$, and the pose ${}^w\boldsymbol{x}_t$ is defined as input. The output is the new, updated map $\mathcal{M}_t$.

In the first steps we estimate the likelihoods of the measurement model $p^{\text{meas}}_{u^*,d^*}$ and $p^{\text{free}}_{u^*,d^*}$ in the column (u^*)-disparity (d^*) space. These values are transformed into a local Cartesian Grid map. Afterwards, $\mathcal{M}_{t-1}^{ac}$ is needed for the estimation of the prediction terms. The map part $\mathcal{M}_{t-1}^{ac}$ is estimated using $\mathcal{M}_{t-1}$ and the inverse motion defined by ${}^w\boldsymbol{x}_t$. Based on the previous two steps, the unary and binary energy terms are computed to construct the undirected graph structure of the MRF. With the help of dynamic graph cuts we are allowed to estimate marginal probabilities $p(\nu_{i,t;1}^{ac})$ in an efficient way. This results in the map $\mathcal{M}_t^{ac}$. Finally, we take the motion matrix ${}^w\Delta\mathsf{M}_t({}^w\boldsymbol{x}_t)$ into account to update the map $\mathcal{M}_t$.

Runtime Behavior. The runtime behavior of the image preprocessing steps of Sec. 3.2 looks like the following: the SGM estimation as well as the Stixel World computation is performed on a FPGA platform at 40 ms. The Stixel tracking requires 35 ms and the Stixel segmentation 63 ms. The global map $\mathcal{M}_t$ is allocated on the GPU. The download and upload of $\mathcal{M}_t^{ac}$ between the CPU and GPU requires 6 ms each. The realization of the measurement model in the column (u^*)-disparity (d^*) space needs 14 ms using one CPU kernel. The transformation into the local Cartesian map is solved via look-up tables and needs less than 3 ms. The prediction step and the estimation of the energy terms requires nearly the same amount of time. The computation of the marginal probabilities using dynamic graph cuts takes 153 ms. In total, the runtime of the novel mapping approach is ≈180 ms using only one kernel of the CPU and the GPU for global grid map updates. The named runtime excludes the runtime of the preprocessing steps.

First Results. We want to finalize this chapter by showing example results of the new mapping approach. Figure 3.17 shows how the global grid map is updated over consecutive time steps. We present six different states of the global occupancy grid map for a time period of 95 image frames. For a better understanding of the scene, we also show the original gray scale images overlapped with the used Stixel sets. The Stixel segmentation needs some frames to detect dynamic obstacles (see Figure 3.17(c)) which leads to artifacts in the map. This effect is clearly shown in Figure 3.17(b), (d) and (e). However, the final example map in Figure 3.17(h) does not include these artifacts because of the used free space model for dynamic obstacles and the incremental map filtering idea.

[1] http://research.microsoft.com/en-us/um/people/pkohli/code.html, (2015-12-07)

Furthermore, the uncertainty of the Stixels can also be recognized from the figure. Far away static obstacles are mapped quite coarsely and "fuzzy" at first. When the ego vehicle approaches these obstacles, the contour becomes crisp and sharp. The reason is that the pose precision of the Stixels is getting better. These effects are clearly shown for the parked cars on the right side of the road. The described behavior corresponds to the uncertain behavior of stereo vision and the Stixel World, respectively. In Chapter 5 we show more qualitative results and point out the advantage of using the novel mapping model based on MRFs.

Algorithm 3: The Overall incremental mapping approach.

Input: New observations defined by the Stixel set $\mathcal{S}_t^{lab}$, the global ego vehicle pose ${}^w\boldsymbol{x}_t$, previous map result $\mathcal{M}_{t-1}$.

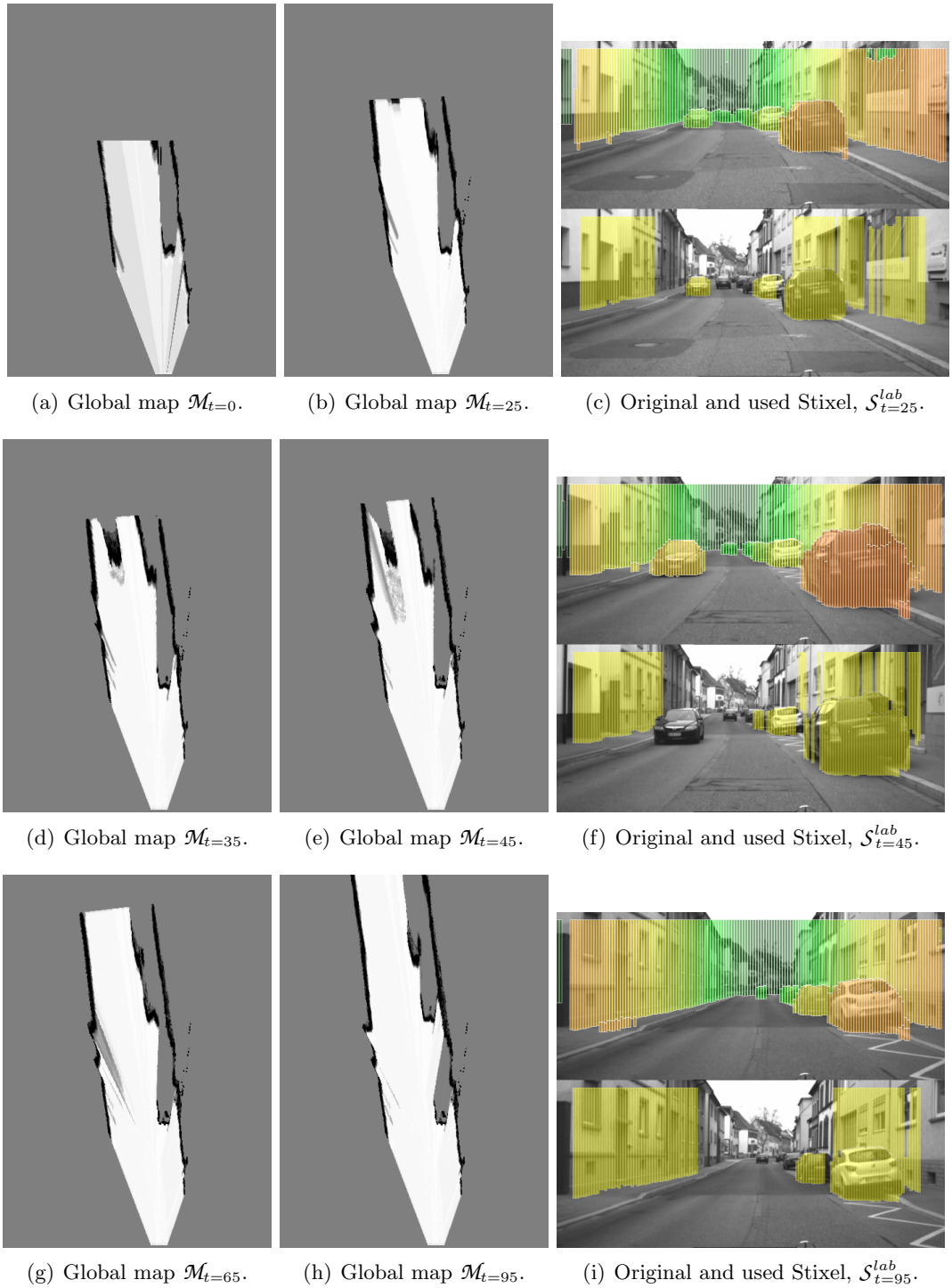
Output: New map $\mathcal{M}_t$.

Estimation of the likelihoods in the column (u^*)-disparity (d^*) space:
for *u^* = 0 to U (Nr. of columns)* **do**
  for *d^* = 0 to D (Nr. of disparity values)* **do**
    $p^{\text{meas}}_{u^*,d^*} = 1$; $p^{\text{free}}_{u^*,d^*} = 1$;
    for *k = 0 to K (Nr. of all Stixels in $\mathcal{S}_t^{lab}$)* **do**
      Estimate likelihoods ((3.27) and (3.29)) in consideration of defined disparity intervals:

$$p^{\text{meas}}_{u^*,d^*} = p^{\text{meas}}_{u^*,d^*} \cdot \left[\alpha\, p_{\text{meas}}\left(d^{lab}_{un,t} \mid d^*, \sigma^2_{d_{m_{i,t}}}\right) + \left(1-\alpha\right) p_{\text{out}}(d^{lab}_{un,t} \mid d^*)\right];$$

$$p^{\text{free}}_{u^*,d^*} = p^{\text{free}}_{u^*,d^*} \cdot \left[\alpha\, p_{\text{free}}\left(d^{lab}_{un,t} \mid d^*, \sigma^2_{d_{m_{i,t}}}\right) + \left(1-\alpha\right) p_{\text{out}}(d^{lab}_{un,t} \mid d^*)\right];$$

Mapping of $p^{\text{meas}}_{u^*,d^*}$ and $p^{\text{free}}_{u^*,d^*}$ into a local Cartesian Grid by following Sec. 3.4.2.2; This results in $p(\mathcal{S}_t^{lab} \mid m^{ac}_{i,t}, {}^w\mathcal{X}_{0:t})$ and $p(\mathcal{S}_t^{lab} \mid \neg m^{ac}_{i,t}, {}^w\mathcal{X}_{0:t})$;
Estimation of the prediction terms $p(m^{ac}_{i,t} \mid \mathcal{S}^{lab}_{0:t-1}, {}^w\mathcal{X}_{0:t})$ using $\mathcal{M}^{ac}_{t-1}$ by following (3.35);
Estimation of the unary energies $E_i(m^{ac}_{i,t} \mid \mathcal{S}^{lab}_{0:t}, {}^w\mathcal{X}_{0:t})$ using (3.38) and (3.39);
Estimation of the binary energies $E_{i,j}(m^{ac}_{i,t}, m^{ac}_{j,t})$ by following (3.41);
Compute probabilities $p(\nu^{ac}_{i,t;1})$ by using dynamic graph cuts presented in Algorithm 1:
Initialize confidence vector: $\boldsymbol{c} = \emptyset$
for *i = 1 to I (Nr. of grid cells of $\mathcal{M}_t^{ac}$)* **do**
  for *j = 0 to 1* **do**
    compute min-marginal energies ϕ_{ij} (see (3.43));
  Estimate marginal probabilities $p(\nu^{ac}_{i,t;1})$ (see (3.46));
  Set: $p(m_{i,t} \mid \mathcal{S}^{lab}_{0:t}, {}^w\mathcal{X}_{0:t}) := p(\nu^{ac}_{i,t;1})$;
  Save: $\boldsymbol{c} = \boldsymbol{c} + [p(m_{i,t} \mid \mathcal{S}^{lab}_{0:t}, {}^w\mathcal{X}_{0:t})]$;
Transform $\boldsymbol{c}$ into $\mathcal{M}_t$ using index function f_i in Sec. 3.4.2.1 and the motion ${}^w\mathsf{M}_t({}^w\boldsymbol{x}_t)$.
return updated map $\mathcal{M}_t$.



(a) Global map $\mathcal{M}_{t=0}$. (b) Global map $\mathcal{M}_{t=25}$. (c) Original and used Stixel, $\mathcal{S}^{lab}_{t=25}$.

(d) Global map $\mathcal{M}_{t=35}$. (e) Global map $\mathcal{M}_{t=45}$. (f) Original and used Stixel, $\mathcal{S}^{lab}_{t=45}$.

(g) Global map $\mathcal{M}_{t=65}$. (h) Global map $\mathcal{M}_{t=95}$. (i) Original and used Stixel, $\mathcal{S}^{lab}_{t=95}$.

Figure 3.17: Example of the novel incremental mapping approach with MRFs for 95 time steps. The figure shows six different states of the global map $\mathcal{M}$. We also present the gray scale images and the original Stixel World and the used Stixel sets $\mathcal{S}^{lab}$ for a better understanding of the scene. Dynamic obstacles are recognized and filtered out correctly which results in reasonable free space areas. Parked cars on the right side are very "fuzzy" at first. As the ego vehicle approaches these obstacles, their contours sharpen. This is due to the natural uncertain behavior of the Stixels and the effect of the incremental map update scheme.

Chapter 4

Incremental Mapping using Uncertain Poses

In this Chapter we present our realization of incremental occupancy grid mapping under the assumption that the pose is uncertain. This leads to the well known SLAM problem. At the beginning, we describe our motivation (Sec. 4.1.1), and the requirements of the SLAM approach (Sec. 4.1.2). In Sec. 4.2 the realization of the Rao-Blackwellized particle filter (RBPF) is presented which is finally used to solve the SLAM problem in this context. We also present the definition of the motion model (Sec. 4.3), the definition of the observation model (Sec. 4.4), and the adaptive resampling scheme (Sec. 4.5).

4.1 Introduction

In this section we discuss our motivation first, and present the requirements of the SLAM approach afterwards. We also state which SLAM approach is used in this chapter.

4.1.1 Motivation

In the previous chapter we optimized the posterior distribution $p(\mathcal{M}_t \mid \mathcal{S}_{0:t}, \mathcal{X}_{0:t})$ with MRFs under the assumption that the pose information $\mathcal{X}_{0:t}$ is *given*. This results in a pure mapping scheme without any pose optimization. As proposed in e.g. [Muffert et al., 2014; Nuss et al., 2015], this assumption is acceptable for small environments and for ego vehicle centered grid maps as long as sensor based ego motion estimation [Badino et al., 2013], or the odometry information is precise.

To map large scale environments with potentially *loop closures* this assumption is untenable and leads to inconsistent global maps. At this point loop closing means that the vehicle is able to correctly postulate that it returns to a previously seen scene. To create consistent maps, it is necessary to optimize both the desired map *and* the vehicle's pose which leads to SLAM problems. The SLAM problem was already mentioned and discussed in the Secs. 2.6-2.8. For clarification, we present results of global occupancy grid maps with and without pose optimization in Fig. 4.1. Figure 4.1(a) shows an occupancy grid map under the assumption that the pose is correct. Because of odometry drift behavior, inconsistencies occur during entering the same area at multiple times.

On the other hand, Fig. 4.1(b) shows the result of the subsequent approach which optimize both, the map and the poses. This results in a consistent map without any ambiguities. Consistent maps are strong requirements for following applications like localization or for path planning. Figure 4.1(c) also shows the original environment overlayed with the optimized driven path as well as the start and end point of the vehicle (red circle).

4.1.2 Requirements of the SLAM Approach

To solve the SLAM problem, specific requirements should be fulfilled to be consistent with the overall concept of this thesis. These requirements are listed as follows.

- **Probabilistic formulation.** Since the previous mentioned mapping approach is formulated in a probabilistic fashion, the SLAM approach should also be modeled in this way.
- **Online capable.** We prefer a SLAM technique which is potentially on-line capable in order to run in our test vehicles.
- **Reusing the Mapping approach.** The desired SLAM approach should be based on grid maps which means that we are able to use the previous mentioned mapping approach of Chapter3.

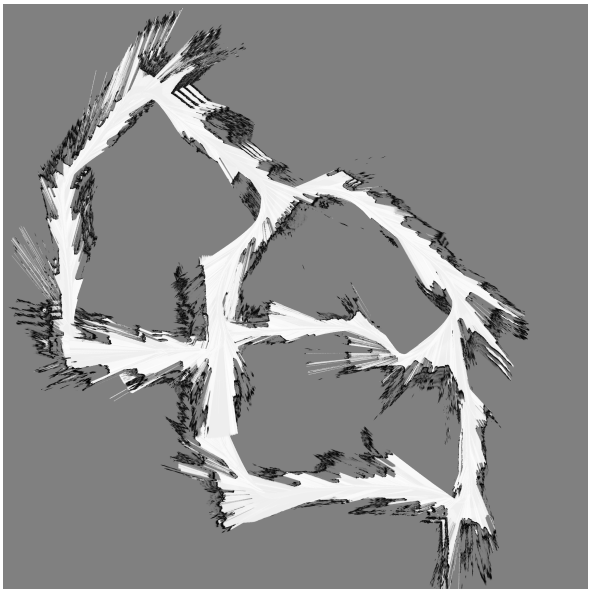
Next we formulate the optimization function and also choose the SLAM technique which fits best to the defined requirements.

4.1.3 Probabilistic Formulation and Selection of the SLAM Technique

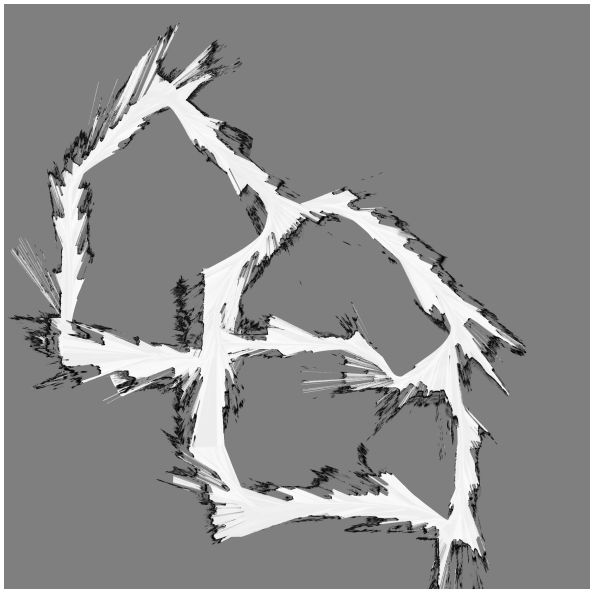
The Optimization of both, the map $\mathcal{M}_t$ and the pose $\mathcal{X}_{0:t}$, given the observations $\mathcal{S}_{0:t}$ and the control information $\mathcal{U}_{0:t-1}$ leads to the estimation of the posterior $p(\mathcal{M}_t, \mathcal{X}_{0:t} | \mathcal{S}_{0:t}, \mathcal{U}_{0:t-1})$. Rao-Blackwellized particle filters (RBPFs), which were already presented in Sec. 2.7.2, are well suited to fulfill the above mentioned requirements. The optimization task is formulated in a probabilistic way, particle filters are on-line feasible, and the new grid mapping technique can be used. Therefore, we decide to apply grid-based RBPFs in this thesis to solve the on-line SLAM problem. RBPFs separate the estimation of the trajectory from the estimation of the map by following the Rao-Blackwellized idea (Sec. 2.7.2). Using (2.45), the posterior $p(\mathcal{M}_t, \mathcal{X}_{0:t} | \mathcal{S}_{0:t}, \mathcal{U}_{0:t-1})$ is decoupled in

$$p(\mathcal{M}_t, \mathcal{X}_{0:t} | \mathcal{S}_{0:t}, \mathcal{U}_{0:t-1}) = \underbrace{p(\mathcal{X}_{0:t} | \mathcal{S}_{0:t}, \mathcal{U}_{0:t-1})}_{\textit{pose posterior}} \underbrace{p(\mathcal{M}_t | \mathcal{S}_{0:t}, \mathcal{X}_{0:t})}_{\textit{map posterior}} . \tag{4.1}$$

The map posterior is solved in a closed form using the occupancy grid mapping approach explained in Chapter3. From now, the focus is on solving the pose posterior $p(\mathcal{X}_{0:t} | \mathcal{S}_{0:t}, \mathcal{U}_{0:t-1})$ with the help of particle filters. Details of particle filters are presented in Sec. 2.7.1. Recent parts of the following work were published in Dömötör [2014] which was supervised by the author of this thesis. In contrast to work of Dömötör [2014], we use the previously described grid mapping approach with MRFs in the realization of the RBPFs.



(a) Occupancy grid map without pose optimization.



(b) Occupancy grid map with pose optimization.

(c) Google earth image of the mapped environment and the optimized driven path (blue) with its start and end point (red).

Figure 4.1: Comparison between occupancy grid maps without (4.1(a)) and with pose optimization (4.1(b)). In Figure 4.1(b) a consistent map is shown whereas ambiguities are clearly visible in Figure 4.1(a), e.g. at the middle-left border or in the upper left area of the map. The satellite image in Figure 4.1(c) was taken from Google Maps and shows the residential area in Böblingen, Germany. In total, we drove 1.9 km with our test vehicle *S 500 Intelligent Drive* (see Sec. 3.2.1).

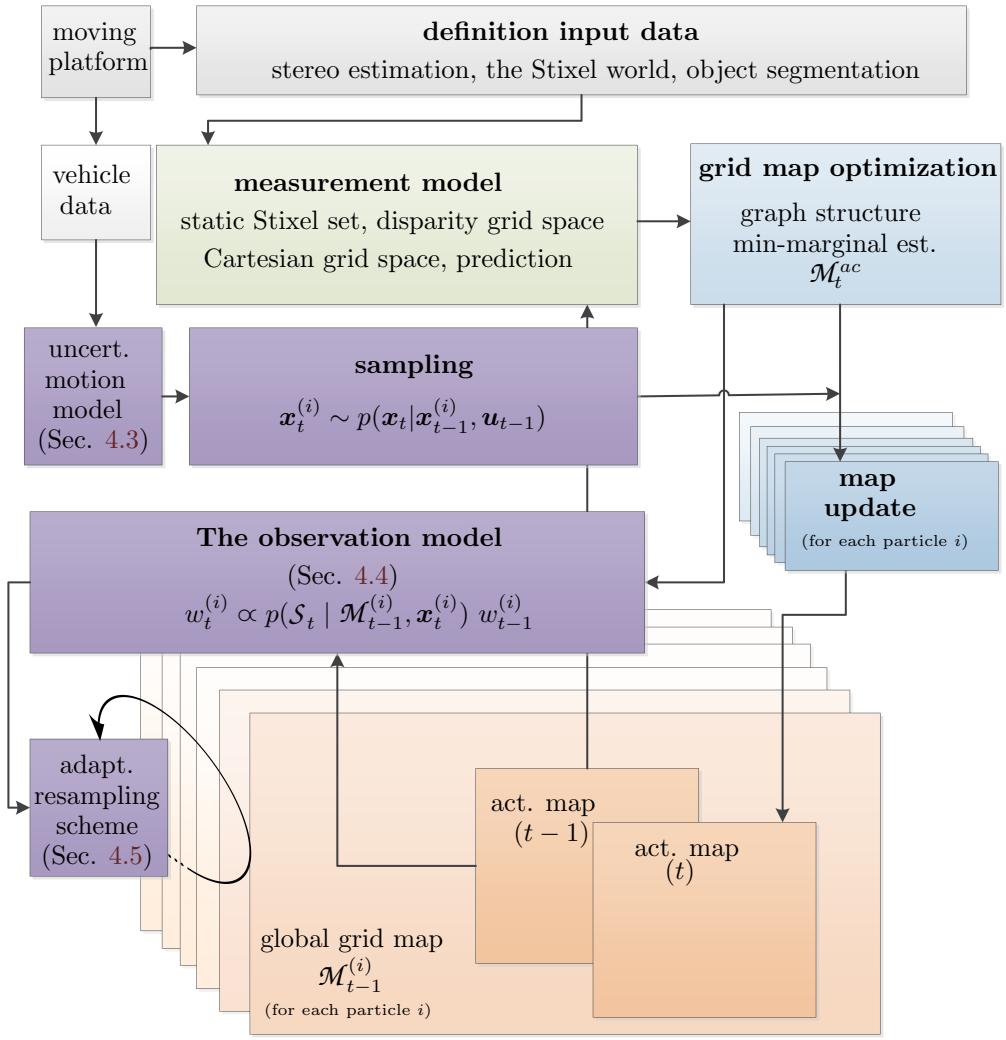


Figure 4.2: The overview of the realization of the grid based SLAM particle filter. In contrast to the novel mapping approach described in Chapter 3 we model the motion model with uncertainties. In detail, sampling with the uncertain motion model is carried out with $\boldsymbol{x}_t^{(i)} \sim p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}^{(i)}, \boldsymbol{u}_{t-1})$. The weight estimation of the different particles is based on the observation model $p(\mathcal{S}_t \mid \mathcal{M}_{t-1}^{(i)}, \boldsymbol{x}_t^{(i)})$. For each particle a map update is done which results in different global maps with different weights (represented by the brightness of the maps). Based on the map and the distribution of the weights an adaptive resampling scheme is used during this thesis.

4.2 Realization of the Rao-Blackwellized Particle Filter

Figure 4.2 shows the overall process of the realization of the RBPF. RBPFs follow the idea of SIR particle filters which major steps are sampling, importance weighting and resampling (see Sec. 2.7.1.2). Therefore, we extend the overview shown in Fig. 3.2 by an uncertain motion model (sampling), an observation model (weighting) and a resampling procedure.

Following the idea of grid-based RBPFs, the particle filter set $\mathcal{P}_t$ is defined by the individual poses $\boldsymbol{x}_t^{(i)}$, the individual maps $\mathcal{M}_t^{(i)}$, and the weights $w_t^{(i)}$ at time t

$$\mathcal{P}_t := \{(\boldsymbol{x}_t^{(i)}, \mathcal{M}_t^{(i)}, w_t^{(i)})\}, \;\; \text{with} 1 \leq i \leq I \;, \tag{4.2}$$

which was already stated in (2.46). Because of the definition in (4.2), Fig. 4.2 also shows I different global grid maps where each map i is updated with the same measurements but with different pose information. The observation model "compares" the current measurements with the maps i which results in the different weights. Depending on the distribution of the weights, resampling is carried out. In the following we discuss the steps of the SIR particle filter in detail.

4.3 Sampling via Odometry Motion Model

As discussed in previous work [Bosse et al., 2003; Grisetti et al., 2007], sampling via the odometry motion model led to poor results since the control information of their robots were not reliable (see Fig. 4.3). Therefore, Grisetti et al. [2007] proposed a method which also incorporates the map as well as precise LIDAR sensor readings into the proposal distribution.

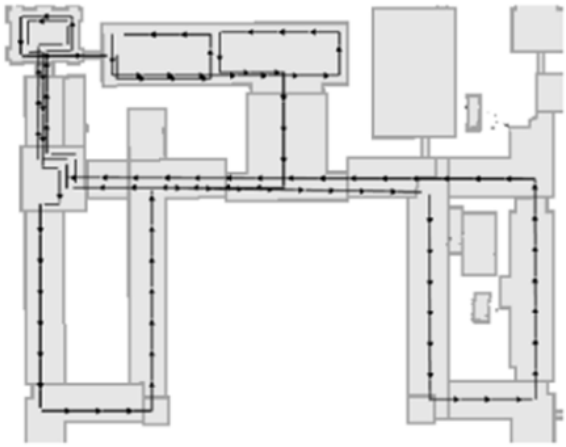
However, the statement with regard to the odometry motion model is not applicable in our thesis due to the following reasons. First, the odometry measurements and the motion behavior of our current test vehicle are much more precise than the motion behavior of robots used by e.g. [Bosse et al., 2003]. This is clearly shown in Fig. 4.3. Take into consideration that in our test vehicle only the serial production odometry information is used and not a high precision IMU-GPS measurement system.

Second, the precision of our sensor readings has not the same quality than LIDAR sensor readings. This is caused by the limited field of view and the distance dependent accuracy behavior of stereo vision (see Sec. 2.2.2.3). Therefore, we use the odometry motion model $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}^{(i)}, \boldsymbol{u}_{t-1})$ to sample the next possible poses:

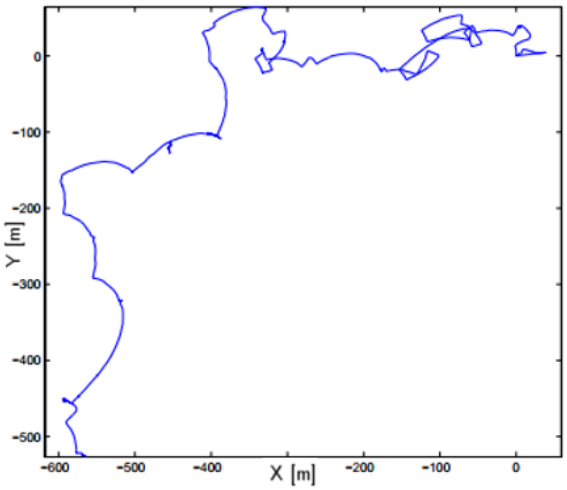
$$\boldsymbol{x}_t^{(i)} \sim p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}^{(i)}, \boldsymbol{u}_{t-1}) \;. \tag{4.3}$$

We apply the motion model which was defined in Chapter 3, Sec. 3.2.2. It is governed by the control vector $\boldsymbol{u}_t = [v, \dot{\varphi}, \Delta t]^\mathsf{T}$. In contrast to the previous chapter, we now model the uncertain behavior of the forward velocity v and the yaw rate $\dot{\varphi}$.

In Dömötör [2014] empirical studies were carried out to estimate empirically the variance of the yaw rate $\sigma_{\dot{\varphi}_t}^2$. For this purpose, a long term measurement of the yaw rate was done during the test vehicle stood still. It turned out that the noise behavior of the yaw rate follows nearly a biased Gaussian distribution.

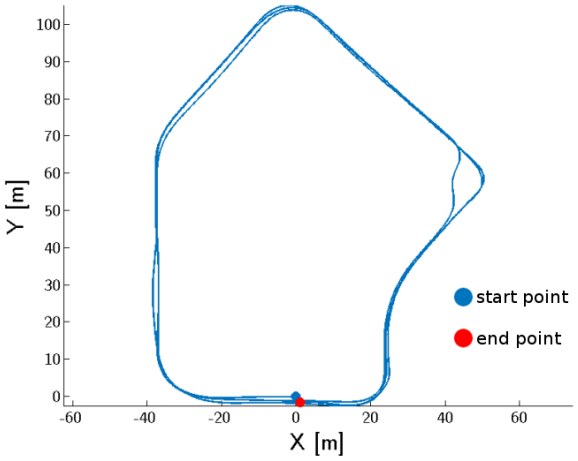


(a) Driven path at MIT-Campus ("Infinite Corridor") Bosse et al. [2003].



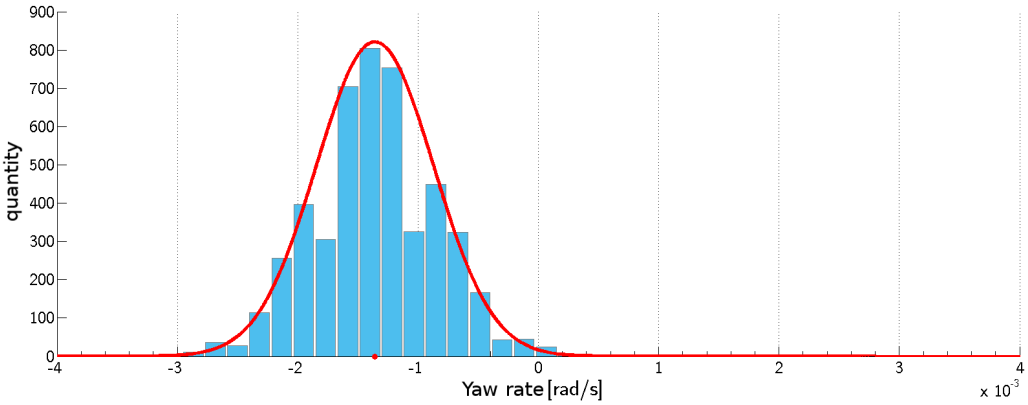
(b) Reconstructed path based on raw odometry data Bosse et al. [2003].

(c) Driven path of our test vehicle *S 500 Intelligent Drive* in a residential area in Böblingen, Germany.

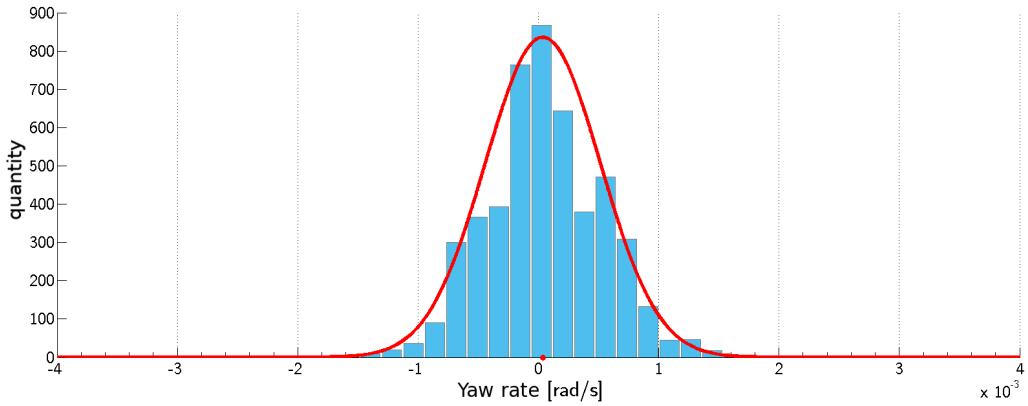


(d) Reconstructed path of our test vehicle *S 500 Intelligent Drive* based on raw odometry data.

Figure 4.3: Comparison of the motion behavior of a standard B21 mobile robot Bosse et al. [2003] and our test vehicle presented in Sec. 3.2.2. In Fig. 4.3(a) the original driven path of the B21 robot is shown. Figure 4.3(b) shows the reconstructed path based on its odometry information. Figure 4.3(c) and Fig. 4.3(d) show an example of the odometry behavior of our test vehicle. It is clearly shown that the control information of our test vehicle is much more precise than the one of the B21.



(a) Distribution of the yaw rate $\dot{\varphi}_t$ without offset correction.



(b) Distribution of the yaw rate $\dot{\varphi}_t$ with offset correction.

Figure 4.4: Distribution of the measured yaw rate while the test vehicle stood still. In (a), measurements without an offset correction are shown. After the estimation of the systematic offset of $-1.3\text{e}^{-3}\frac{\text{rad}}{\text{s}}$ the behavior of the yaw rate is approximately mean free. In both cases, the data in (a) and (b) represents nearly a Gaussian distribution which is represented by the red line fit.

Figure 4.4 shows the distribution of the yaw rate with and without offset correction. The noise behavior of the velocity is also be assumed to be Gaussian although no empirical studies were carried out. Here, $\sigma_{v_t}^2$ is tuned by hand. In summary, we define that the individual velocities $v_t^{(i)}$ and yaw rates $\dot{\varphi}_t^{(i)}$ can be sampled by

$$v_t^{(i)} \sim \mathcal{N}(v_t, \sigma_{v_t}^2) \quad \text{and} \tag{4.4}$$

$$\dot{\varphi}_t^{(i)} \sim \mathcal{N}(\dot{\varphi}_t, \sigma_{\dot{\varphi}_t}^2) \ . \tag{4.5}$$

Based on (4.4) and (4.5), the motion description of Sec. 3.2.2, and the sampling idea described in Thrun et al. [2005, chapter 5.3.2], sampling new pose hypothesis is straight forward.

4.4 Importance Weighting via Observation Model

Following the idea of RBPFs (see Sec. 2.7.2, (2.48)), the individual weights $w_t^{(i)}$ are recursively estimated by

$$w_t^{(i)} \propto p(\mathcal{S}_t|\mathcal{M}_{t-1}^{(i)}, \boldsymbol{x}_t^{(i)})\ w_{t-1}^{(i)}\ . \tag{4.6}$$

Here, the posterior $p(\mathcal{S}_t|\mathcal{M}_{t-1}^{(i)}, \boldsymbol{x}_t^{(i)})$ defines the observation model which is realized by a grid based map matching technique in this thesis. Following the definitions of the previous presented mapping approach we use the current observations $\mathcal{S}_t$ to estimate the Cartesian occupancy grid map $\mathcal{M}_t^{ac}$. Afterwards, $\mathcal{M}_t^{ac}$ is matched to all global maps $\mathcal{M}_{t-1}^{(i)}$ to estimate the individual weights.

Next to correlation based approaches [Konolige and Chou, 1999], score based techniques [Schröter et al., 2007] are often used to estimate how good grid maps fit to each other. Here, we choose the latter technique which was successfully used in the work of Dömötör [2014]. How good static environment information of both maps is aligned to each other is defined by the matching score $\rho^{(i)}_{\mathcal{M}_t^{ac},\mathcal{M}_{t-1}^{(i)}}$. The better $\mathcal{M}_t^{ac}$ aligns with $\mathcal{M}_{t-1}^{(i)}$, the better should be the score value. We only increment the score value, if both grid cells $m_{k,t}^{ac}$ and $m_{k,t-1}^{(i)}$ are occupied. The index k stands for the *k-th* element of all grid cells K of $\mathcal{M}_t^{ac}$ and $\mathcal{M}_{t-1}^{(i)}$, respectively. If a difference in both grid cell arguments exists, we decrease the score to punish misalignments. Free space area is not considered during the score estimation. Mathematically, the score value $\rho^{(i)}_{\mathcal{M}_t^{ac},\mathcal{M}_{t-1}^{(i)}}$ is defined by

$$\rho^{(i)}_{\mathcal{M}_t^{ac},\mathcal{M}_{t-1}^{(i)}} = \sum_{k=1}^{K} \begin{cases} 1 & \text{if} \quad \left(\ p(m_k^{(i)}) > 0.5\ \right) \wedge \left(\ p(m_{k,t}^{ac}) > 0.5\ \right) \\ -1 & \text{if} \quad \left(\ p(m_k^{(i)}) < 0.5\ \right) \wedge \left(\ p(m_{k,t}^{ac}) > 0.5\ \right) \\ -1 & \text{if} \quad \left(\ p(m_k^{(i)}) > 0.5\ \right) \wedge \left(\ p(m_{k,t}^{ac}) < 0.5\ \right) \\ 0 & \text{else}\ . \end{cases} \tag{4.7}$$

Here, $p(m_k^{(i)})$ is the probability of the single grid cell m_k taken from the individual global maps $\mathcal{M}_{t-1}^{(i)}$. The probability $p(m_{k,t}^{ac})$ is taken from the observation map $\mathcal{M}_t^{ac}$.

The score value in (4.7) is a large positive integer, if the global and the observation map are very similar. Therefore, the weight of this particle should also be high. On the other hand, the score value is a large negative integer, if the global and the observation map are misaligned. According to this fact, the weight of the current particle should be small. Based on these facts, the desired observation model is approximated by an exponential function with

$$p(\mathcal{S}_t|\mathcal{M}_{t-1}^{(i)}, \boldsymbol{x}_t^{(i)}) \approx \exp(\alpha\ \rho^{(i)}_{\mathcal{M}_t^{ac},\mathcal{M}_{t-1}^{(i)}}), \tag{4.8}$$

where α is a tuning parameter which has to be defined empirically. As stated in [Schröter et al., 2007], the parameter α influences the spread in the particle weights and, consequently, influences the convergence behavior of the particle filter. The score value finds its maximum if the observation map $\mathcal{M}_t^{ac}$ fits nearly perfectly to one of the maps $\mathcal{M}_{t-1}^{(i)}$ of the particle set. Especially by re-entering previous mapped regions (loop closures) this should be the case.

Even though the score based map matching technique is reliable, challenges occur which were already mentioned in [Dömötör, 2014]. Due to the limited field of view of the stereo camera system, we only match a small and limited map area in front of the ego vehicle during the estimation of the score values. This is a major drawback compared to approaches which use 360 deg LIDAR scans. Furthermore, the accuracy behavior of stereo vision (see Sec. 2.2.2.3) leads to the fact that the resulting observation grid maps $\mathcal{M}_t^{ac}$ are only very precise and crisp near by the ego vehicle's position. This was already presented in Sec. 3.7 and Fig. 3.17.

To overcome the mentioned burden the weight estimation runs with a lower frequency than the sampling of the poses. During the suspension of the weight estimation, the Stixel sets are integrated into i different smaller, local grid maps using the different particle poses $\boldsymbol{x}_t^{(i)}$. This allows us to create a larger field of view of the local environment around the ego vehicle. Furthermore, the precision of the local maps increases, as already shown and discussed in Fig. 3.17. For the score, and consequently for the weight estimation these local grid maps are matched against the global maps $\mathcal{M}_{t-1}^{(i)}$. Simply put, we use the environment information near by or even behind the ego vehicle for the weight estimation. Therefore, also the global map update for each particle runs with the same frequency than the weight estimation step. The definition of the parameters is discussed during evaluation in Sec. 6.1.

4.5 The Adaptive Resampling Scheme

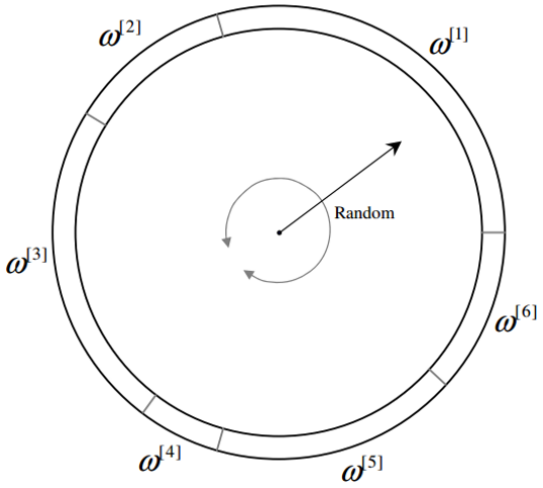
As described in Sec. 2.7.2 we use adaptive resampling in this thesis. This means we only apply resampling if it is really needed. This helps to avoid the particle deprivation problem [van der Merwe et al., 2000; Doucet et al., 2001; Grisetti et al., 2007]. As defined in (2.49), the effective number of particles N_{eff} is estimated by

$$N_{\text{eff}} = \left[\sum_{i=1}^{N} \left(w^i \right)^2 \right]^{-1} . \tag{4.9}$$

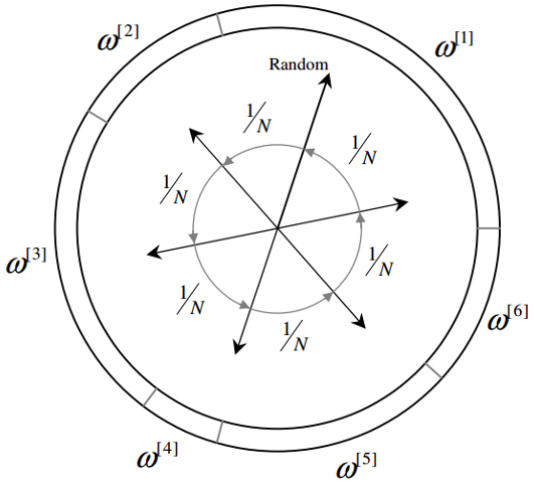
Only when this quantity falls under a threshold, resampling is carried out. The resampling scheme produces the particle set $\mathcal{P}_t$ based on the previous set $\mathcal{P}_{t-1}$ and their individual weights w_t^i. Two common resampling schemes are the multinomial, and the systematic resampling [Blanco, 2009].

For a better understanding how both algorithms work, we use an analogy to the roulette wheel which is shown in Fig. 4.5. In this scenario, each particle represents a specific position along the arc of the wheel, where the size of the circular segment corresponds to the weight of each particle. Using multinomial resampling we draw N-times independent random numbers between 0 and 360. These numbers are matched to the corresponding circular segment which results in the new particle set. Because of the independence of the drawings it can be possible that particle with higher weights gets eliminated whereas particle with lower weights survive. This is very unlikely, but it can happen.

Therefore, we prefer the systematic resampling where a random number is drawn only once. Based on this number all other particles are drawn in an equidistant way. This means that the distribution of the particles represents the target distribution more systematically. If all particles have the same weight $\hat{w}_t^i = 1/N$, the particle distribution is exactly reproduced using this technique.



(a) Multinomial resampling: drawing N-times independent random numbers between 0 and 360 leads to the new distribution of the particles.



(b) Systematic resampling: a random number is drawn only once. Based on a systematic segmentation of the circumference the new particle set is defined.

Figure 4.5: Two major resampling schemes. The multinomial resampling technique is shown in 4.5(a) and the systematic resampling in 4.5(b). The images were taken from [Blanco, 2009]. With the analogy to the roulette wheel the process of the resampling step is intuitive.

Chapter 5

Evaluation with Known Poses

In this chapter we evaluate the novel grid based mapping approach of Chapter 3 under the assumption that the pose of the vehicle is known. These experiments are divided into two parts. In the first part (Sec. 5.1) we evaluate the mapping approach based on artificial stereo image sequences. This helps us to evaluate different configurations of our mapping approach under optimum conditions. In Sec. 5.2 the performance of our approach is evaluated based on real-world data. To create reference maps, we rely on a benchmark data set which includes optimized pose information, stereo image sequences, as well as high precision laser scans. The used evaluation techniques were presented in Sec. 2.9 before.

5.1 Evaluation with an Artificial Ground Truth Data Set

In this section we evaluate the novel mapping approach with the help of artificial image sequences. The major idea is to test different parameter settings under optimum conditions. At the beginning (Sec. 5.1.1), we describe how we generate the reference occupancy grid map which is defined as ground truth (GT) from now. Sec. 5.1.2 and Sec. 5.1.3 deal with the estimation of detection rates and geometrical accuracies. Finally, we give a summary in Sec. 5.1.4.

5.1.1 Setup of Artificial Ground Truth Data and Preprocessing Steps

5.1.1.1 Generation of an Artificial Ground Truth Data Set

To generate the artificial GT data we rely on a 3D simulation environment that renders artificial stereo image sequences. Here, we use the open source library POV-Ray[1] which is a high quality software environment to create three dimensional scenes and images based on ray-tracing techniques. First, we define the trajectory of the vehicle using a realistic physics engine which is controlled by steering angle and acceleration. The driven path has a length of nearly 1 700 m. We also include a stereo camera model into the vehicle which has nearly the same stereo configuration then the stereo system in the test vehicle described in Sec. 2.2.1 and Sec. 3.2.1. The gray-scale 12 bit images have a size of $W \times H = 1024 \times 440$ pel, and the baseline is 0.23 m. The sequence is rendered with 25 Hz.

[1] http://www.povray.org/

In the second step, a 3D city model is created along the trajectory of the vehicle. The city model includes no dynamic obstacles like driving cars or pedestrians since we are interested to create optimum conditions. Figure 5.1 shows example images of the simulated city model. Because of the fact that all object points of the 3D city model are exactly known we produce a GT occupancy grid map $\mathcal{M}_{\mathrm{GT}}$ by projecting all rendered 3D points into the planar ground plane. Based on the ray-tracing and the driven path it is also possible to generate the GT free space area. Figure 5.2 shows the resulting GT grid map $\mathcal{M}_{\mathrm{GT}}$. The grid map has a dimension of 8000×8000 grid cells using a grid cell resolution of 10 cm. The library POV-Ray is also used to render artificial stereo images. The image sequence has length of 7475 frames.

Figure 5.1: Example images of the self generated 3D city model. The open source library POV-Ray is used to create the virtual environment. A realistic engine model is applied to define the GT vehicle path. Each single 3D point of the city model is known which allows us to generate the GT occupancy grid map $\mathcal{M}_{\mathrm{GT}}$. We also utilized the 3D city model to render artificial stereo image sequences which are the basis for our mapping approach (see Fig. 5.3).

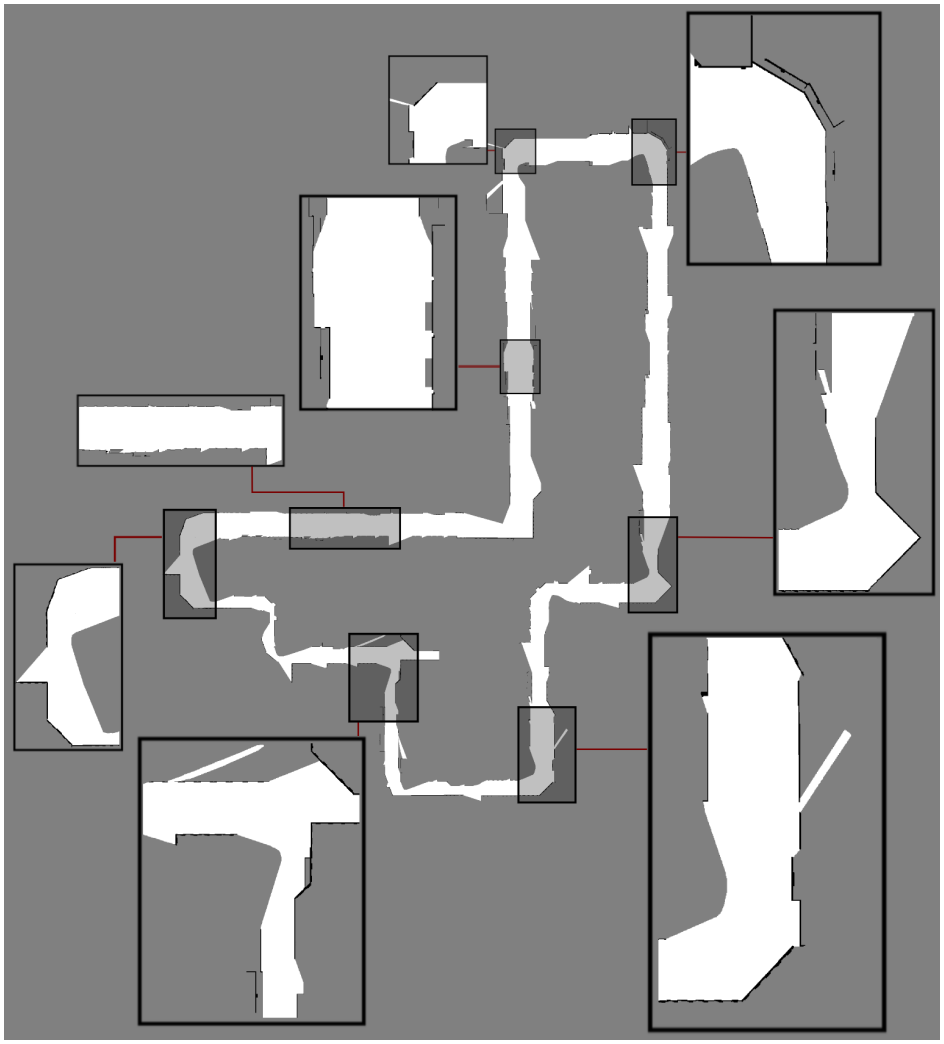


Figure 5.2: The GT occupancy grid map $\mathcal{M}_{\mathrm{GT}}$. Close-ups of the GT map are also shown to visualize details of the map and its precision. The map size is 8000×8000 grid cells which results in an area of $800 \times 800\,\mathrm{m}^2$. The driven path of the vehicle is $1,697\,\mathrm{m}$ long. The stereo image sequence has a length of 7475 frames.

5.1.1.2 Preprocessing of Input Data

Figure 5.3 shows the preprocessing steps based on the rendered stereo images. As stated in previous sections, we use SGM to estimate dense disparity images. Because of optimum disparity images we also obtain best possible Stixel World results. The quantization error only depends on the Stixel width which is also shown in Fig. 5.3. Since no dynamic obstacles are modeled, tracking and segmentation of the Stixel World is not required in this evaluation. Figure 5.3 also shows the result of the local column-disparity map as well as the local Cartesian grid map.

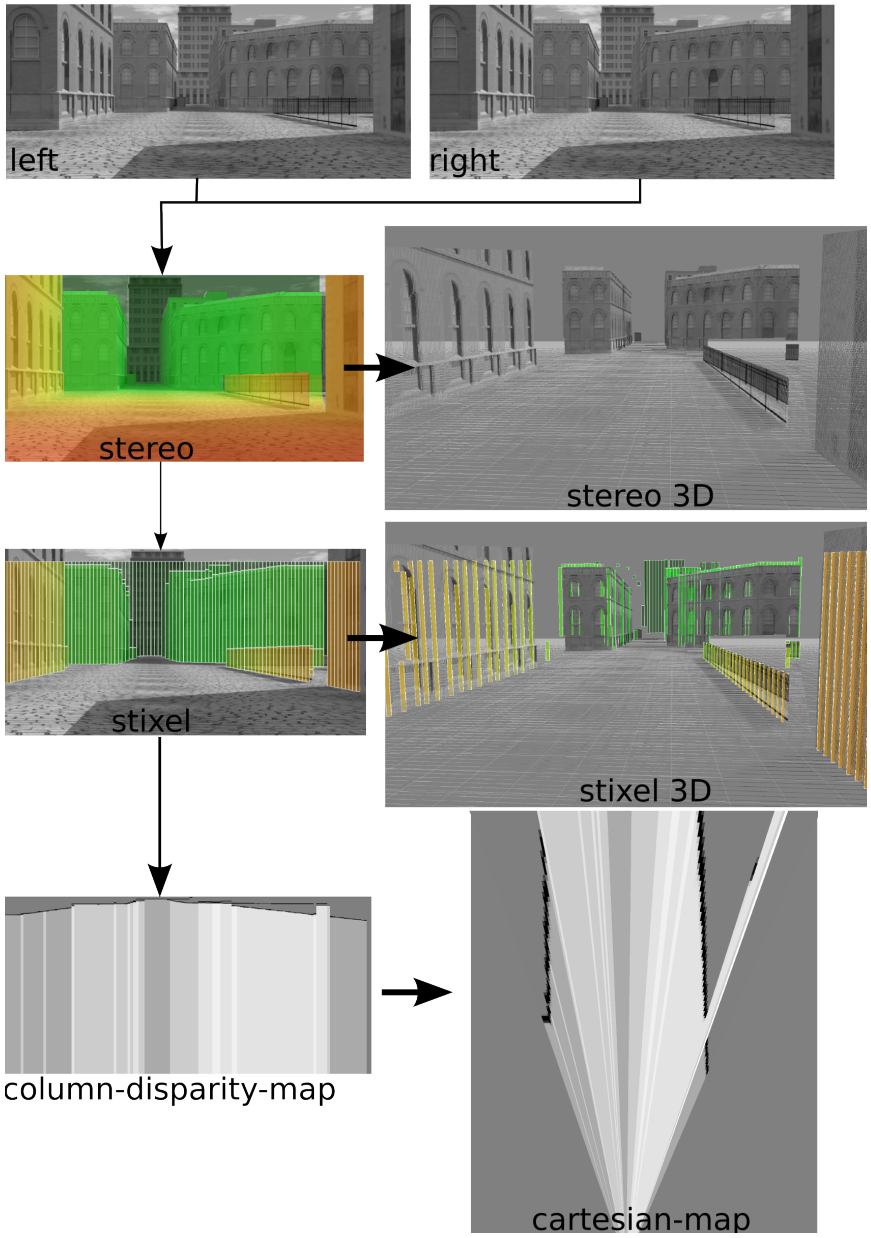


Figure 5.3: Process chain of the generation of input data with rendered sequences. Because of the 3D simulation environment, perfect stereo image sequences are rendered via ray tracing. This results in optimum disparity images and a perfect 3D point cloud for the current image pair. Consequently, the Stixel result is also perfect which is observed from the projection of the Stixels into the 3D point cloud. We only observe a quantization error which highly depends on the Stixel width. The figure also shows the column-disparity occupancy grid map as well as the local occupancy Cartesian grid map. A Stixel width of $s_w = 7\,\text{pel}$ and a disparity sampling rate of $d_s = 8$ are chosen.

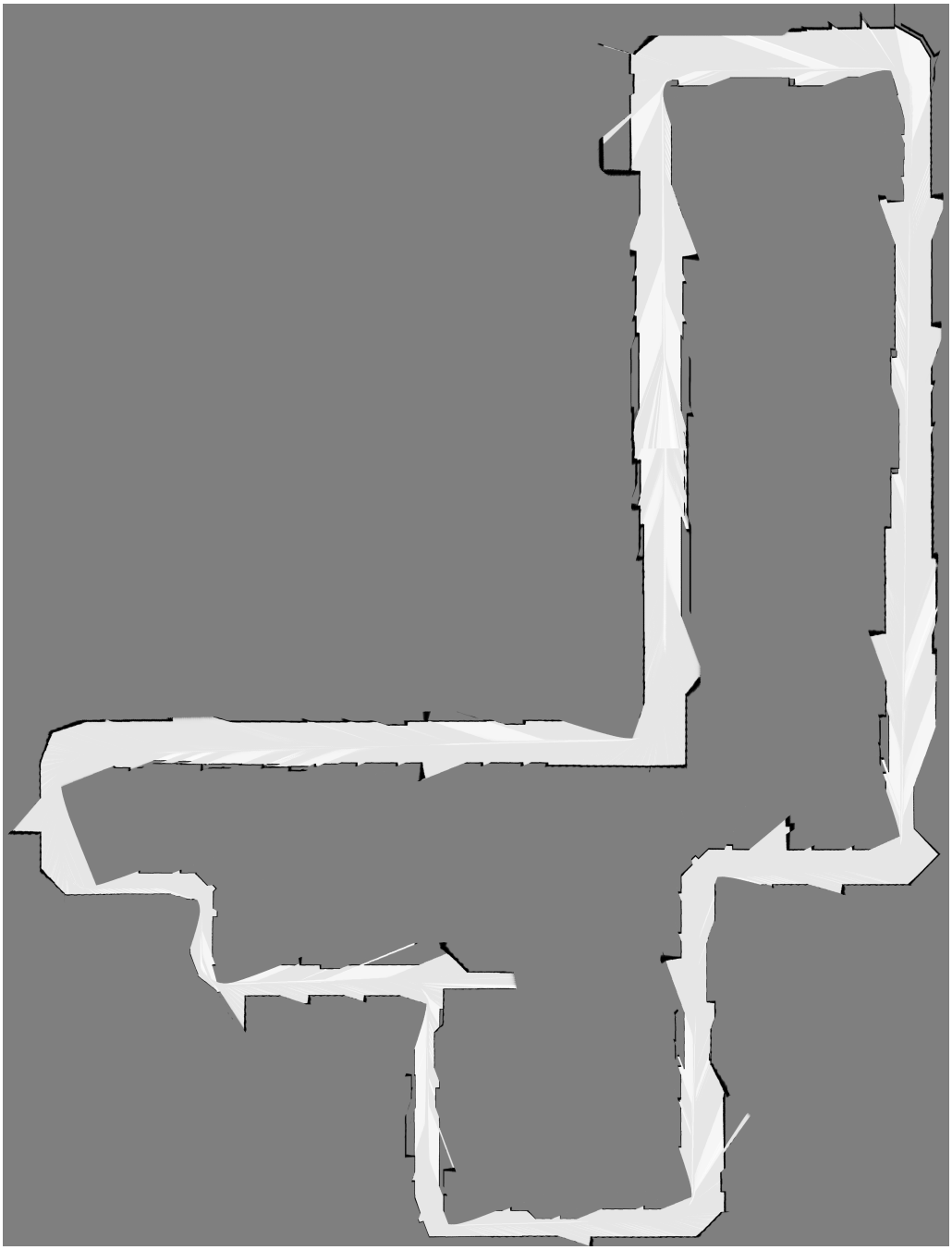


Figure 5.4: The occupancy grid map $\mathcal{M}_{\mathrm{GC}}$ based on the novel mapping approach with dynamic graph cuts for the rendered sequence. The parameter setup described in Sec. 5.1.1.3 is used. A Stixel width of $s_w = 3\,\mathrm{pel}$ and a sampling rate of $d_s = 8$ is selected. The map is based on the rendered image sequence with a length of 7475 frames. Close-ups of the map are shown in Fig. 5.5.

5.1.1.3 Parameter Settings

The novel occupancy mapping approach of Chapter 3 includes a list of parameters which have to be defined. In the following, we define important parameters for image processing, for the map creation, and for the grid map optimization.

Image Processing Settings. For the image processing we use the following parameters. The rendered images have a size of $W \times H = 1024 \times 440\,\text{pel}$. The general disparity range is defined by $d_{min} = 0\,\text{pel}$ and $d_{max} = 128\,\text{pel}$. We evaluate the influence of the Stixel width s_w in this section. Therefore, we vary this parameter and set $s_w = 1, 3, 5, 7, 9\,\text{pel}$. The number of Stixel segments for each column u is limited by the first two Stixel obstacles which means $N_u = 2$. The empirical standard deviation of the Stixels $\tilde{\sigma}^2_{d_{un}}$ and the outliers probability p^{out}_{un} are estimated during the Stixel generation process. They are not set manually.

Map Settings. The map settings are defined as follows. The local column-disparity (u^*, d^*) space is defined by $u^* \in [0, W]$ and $d^* \in [128, 0]$. The disparity sampling rate d_s of the disparity space is a key parameter which is defined manually. Because of that, we validate different intervals with $d_s = 2, 4, 8, 16$. The resolution of the a Cartesian grid cell is 0.1 m. The size of the local Cartesian grid map has a size of $40 \times 40\,\text{m}^2$.

Optimization Settings. The transition probabilities in the prediction step are defined by $p(m_{i,t} \mid m_{i,t-1}) = 0.95$ and $p(\neg m_{i,t} \mid m_{i,t-1}) = 0.05$ (see also Sec. 3.4.3). The control parameter for the binary terms are defined by $k_{ij} = 0.08$ and $\lambda_b = 2$ (see also Sec. 3.5).

Figure 5.4 shows the final grid map $\mathcal{M}_{\text{GC}}$ using the novel mapping approach based on MRFs and dynamic graph cuts. We use a setup of $s_w = 3\,\text{pel}$ and $d_s = 8$. For clarification, the index GC stands for the dynamic graph cut solution. Figure 5.5 shows six different close-ups which compare the map $\mathcal{M}_{\text{GC}}$ with the GT map $\mathcal{M}_{\text{GT}}$. Further results with different configurations are shown in Appendix B.

5.1.2 Classification Accuracy

In this section we validate how good the occupancy grid maps $\mathcal{M}_{\text{GC}}$ fits to the GT map $\mathcal{M}_{\text{GT}}$ in a quantitative way. Here, classification techniques are used which were presented in Sec. 2.9.2.

5.1.2.1 Description of the Experiment

To achieve classification accuracies we overlay the GT map $\mathcal{M}_{\text{GT}}$ with the estimated map $\mathcal{M}_{\text{GC}}$ and count grid cells which are correctly classified as obstacles or free space. For this purpose, all grid cells with a probability higher than 0.5 are classified as obstacles, and cells with a probability lower than 0.5 are classified as free space area. Unknown areas are excluded in this experiment. We achieve detection rates of obstacles and detection rates of free space in percent [%] by normalizing.

Figure 5.6 shows an example of the overlay of these grid maps for a specific scenario. The overlay is illustrated on the right of this figure. Here, the red areas describe an “over-segmentation”. This

means in this case that we falsely estimate more occupied grid cells than exist in the GT map. Blue areas describe "under-segmentation" which means in this case that we missed obstacles during map estimation. In this example, the Stixel width is set by $s_w = 1$ pel and the disparity sampling rate is set by $d_s = 16$. The detection rate of obstacles is 89.5 %, and the detection rate of free space is 97.16 %. In the following, we vary the Stixel width s_w and the disparity sampling rate d_s to get insights which configuration performs best. We are also interested how good the novel mapping approach performs against occupancy grid mapping approaches which neglect the dependencies of neighboring grid cells. Therefore, we also estimate occupancy grid maps following the approach of Muffert et al. [2014]. This approach is also based on existence estimation (see Sec. 2.4.2 and Sec. 3.4.1), but does not model the dependencies between neighboring grid cells. These maps are called $\mathcal{M}_{\mathrm{EX}}$.

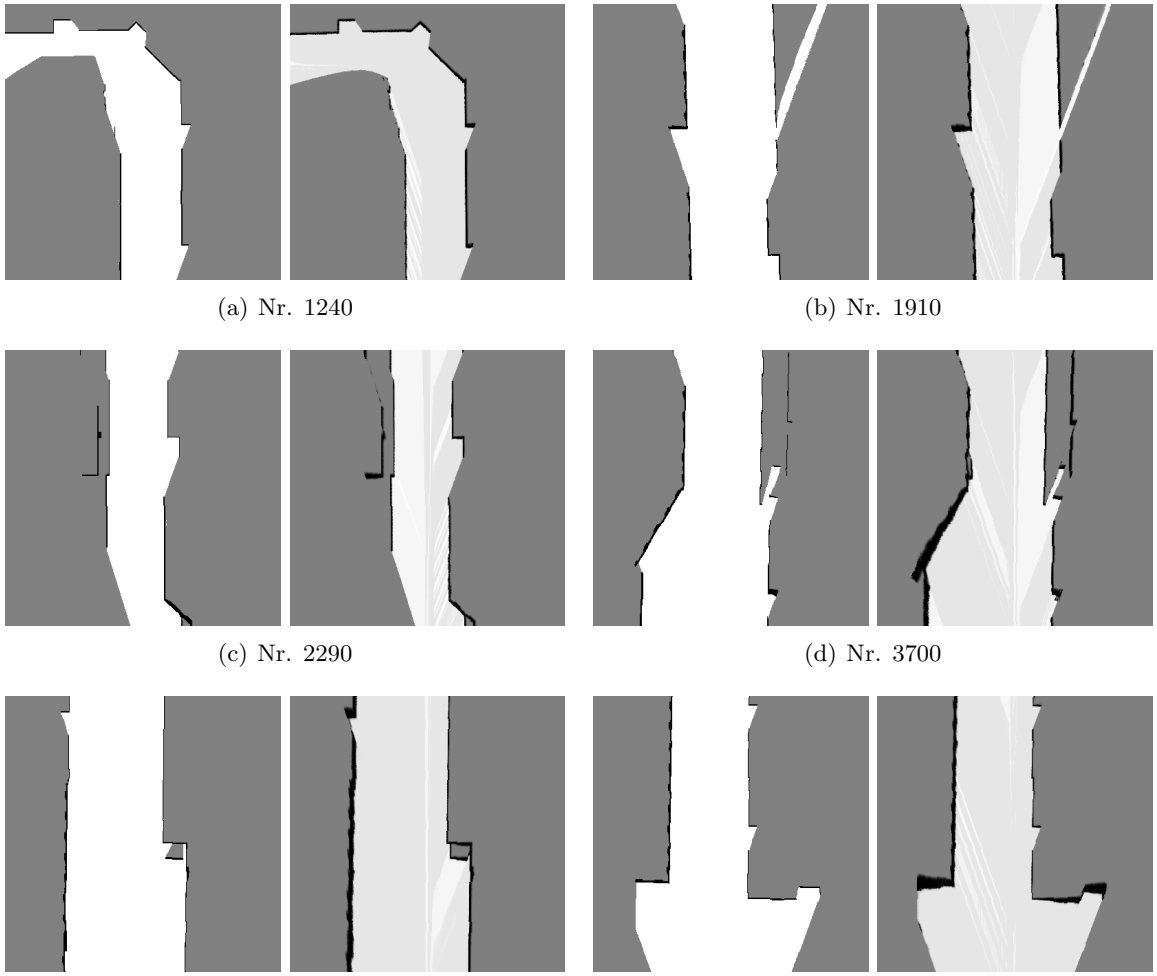
5.1.2.2 Results of Detection Rates

Tab. 5.1 shows classification accuracies for different parameter setups. The table includes five major blocks where each block is for a specific Stixel width s_w. Each block itself includes classification results for different disparity sampling rates d_s. In total, 20 different setups are evaluated. For each setup the table includes the detection rates of obstacles and detection rates of free space of the mapping approach $\mathcal{M}_{\mathrm{GC}}$. The table also includes the results of the mapping approach $\mathcal{M}_{\mathrm{EX}}$. For a better interpretation of the data, the classification results are also shown as a diagram in Fig. 5.7. In this figure, the detection rate of free space is plotted against the detection rate of obstacles. The best performance is in the upper right corner, where both detection rates become 100.0 %. The different Stixel widths are represented by different colors. Here, red for $s_w = 1$, dark red for $s_w = 3$, green for $s_w = 5$, dark green for $s_w = 7$, and blue for $s_w = 9$. The different sampling rates are visualized with different symbols. We use rectangles for $d_s = 2$, circles for $d_s = 4$, stars for $d_s = 8$, and triangles for $d_s = 16$. The results for the novel mapping approach $\mathcal{M}_{\mathrm{GC}}$ are represented with dashed lines, and the results for method $\mathcal{M}_{\mathrm{EX}}$ with solid lines. In addition, Appendix B.1 shows the overlay of the complete occupancy grid map with $\mathcal{M}_{\mathrm{GT}}$.

5.1.2.3 Discussion

Based on the results of Tab. 5.1, we observe that the novel mapping approach achieves the best detection rate of obstacles of 99.11 % with a Stixel width of $s_w = 1$ and a disparity sampling rate of $d_s = 2$. The lowest detection rate of obstacles is 98.01 % with a setup of $s_w = 9$ and $d_s = 16$. In comparison, the approach $\mathcal{M}_{\mathrm{EX}}$ performs best by a setup of $s_w = 3$ and $d_s = 2$ with a detection rate of 98.70 %. The approach shows its lowest performs using a setup of $s_w = 1$ and $d_s = 16$.

With regard to free space, $\mathcal{M}_{\mathrm{GC}}$ has the best performance with 97.16 % and the approach $\mathcal{M}_{\mathrm{EX}}$ with 97.20 %. In both cases, the setup is defined by $s_w = 1$ and $d_s = 16$. The lowest performance of free space detection is 93.06 % for $\mathcal{M}_{\mathrm{GC}}$ and 93.08 % for $\mathcal{M}_{\mathrm{EX}}$ using the setup $s_w = 9$ and $d_s = 2$. The novel mapping method $\mathcal{M}_{\mathrm{GC}}$ outperforms the method $\mathcal{M}_{\mathrm{EX}}$ with regard to the detection rate of obstacles. Our assumption for this reason is, that the influence of the binary terms lead to a wider representation of obstacles. Based on the smoothness effect of the binary terms, more cells become occupied. The method $\mathcal{M}_{\mathrm{EX}}$ performs slightly better in the detection of free space using artificial data.



(a) Nr. 1240 (b) Nr. 1910

(c) Nr. 2290 (d) Nr. 3700

(e) Nr. 5140 (f) Nr. 5650

Figure 5.5: Comparison between close-ups of the GT map $\mathcal{M}_{\text{GT}}$ and the estimated map $\mathcal{M}_{\text{GC}}$. To generate $\mathcal{M}_{\text{GC}}$, a Stixel width of $s_w = 3\,\text{pel}$ and a disparity sampling rate $d_s = 8s$ is used. Obstacles as well as free space are estimated correctly. The discretization of the column-disparity space and the transformation in the Cartesian grid map afterwards leads to the fact, that in some cases obstacles are more spread out than GT provides. This is clearly shown in 5.5(c), 5.5(d) and 5.5(f).

In Fig. 5.4 and Fig. 5.5 we also observe that the mapping results are close to a binary solution. This statement is especially valid for obstacles. The reason for this is that the precision of the Stixels is very high because GT disparity maps without any noise are used (see Fig. 5.3). Consequently, the Stixel depth has a very high precision $\tilde{\sigma}^2_{d_{un}}$. Since the disparity images are nearly perfect, we also observe low outlier probabilities p^{out}_{un}. Both settings produce a very sharp Gaussian function for the measurement model with regard to the disparity space.

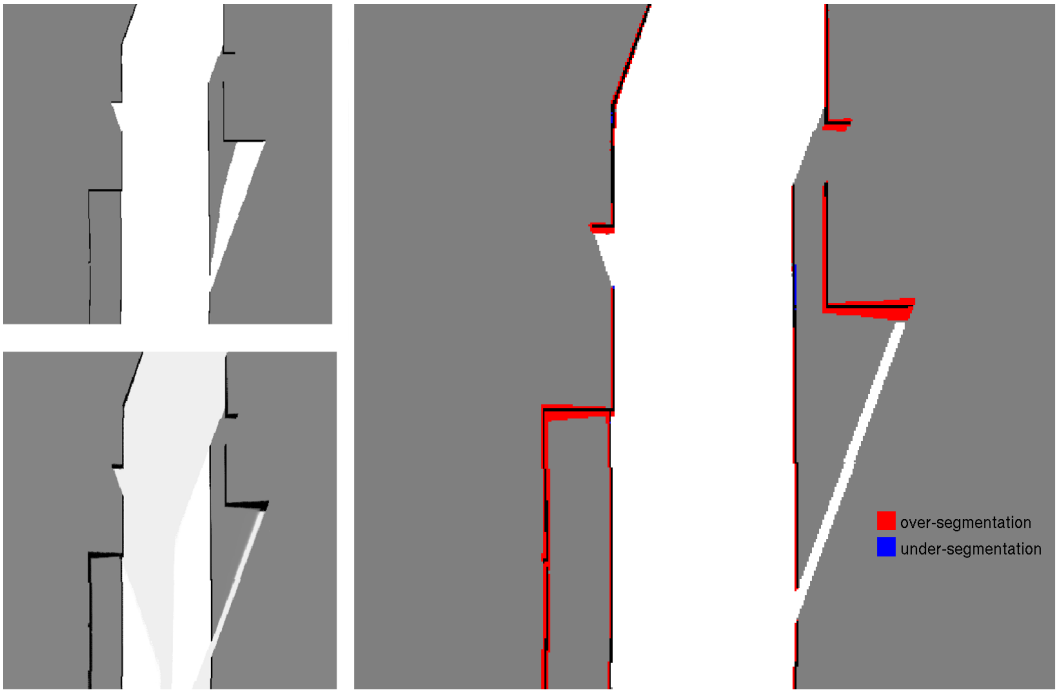


Figure 5.6: The overlay of the GT map close-up with the estimated map close-up for frame number 3354. The upper left image represents the GT map $\mathcal{M}_{\mathrm{GT}}$. The lower left image shows the result of the estimated map $\mathcal{M}_{\mathrm{GC}}$. The settings are $s_w = 1\,\mathrm{pel}$ and $d_s = 16$. On the right side the overlay of both maps is shown. Red areas describe an "over-segmentation", and blue areas describe an "under-segmentation". For this setup, the detection rate for obstacles is 89.5 % and for free space 97.16 %. An "under-segmentation" is almost not observed in this scenario.

Nevertheless, the transformation into the Cartesian space produces a smearing effect (see Sec. 3.4.2). This effect strongly depends on the distance to the vehicle which is simulated in Fig. 5.8. The figure shows a high peaked, continuous signal in the disparity space, its discrete projection in the Cartesian space, and the transformation into an equidistant Cartesian 1D grid space. It illustrates that obstacles in smaller distances are perfectly transformed into the regular Cartesian grid space. Obstacles which are far away produce a smearing effect because of the chosen interpolation techniques described in Sec. 3.4.2.

Figure 5.7 also allows additional insights with regard to the overall performance of both methods. Taking both detection rates into account, the novel approach performs better than $\mathcal{M}_{\mathrm{EX}}$, especially for the disparity sampling rate $d_s = 8$ in combination with the Stixel widths $s_w = 1, 3, 5$. Therefore, we suggest to use one of these setups for the following evaluation steps. We also recognize that the overall performance behavior of the detection rates is mainly influenced by the disparity sampling rate d_s. For both mapping methods and for all Stixel widths, the following statements are true. The highest detection rates of free space are achieved with $d_s = 16$, but we also get the lowest detection rates of obstacles with this disparity sampling rate.

Table 5.1: Classification accuracies for different parameter setups. The map $\mathcal{M}_{\text{GT}}$ (see Fig. 5.2) is used as GT data. We vary the Stixel width s_w and the disparity sampling rate d_s. In total, 20 different parameter setups are tested. Next to the validation of the novel approach $\mathcal{M}_{\text{GC}}$, we also validate the approach of Muffert et al. [2014] which results are defined as $\mathcal{M}_{\text{EX}}$. We highlight the lowest detection rates in red and the highest detection rates in blue. In total, 92.21 % of the grid cells of the GT grid map are free, 7, 79 % are occupied cells. Grid cells with an unknown state are not considered in this ratio.

	detection rate of obstacles [%]		*detection rate of free space [%]*	
	$\mathcal{M}_{\text{EX}}$	$\mathcal{M}_{\text{GC}}$	$\mathcal{M}_{\text{EX}}$	$\mathcal{M}_{\text{GC}}$
stixel width $s_w = 1$				
$d_s = 16$	97.38	98.50	97.20	97.16
$d_s = 08$	98.25	98.82	96.68	96.63
$d_s = 04$	97.68	98.60	96.18	96.15
$d_s = 02$	98.68	99.11	93.49	93.48
stixel width $s_w = 3$				
$d_s = 16$	98.15	98.67	96.96	96.92
$d_s = 08$	98.51	98.88	96.48	96.42
$d_s = 04$	98.45	98.91	95.97	95.92
$d_s = 02$	98.70	99.03	93.36	93.35
stixel width $s_w = 5$				
$d_s = 16$	98.02	98.48	96.79	96.74
$d_s = 08$	98.52	98.79	96.32	96.26
$d_s = 04$	98.65	98.89	95.79	95.72
$d_s = 02$	98.67	98.92	93.28	93.27
stixel width $s_w = 7$				
$d_s = 16$	97.78	98.24	96.52	96.47
$d_s = 08$	98.23	98.49	96.10	96.03
$d_s = 04$	98.54	98.71	95.40	95.30
$d_s = 02$	98.51	98.72	93.15	93.14
stixel width $s_w = 9$				
$d_s = 16$	97.51	98.01	96.30	96.25
$d_s = 08$	98.05	98.34	95.91	95.83
$d_s = 04$	98.36	98.61	95.21	95.08
$d_s = 02$	98.45	98.64	93.08	93.06

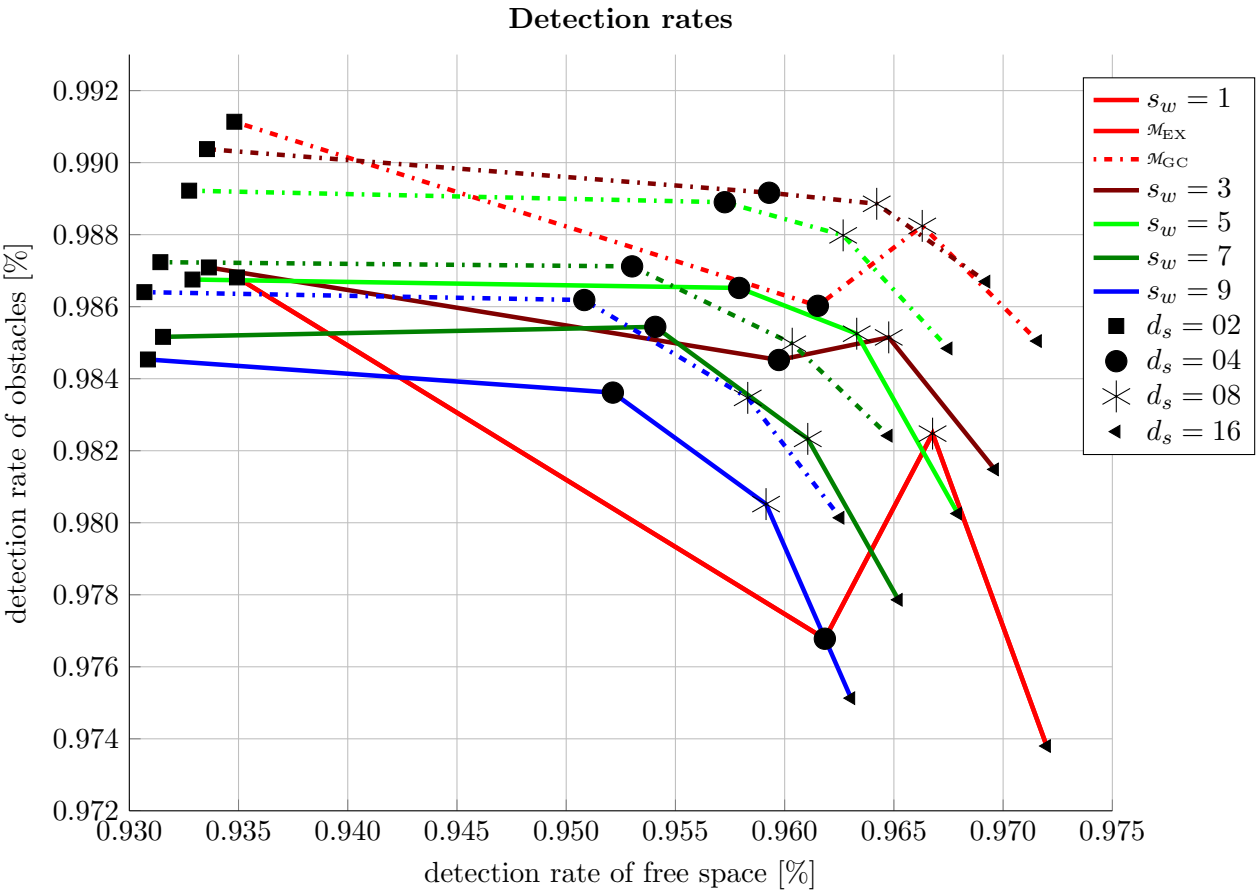


Figure 5.7: Detection rates of free space and detection rates of obstacles for 20 different configurations. The different colors present different Stixel widths. The symbols represent different disparity sampling rates . The solid lines are used for method $\mathcal{M}_{\mathrm{EX}}$, the dashed lines are used for method $\mathcal{M}_{\mathrm{GC}}$. It is clearly shown, that the approach based on MRFs outperforms the method $\mathcal{M}_{\mathrm{EX}}$.

On the other hand, the highest detection rates of obstacles are obtained by $d_s = 2$, whereas this configuration shows the lowest performance for free space detection. This contradictory behavior is explained as follows: The lower the disparity space is sampled, the higher is the quantization error. A high quantization error leads to a strong over-segmentation of obstacles and, consequently to good obstacle detection rates but a poor free space quality. On the other hand, a finely sampled disparity space leads to under-segmentation of obstacles. This results in the fact, that we miss static obstacles, but also increase the quality of free space. Both methods tend to an over-segmentation. This means that the resulting maps have more occupied areas than really exist. This topic is discussed in Sec. 5.1.4 again. Based on the insights of this section, we propose to use a Stixel width of $s_w = 3$ in combination with a disparity sampling rate of $d_s = 16$ or $d_s = 8$.

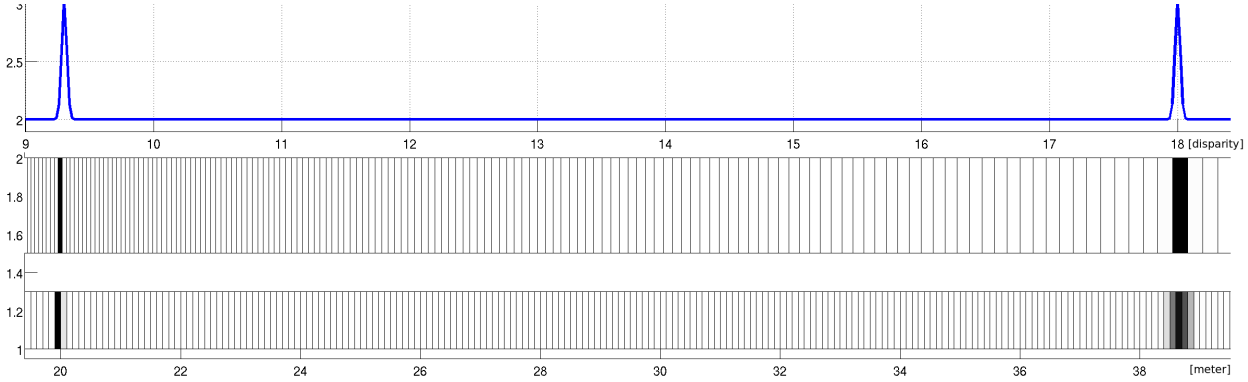


Figure 5.8: A high peaked measurement model in the disparity space (top), its discrete projection in the Cartesian space (middle), and its transformation into an equidistant Cartesian 1D grid space (bottom). The interpolation method described in Sec. 3.4.2 was chosen.

5.1.3 Geometrical Accuracy

In this section we evaluate the geometrical accuracies of the novel occupancy grid maps based on GT data. The goal is to estimate how accurate static obstacles are represented in the map with regard to the vehicle's point of view. Therefore, we describe the idea of the derivation of geometrical entities from grid maps first, followed by the description of correspondences and their weight estimation. Based on these results, we are able to estimate (weighted) absolute mean errors (W)MAEs which were already defined in Sec. 2.9.1. These measures allow us to make quantitative statements with regard to the geometrical accuracy of the generated maps.

5.1.3.1 Description of the Experiment

To achieve geometrical accuracy assessments, the following situation is regarded. The generated map should be used for localization or path planning purposes. For this task the best geometrical accuracy of static obstacles is needed. In an optimum case, the geometry of static obstacles, like walls and buildings, fits best to the GT map. In our experiment we assume that we drive again along the given pose and scan the environment with a laser scanner pointed in driving direction. Here the "trick" is, that the environment is represented by the created maps $\mathcal{M}_{\mathrm{GC}}$, $\mathcal{M}_{\mathrm{EX}}$, and $\mathcal{M}_{\mathrm{GT}}$. The laser scanner reports us distances to static obstacles for each emitted ray. This results in distance measurements with regard to the vehicle coordinate system.

For a better understanding, this experiment is illustrated in Fig. 5.9. It shows the ray casting, as well as the derived 2D hit points of obstacles for a GT map sample and for an estimated map sample. The scan procedure is applied for all map types $\mathcal{M}_{\mathrm{GC}}$, $\mathcal{M}_{\mathrm{EX}}$ and $\mathcal{M}_{\mathrm{GT}}$. The scan step involves, that more static obstacles close by the vehicle are hit than obstacles which are far way (see also Figs. 5.9(a)-5.9(d)). This results in the fact, that obstacles close to the vehicle are more considered during evaluation than obstacles which are far away. This fact is reasonable since it is necessary that especially the environment close to the vehicle is mapped correctly, e.g. during path planning. To avoid correlations between the scans, the scanning is carried out every 10 th frame.

5.1.3.2 Definition of Geometrical Map Errors

To estimate geometrical map errors we have to align the detected points from the estimated map and the GT map. To solve this correspondence problem, the nearest neighbor search is used for each scan run. The maximum search radius is 1.0 m. Points outside this search window are defined as outliers and are neglected in the following evaluation steps. We only allow $1:1$ correspondences which means that only one GT point is associated with only one measurement point. An example is shown in Figs. 5.9(c)-5.9(f). Following the definition in Sec. 2.9.1 we define the geometrical accuracy of an estimated map by the distance errors of static obstacles. Therefore, we estimate the distances for all three map types d^i_{GC}, d^i_{EX}, and d^i_{GT} based on the detected points. We define the geometrical error Δg^i_m by

$$\Delta g^i_m \coloneqq d^i_{\mathrm{GT}} \; - \; d^i_m \text{ with } m = \{\mathrm{GC}, \mathrm{EX}\}. \tag{5.1}$$

5.1.3.3 Weight Estimation

Following the definition in Sec. 2.9.1, we want to exploit the full information of GT data and estimated data which means to also take the precision of the geometrical errors into account. Unfortunately, occupancy grid maps do not provide accuracy information of static obstacles by default. They are made to represent the occupancy probabilities.

However, in order to estimate precision or weight values for the estimated geometrical errors, we make use of the ray casting technique again. We collect all probability values along the ray until a static obstacle is hit. A sigmoid function is fit into these probability values with regard to the traveled (ray) distance. This step is visualized in Fig. 5.10. In general, the sigmoid function $f_{sig}(x, \lambda, \tau_{\min}, \tau_{\max})$ is controlled by the min/max range values $\tau_{\min}$ and $\tau_{\max}$, the slope factor λ and the turning point x. The sigmoid function is defined by

$$f_{sig}(x, \lambda, \tau_{\min}, \tau_{\max}) = \frac{(\tau_{\max} - \tau_{\min})}{(1 + e^{(-x/\lambda)})} + \tau_{\min}. \tag{5.2}$$

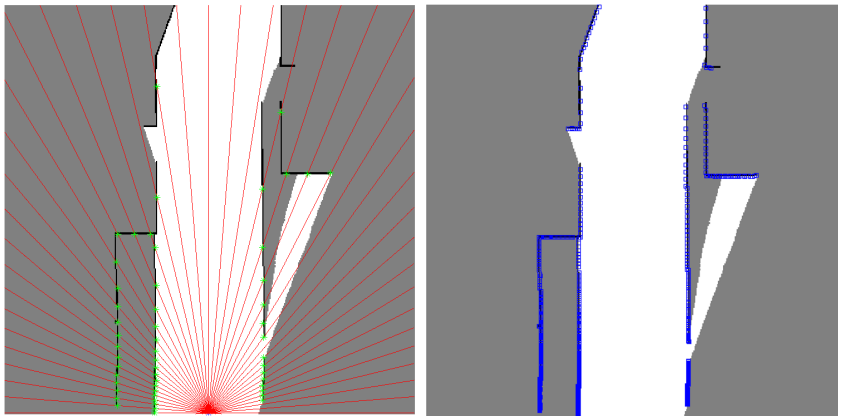
In our case the turning point x is defined by the probability occupancy value of a hit point and its corresponding traveled distance along a ray. The range interval is defined in the surrounding of the turning point with $\tau_{\min} = x - 0.3\,\mathrm{m}$ and $\tau_{\max} = x + 0.3\,\mathrm{m}$. For our purpose, the important factor is the slope value λ. If the value is close to zero, the sigmoid function is nearly a binary step function. As one can seen in Fig. 5.10, this is the case for the points taken from $\mathcal{M}_{\mathrm{GT}}$.

This is consistent with the idea of GT data, since $\mathcal{M}_{\mathrm{GT}}$ is binary. In contrast, the estimated maps $\mathcal{M}_{\mathrm{GC}}$ and $\mathcal{M}_{\mathrm{EX}}$ show a different behavior. Here, the slope factor differs considerably from the slope values of $\mathcal{M}_{\mathrm{GT}}$. This is reasonable because of the uncertain behavior of the grid maps.

We take these insights into account for the estimation of precision values and weights, respectively. For each distance d^i_{m} the corresponding slope value $\lambda_{d^i_{\mathrm{m}}}$ is estimated. The decision is made, that the slope represents the precision of the distance d^i_{m}. Based on this statement the weights for the geometrical errors are defined by

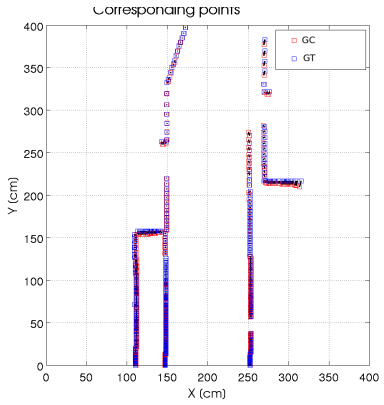
$$w^i_{\Delta g_m} = \frac{1}{\lambda^2_{d^i_m} + \lambda^2_{d^i_{\mathrm{GT}}}} \; . \tag{5.3}$$

The distribution of estimated slopes $\lambda_{d^i_{\mathrm{GC}}}$ of configuration $s_w = 3$ and $d_s = 08$ is shown in Fig. 5.11. The figure also includes the slopes values $\lambda_{d^i_{\mathrm{GT}}}$ of the GT map $\mathcal{M}_{\mathrm{GT}}$. As assumed, the slopes $\lambda_{d^i_{\mathrm{GT}}}$ are considerably smaller than the slopes $\lambda_{d^i_{\mathrm{GC}}}$. We also observe outliers at positions 0.05, 0.09, and 0.18. These outliers can be explained by errors during the sigmoid fitting process. Compared to the GT data, the slopes for $\mathcal{M}_{\mathrm{GC}}$ $\lambda_{d^i_{\mathrm{GC}}}$ are more spread out. The average value for the slope is about 0.05. For visualization purposes, we clip the histograms at 0.4 at the x-axis and clip at 55 000 at the y-axis. About 300 000 slope values are estimated for each method.

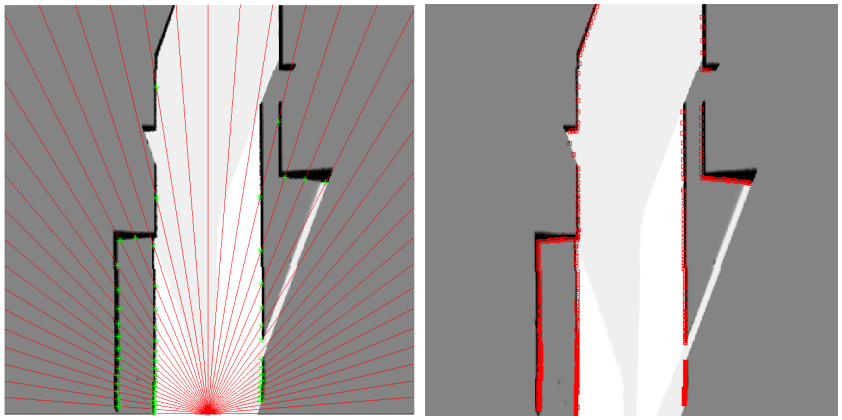


(a) Ray casting of $\mathcal{M}_{\mathrm{GT}}$ for a specific frame i

(b) Scanned points of $\mathcal{M}_{\mathrm{GT}}$

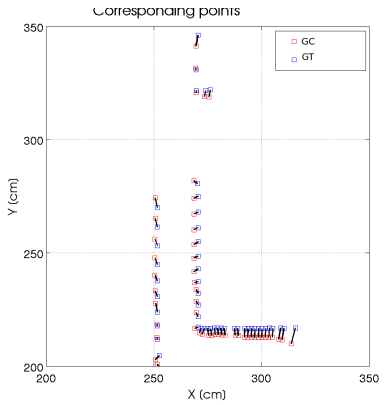


(c) Correspondences and geometrical errors Δg^i_m



(d) Ray casting of $\mathcal{M}_{\mathrm{GC}}$ for a specific frame i

(e) Scanned obstacle points of $\mathcal{M}_{\mathrm{GC}}$



(f) Correspondences and geometrical errors Δg^i_m (zoom)

Figure 5.9: Derivation of geometrical errors for occupancy grid maps. The estimated map, as well as the GT map are scanned with a simulated laser scanner to derive geometrical errors Δg^i_m (Figs. 5.9(a)-5.9(d)). This results in point clouds and correspondences (Figs. 5.9(b)-5.9(e)), and in the distance errors Δg^i_m (Figs. 5.9(c)-5.9(f)).

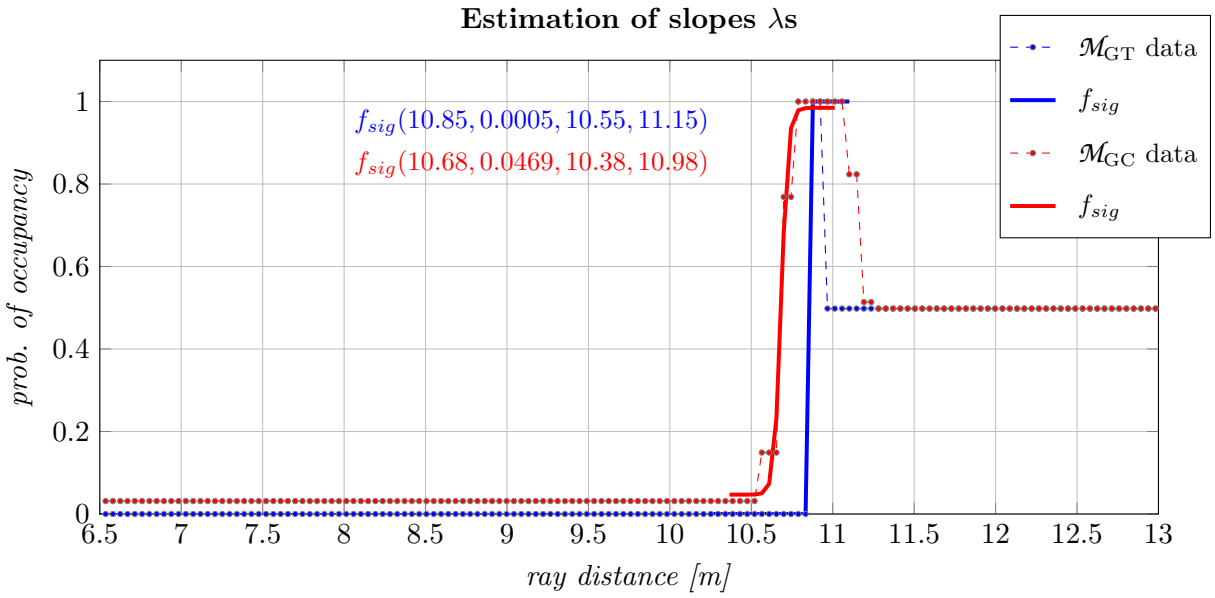


Figure 5.10: Concept of the precision estimation in occupancy grid maps. A sigmoid function is fit into the probability values along the ray distance. For GT, the probability behavior follows a very sharp 1D sigmoid function with the slope $\lambda_{d^i_{\mathrm{GT}}}$. For the estimated map $\mathcal{M}_{\mathrm{GC}}$ the sigmoid is wider which results in a larger slope value. The slope values are used to estimate precision values for the hit points.

5.1.3.4 Results of Geometrical Map Errors

Following Sec. 2.9.1, we decide to use the mean absolute error (MAE) and the weighted mean absolute error (WMAE) as measures for geometrical accuracy assessments. Using (5.1) and (5.3) the $\mathrm{MAE_m}$ is defined by

$$\mathrm{MAE_m} = \frac{1}{N} \sum_{i=1}^{I} \mid \Delta g_m^i \mid \quad , \tag{5.4}$$

and the $\mathrm{WMAE_m}$ is computed by

$$\mathrm{WMAE_m} = \frac{1}{\sum_{i=1}^{I} w^i_{\Delta g_m}} \sum_{i=1}^{I} w^i_{\Delta g_m} \mid \Delta g_m^i \mid \quad . \tag{5.5}$$

As in Sec. 5.1.2, the 20 configurations of different Stixel widths and disparity sampling rates are tested. The results for the $\mathrm{MAE_m}$ and $\mathrm{WMAE_m}$ are shown in Tab. 5.2. For each configuration, we use about 300 000 distances for the estimation of the measures. Similar to the previous section, we also visualize the different $\mathrm{WMAE_m}$ in a diagram It allows a better interpretation of the data and is illustrated in Fig. 5.13. The distributions for the slope values $\lambda_{d^i_{\mathrm{GC}}}$ and $\lambda_{d^i_{\mathrm{EX}}}$ are shown in Fig. 5.12. As in Fig. 5.11, we clip the histograms at 0.4 at the x-axis and clip at 55 000 at the y-axis.

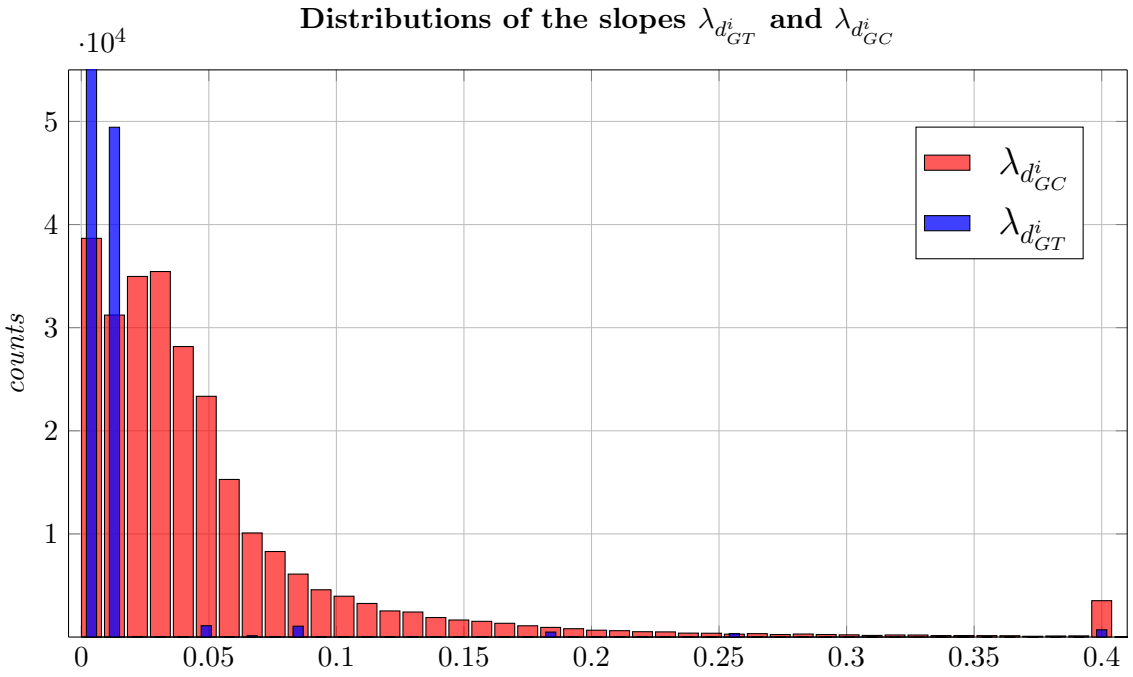


Figure 5.11: Distributions of the slope values for $\lambda_{d^i_{\mathrm{GC}}}$ of configuration $s_w = 3$ and $d_s = 08$ (red), and for the GT slope values $\lambda_{d^i_{\mathrm{GT}}}$ (blue). The slopes $\lambda_{d^i_{\mathrm{GC}}}$ are more spread out. The average is about 0.051. The values $\lambda_{d^i_{\mathrm{GT}}}$ are close to zero. Some outliers occur which is caused by fitting issues.

5.1.3.5 Discussion

Taken the MAEs in Tab. 5.2 into account, method $\mathcal{M}_{\mathrm{GC}}$ performs best using the Stixel width $s_w = 3$ and the disparity sampling rate $d_s = 16$. A $\mathrm{MAE}_{\mathrm{GC}}$ of 0.083 m is achieved. Method $\mathcal{M}_{\mathrm{EX}}$ performs best using the Stixel width $s_w = 1$ and the disparity sampling rate $d_s = 16$. Here, the $\mathrm{MAE}_{\mathrm{EX}}$ is 0.077 m. Method $\mathcal{M}_{\mathrm{GC}}$, as well as method $\mathcal{M}_{\mathrm{EX}}$ show their lowest performance using a Stixel width of $s_w = 5$ and a disparity sampling rate of $d_s = 02$. Here, the MAEs are 0.173 m (GC) and 0.171 m (EX) respectively. As seen in Tab. 5.2 and Fig. 5.13, the geometrical accuracy increases the greater the disparity sampling rates are. This is independent from the chosen mapping method and Stixel width. It is also coherent with the results of Sec. 5.1.2. If we chose high disparity sampling rates, the influence of over-segmentation is low and, consequently obstacles fit better to GT. This results in lower geometrical errors.

Without doubt, a benefit of the new method $\mathcal{M}_{\mathrm{GC}}$ compared to $\mathcal{M}_{\mathrm{EX}}$ is not observed as long as the MAE is taken into account. This is also visualized in the four histograms of Fig. 5.14 where the error distributions for the absolute geometrical errors $\mid \Delta g^i_m \mid$ are shown. Here, the distributions of the absolute errors are nearly the same for both methods. As one can see, the error distributions are spreading out with decreasing the disparity sampling rates which results in larger $\mathrm{MAE}_{\mathrm{m}}$. Using the lowest disparity interval, about 3% of the geometrical errors are larger than 0.6 m (see Fig. 5.14, bottom left). For all other configurations, the rate is lower than 1%.

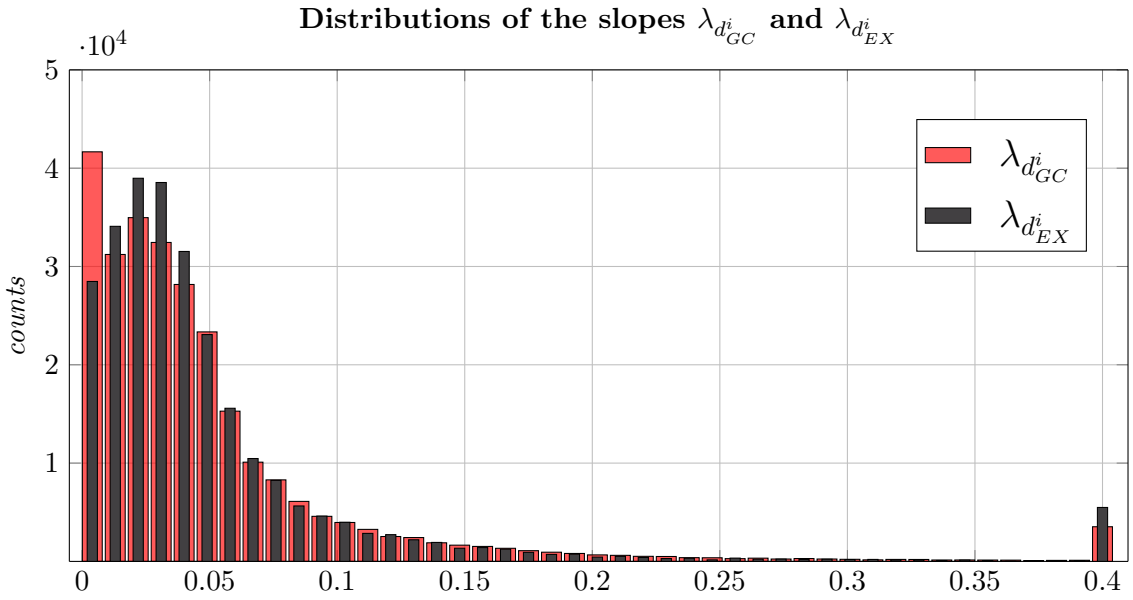


Figure 5.12: Distributions of the slope values for $\lambda_{d^i_{\mathrm{GC}}}$ and $\lambda_{d^i_{\mathrm{EX}}}$ of configuration $s_w = 3$ and $d_s = 08$. The average slope value for method $\mathcal{M}_{\mathrm{GC}}$ is 0.0509 and for method $\mathcal{M}_{\mathrm{EX}}$ is 0.0522.

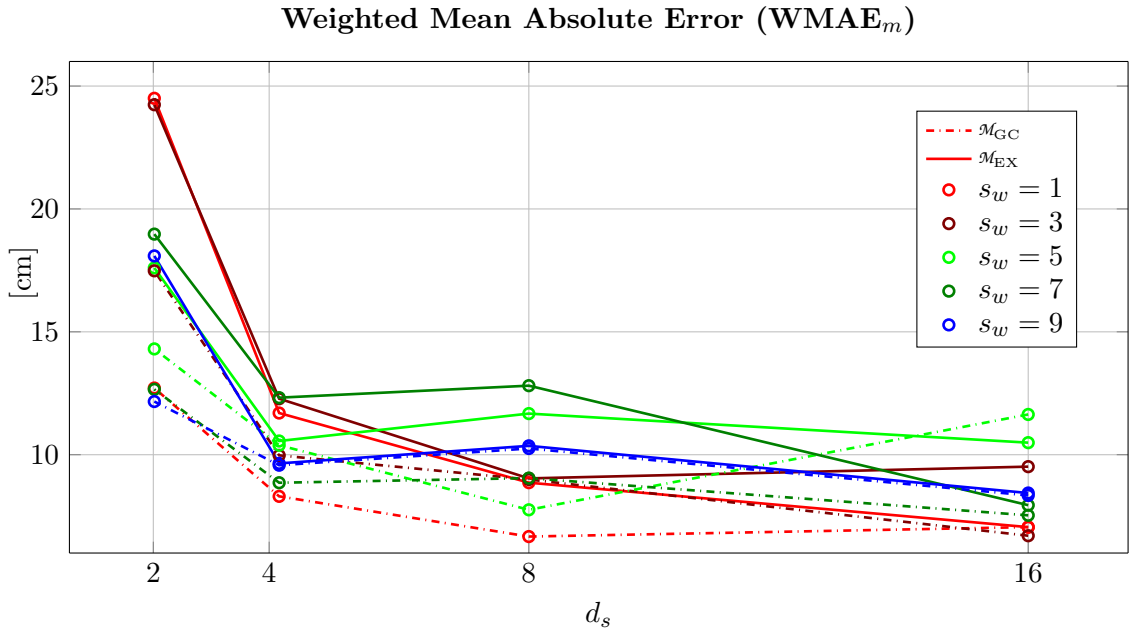


Figure 5.13: Weighted mean absolute errors ($\mathrm{WMAE_m}$) for all 20 configurations. The x-axis represents the disparity sampling rates. The y-axis the $\mathrm{WMAE_m}$. The different colors present different Stixel widths. The solid lines represent method $\mathcal{M}_{\mathrm{EX}}$, and the dashed lines method $\mathcal{M}_{\mathrm{GC}}$.

Table 5.2: Mean absolute errors (MAE_ms) and weighted mean absolute errors (WMAE_ms) of method $\mathcal{M}_{\text{GC}}$ and $\mathcal{M}_{\text{EX}}$ for 20 different configurations. The table has the same structure than Tab. 5.1. The estimation is based on geometrical errors Δg_m^i where m represents the used mapping method $\mathcal{M}_{\text{GC}}$ and $\mathcal{M}_{\text{EX}}$, respectively. For the weight estimation, we use the method in Sec. 5.1.3.3. We highlight the highest errors in red and the lowest errors in blue. For each setup, about 300 000 measurement values are taken into account.

	MAE_m [in meter]		WMAE_m [in meter]	
	$m = \text{GC}$	$m = \text{EX}$	$m = \text{GC}$	$m = \text{EX}$
stixel width $s_w = 1$				
$d_s = 16$	0.089	0.077	0.070	0.070
$d_s = 08$	0.097	0.095	0.066	0.088
$d_s = 04$	0.108	0.106	0.083	0.116
$d_s = 02$	0.170	0.168	0.127	0.245
stixel width $s_w = 3$				
$d_s = 16$	0.083	0.081	0.067	0.095
$d_s = 08$	0.100	0.098	0.089	0.090
$d_s = 04$	0.111	0.109	0.099	0.122
$d_s = 02$	0.172	0.170	0.174	0.242
stixel width $s_w = 5$				
$d_s = 16$	0.086	0.083	0.116	0.104
$d_s = 08$	0.102	0.100	0.077	0.116
$d_s = 04$	0.113	0.111	0.103	0.105
$d_s = 02$	0.173	0.171	0.143	0.176
stixel width $s_w = 7$				
$d_s = 16$	0.090	0.087	0.075	0.079
$d_s = 08$	0.105	0.103	0.090	0.128
$d_s = 04$	0.120	0.116	0.088	0.123
$d_s = 02$	0.171	0.169	0.126	0.189
stixel width $s_w = 9$				
$d_s = 16$	0.093	0.090	0.083	0.084
$d_s = 08$	0.109	0.105	0.102	0.103
$d_s = 04$	0.124	0.119	0.095	0.096
$d_s = 02$	0.170	0.168	0.121	0.180

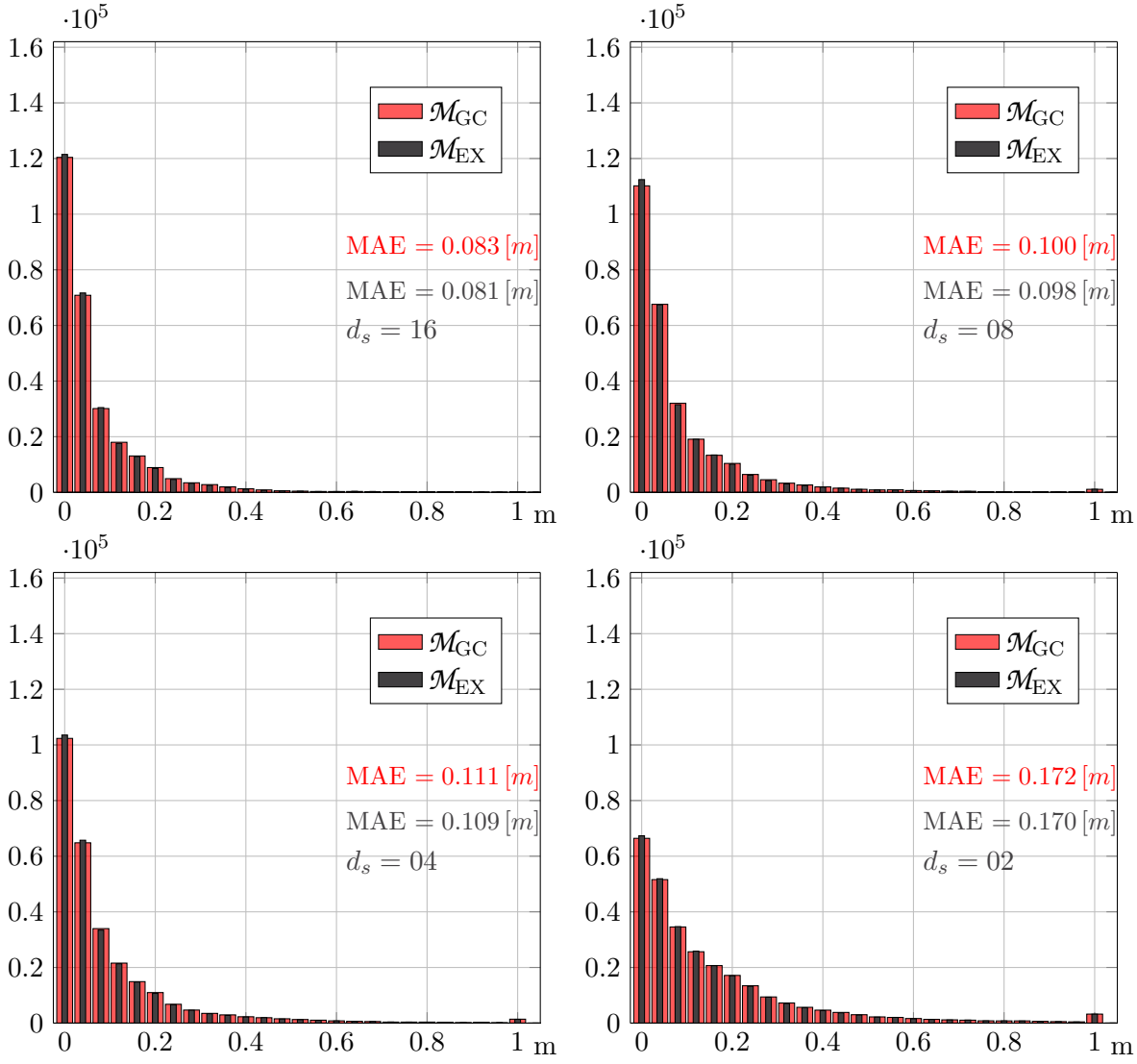


Figure 5.14: Histograms of absolute geometrical errors $\mid \Delta g_m^i \mid$ for a Stixel width $s_w = 3$ and the disparity sampling rates $d_s = 16$, $d_s = 08$, $d_s = 04$, and $d_s = 02$ for both methods $\mathcal{M}_{\mathrm{GC}}$ and $\mathcal{M}_{\mathrm{EX}}$. The histograms also include the mean absolute errors ($\mathrm{MAE_m}$). The histograms are clipped by 1 m.

Although no improvements are observed using the MAE as measure, we are still able to achieve geometrical accuracies in the range of a grid cell resolution. It is able to create occupancy grid maps with geometrical accuracies under 10 cm as long as at least a disparity sampling rate of $d_s = 08$ is chosen. We state this achievement as a success in this thesis. Here, the influence of the Stixel width is not the important factor which was already mentioned in Sec. 5.1.2.3. The disparity sampling rate is the key parameter in this case.

The MAE does not exploit all information since the weight information, and consequently the relative importance of each geometrical error, is neglected. Therefore, the WMAEs are also presented and discussed in this section. Taking the weights into account, the novel mapping approach performs in average better than method $\mathcal{M}_{\mathrm{EX}}$ which can be observed in Tab. 5.2.Method $\mathcal{M}_{\mathrm{GC}}$ performs best with a $\mathrm{WMAE}_{\mathrm{GC}}$ of 0.066 m using $s_w = 1$ and the disparity sampling rate $d_s = 08$. Method $\mathcal{M}_{\mathrm{EX}}$ has its best performance with a setup of $s_w = 1$ and $d_s = 16$. Here the $\mathrm{WMAE}_{\mathrm{EX}}$ is 0.070 m.

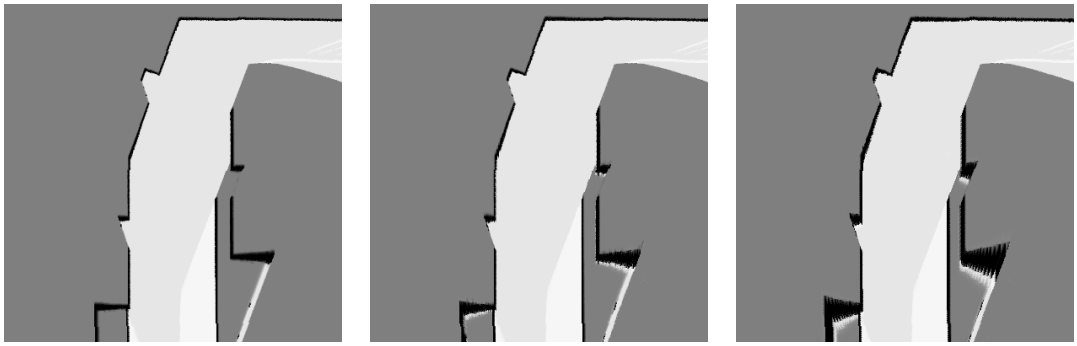
The largest WMAEs are 0.245 m for method $\mathcal{M}_{\mathrm{EX}}$ using the setup of $s_w = 1$ and $d_s = 02$, and 0.174 m for $\mathcal{M}_{\mathrm{GC}}$ using the setup of $s_w = 3$ and $d_s = 02$. With regard to the WMAEs, the benefit of method $\mathcal{M}_{\mathrm{GC}}$ is also shown in Fig. 5.13. Here, especially the configuration for the disparity sampling rates $d_s = 02$ and $d_s = 08$ in combination with a Stixel size of $s_w = 5$ should be mentioned. The $\mathrm{WMAE}_{\mathrm{GC}}$ is 0.143 m and the $\mathrm{WMAE}_{\mathrm{EX}}$ is 0.176 m for a disparity sampling rate of $d_s = 02$. For $d_s = 08$ the $\mathrm{WMAE}_{\mathrm{GC}}$ is 0.077 m and therefore almost 4 cm better than the WMAE for method $\mathcal{M}_{\mathrm{EX}}$. It surprises, that the method $\mathcal{M}_{\mathrm{EX}}$ performs slightly better than $\mathcal{M}_{\mathrm{GC}}$ with a setup of $s_w = 5$ and $d_s = 16$. No benefits are visible using the Stixel width $s_w = 9$, except by using the disparity sampling rate $d_s = 02$.

Taking the distribution of the estimated slopes in Fig. 5.12 into account, the reason for a better performance of the novel mapping approach becomes visible. The amount of slopes values which are equal or smaller than 0.004 is by a factor of about 1.50 higher for method $\mathcal{M}_{\mathrm{GC}}$ than for method $\mathcal{M}_{\mathrm{EX}}$. This means that for method $\mathcal{M}_{\mathrm{GC}}$ more steeper sigmoid functions exists, and consequently more sharper obstacles in the occupancy grid maps are available. We also observe that the total number of slope values equal or larger than 0.4 is considerably smaller for the novel mapping approach than for method $\mathcal{M}_{\mathrm{EX}}$. The mean slope value for $\mathcal{M}_{\mathrm{GC}}$ is 0.051 and 0.052 for method $\mathcal{M}_{\mathrm{EX}}$. Because of these insights, we state that we are able to generate more precise static obstacles with the novel approach, as long as a meaningful parameter setting is chosen (like $s_w = 3$ and $d_s = 16$).

5.1.4 Summary and Final Discussion Using Artificial Ground Truth Data

5.1.4.1 Summary

In this section we presented quantitative evaluation results based on artificial data and optimum conditions. Results of detection rates, as well as geometrical errors of obstacles were presented. We validated the novel occupancy grid map approach against a method which does not take the dependency of neighboring grid cells into account. The results show that static obstacles, as well as free space is generated in a precise and reliable way. This statement is confirmed by obstacle detection rates and free space detection rates larger than 95 % using a disparity sampling rate of at least $d_s = 08$. We observed that a Stixel width of 3, 5, or 7 should be chosen. It was shown that the novel mapping approach performs better with regard to obstacle detection rates. Here, an enhancement of 0.42 % compared to method $\mathcal{M}_{\mathrm{EX}}$ was observed. The geometrical error analysis pointed out, that it is important to take weight information into account. The WMAEs are in the range of 10 cm or less if a disparity sampling rate of at least $d_s = 08$ is chosen.



(a) $s_w = 5\,pel$, $d_s = 16$ (b) $s_w = 5\,pel$, $d_s = 4$ (c) $s_w = 5\,pel$, $d_s = 2$

Figure 5.15: The influence of the quantization error and the wedge effect based on the disparity interval steps $d_s = 16, 04, 02$. The smaller the disparity steps d_s, the bigger is the wedge effect which is clearly shown for horizontally orientated walls. Appendix B.2 presents all 20 configurations.

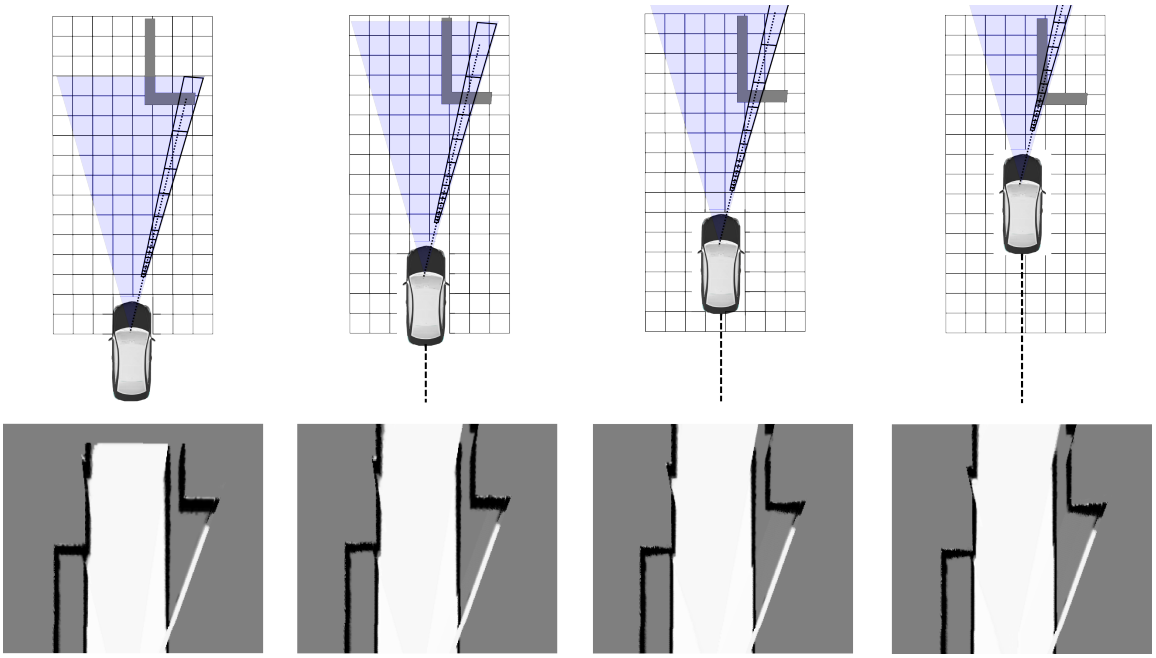


Figure 5.16: Explanation of the *wedge effect* during four incremental map update steps. The light blue rectangles describe the limited field of view of the stereo camera. The dotted line represents the driven path of the vehicle. When the vehicle approaches the L-shaped obstacle on the right, only areas, which are in the field of view of the camera, are updated. Because of the incremental map update and a finer disparity grid resolution, the contour of the obstacles becomes more precise during approaching. This results in a horizontal orientated wall which has the form of a wedge.

5.1.4.2 Final Discussion

The Secs. 5.1.2-5.1.3 show that over-segmentation as well as geometrical errors grow systematically with regard to the lateral distance between obstacles and vehicle. These insights are not proven in a quantitative way, but they become apparent taking the example maps shown in Fig. 5.5, Fig. 5.6, and Fig. 5.9 into account. On the one hand GT and the estimated maps are aligned perfectly in many regions, especially for vertically orientated walls close to the vehicle's position.

On the other hand, horizontally orientated walls which are far away from the vehicle are considerably wider and more spread out than GT provides. The variation of the dimension of obstacles is caused by the quantization of the disparity space and the transformation in the Cartesian grid map afterwards. If Stixels are observed in the outer bound of the field of view of the stereo camera, one cell in the column-disparity space influences several Cartesian grid cells which can be seen in Fig. 5.8. The smaller the value d_s, the more Cartesian grid cells are effected which results in a larger expansion of objects. This is clearly visible in Fig. 5.15.

Only if the vehicle approaches the object and the object is still in the field of view of the camera, the previous mentioned effect is decreasing until several column-disparity cells fall into one Cartesian grid cell. Because of the incremental map update, the contour of these obstacle parts which are repeatedly observed becomes more precise during approaching. As seen in e.g. Fig. 5.9, this results into static obstacles which have the form of a wedge. We define this phenomena as the *wedge effect* which is also illustrated in Fig. 5.16 and Fig. 5.15. The choice of a large d_s helps to reduce the influence of the wedge effect and produce more precise maps.

One of the core achievements of this evaluation section is to find a meaningful configuration with regard to the disparity sampling rate and Stixel width. Based on the presented results, we come to the decision that a Stixel width of 3 and a disparity sampling rate of $d_s = 16$ should be chosen for the novel mapping approach. Obviously we can create more precise occupancy maps using the novel mapping approach compared to occupancy grid mapping techniques which do not take the dependency of neighboring grid cells into account.

The influence of the parameters during prediction as well as the parameters for the binary terms were not discussed in this section. This evaluation should be considered in future evaluation steps. We did not add artificial noise to the GT disparity images to measure the influence of outliers. This should also be done in future.

5.2 Evaluation with Real-World Data

In this section we evaluate the novel mapping approach with the help of real-world data. At the beginning, the used data set is introduced, and how reference occupancy grid maps $\mathcal{M}_{\text{RE}}$ are generated. In Sec. 5.2.2 we explain how we apply our novel mapping approach based on this data. We also present first qualitative results in this section. Similar to the previous evaluation section, classification and geometrical accuracies are presented in Sec. 5.2.3 and Sec. 5.2.4 respectively. At the end, we give a conclusion of the evaluation with real-world data.

5.2.1 Description of the Data Set

This section describes the data set which is used for real-world evaluation. The KITTI vision benchmark suite (Sec. 5.2.1.1), and the process of reference occupancy grid map estimation (Sec. 5.2.1.2) is introduced in the following subsections.

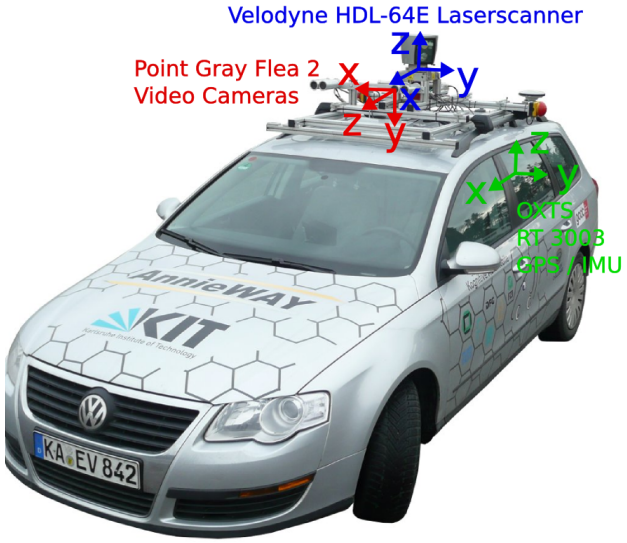
5.2.1.1 The KITTI Vision Benchmark Suite

For the evaluation of the novel mapping approach we use the raw data sets of the real-world KITTI vision benchmark suite [Geiger et al., 2012, 2013]. Based on the autonomous driving platform Annieway [Kammel et al., 2008], the team around Geiger et al. [2012] created these data sets which include

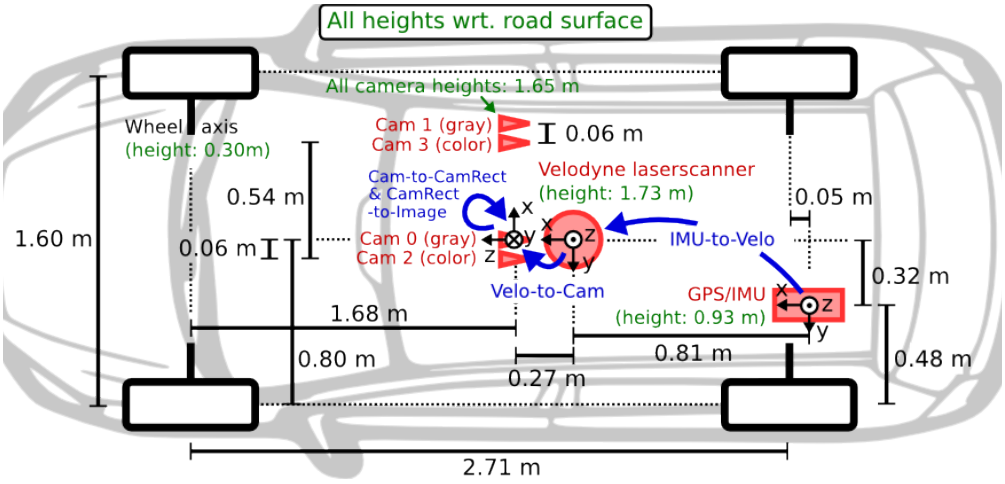
- rectified 8 bit gray-scale stereo image sequences with a resolution of 1242×375 pel, recorded with a frequency of 10 Hz,
- 3D Velodyne point clouds [Velodyne, 2010] with 100 000 points per frame, recorded with a frequency of 10 Hz,
- and data of a high precision IMU/GPS system which provides position, speed and acceleration data, recorded with a frame rate of 100 Hz.

The different data types are synchronized to each other, and internal as well as external calibration parameters of the sensors are available. The master camera is triggered when the laser scanner is facing in forward direction. In addition, an open source development kit in Matlab and C++ is provided which allows an easier handling of the data. The test vehicle and the alignment of the sensors are presented in Fig. 5.17. Example images of the data set are shown in Fig. 5.18. The research team captured sequences in urban, residential, and highway situations around the city of Karlsruhe in Germany.

The described data set fits perfectly to our interests which is explained in more detail below. The precise position information based on the IMU/GPS system allows us to apply "mapping with known poses" also for real-world data. The rectified stereo image sequences with their provided calibration parameters are converted into our internal framework which allows us to apply the image processing steps of SGM estimation, Stixel estimation, and Stixel segmentation (see Sec. 3.2.3). Based on these preprocessing steps, the novel grid mapping approach is carried out. The point clouds of the Velodyne laser scanner are used to create reference occupancy grid maps which is presented in the section below (see Sec. 5.2.1.2).



(a) The test vehicle



(b) The alignment of the sensors

Figure 5.17: The used test vehicle and the sensor alignment of the KITTI vision benchmark suite [Geiger et al., 2012]. The test vehicle is a Volkswagen Passat B6. The multi sensor system is equipped with stereo cameras (colored and gray-scaled), a Velodyne HDL-64E and the deeply coupled IMU/GPS navigation system OXTS RT 3003 (see Fig. 5.17(a)). As one can see in Fig. 5.17(b), a full description of the external calibration parameters is given. This makes the data set transparent and also unique. Detailed information and access to the raw data sets can be found in [Geiger et al., 2012, 2013]. The images were taken from Geiger et al. [2012].

Figure 5.18: Sample images of the KITTI vision benchmark suite. The team around Geiger et al. [2012] captured the sequences in urban, residential, and highway scenarios around the city of Karlsruhe, Germany. In these samples, only images are shown which were used in the evaluation. Here, our focus is on residential and urban scenarios.

For the evaluation eight different sequences are chosen whose characteristics are listed in Tab. 5.3. Here, we focus mainly on residential areas. With regard to the captured environment and their effects on the mapping results, the sequences vary in their level of difficulty. The sequences 0022, 0023, 0039, 0064 and 0095 include residential areas with narrow streets, less traffic and well structured environment. In sequence 0087, a small trail with unstructured environment, lots of vegetation, and trees was recorded. Sequence 0091 represents an inner city scenario with lots of pedestrians, sitting people, and street furniture. Sequence 0033 is the longest sequence which includes mainly residential areas, but also parts of open space and rural roads.

Table 5.3: List of the used KITTI image sequences, the map sizes, and their properties.

Name	**Map size [pel]**		**Properties**	
26.09.11	height	width	frames/min	scene description
0022	2600	2200	800/1:20	residential area, narrow streets, well structured environment
0023	4600	700	474/0:48	residential area, narrow streets, well structured environment
0039	3100	700	395/0:40	residential area, narrow streets, well structured environment
0064	1600	4500	570/0:57	residential area, narrow streets, well structured environment
0087	3200	1300	728/1:13	small trail, bushes/vegetation
0091	2400	1000	339/0:34	inner city,42 pedestrians, sitting people, street furniture
0095	2900	700	268/0:27	residential area, narrow streets, well structured environment
30.09.11				
0033	6400	5600	1594/2:40	residential area, open space, rural roads
		in total:	5168/8:39	

5.2.1.2 Reference Occupancy Grid Map Estimation

Under the assumption that the pose information of the IMU/GPS unit is correct, the Velodyne laser point clouds are used to generate reference occupancy grid maps $\mathcal{M}_{\mathrm{RE}}$. We take the reasonable assumption that the precision of the Velodyne points is much more precise than the Stixel depth information computed from disparity images. This statement is valid especially for far away measured obstacles. The generation of the reference map is a semi-automated procedure. The main processing steps are also visualized in Fig. 5.19. At the beginning, ground plane points are estimated using a RANSAC plane fitting technique. These points are removed afterwards. This results

in reduced point clouds which represent potentially obstacle information. To allow a meaningful comparison between reference and estimated data, only Velodyne points are considered which are in the field of view of the stereo camera. This is realized by projecting the reduced 3D points into the image plan using the given internal and external calibration files (see also Fig. 5.17(b)). Since we provide 2D maps, the reduced Velodyne points are projected into the ground plane. All these points potentially represent static obstacles which should also be observed by the stereo camera. For free space estimation, points are sampled along the rays between the current position and the static points. Afterwards, these sampled points are classified as free space points. The static object points as well as the free space points are integrated over time with the use of the given pose information into a global, earth fixed Cartesian grid map with a cell resolution of 10 cm. Here, we do not use any uncertainty model and count hits per cell only. At the end, we use thresholds to define, when a grid cell is free and occupied, respectively. This results in a coarse reference grid map which is also shown in Fig. 5.19. In the last step, the coarse reference maps are inspected by hand to remove dynamic obstacles, smooth the free space area, and remove outliers and clutter. The final reference occupancy grid maps $\mathcal{M}_{\mathrm{RE}}$ are shown in Fig. 5.20, Fig. 5.22, and Appendix C.1.

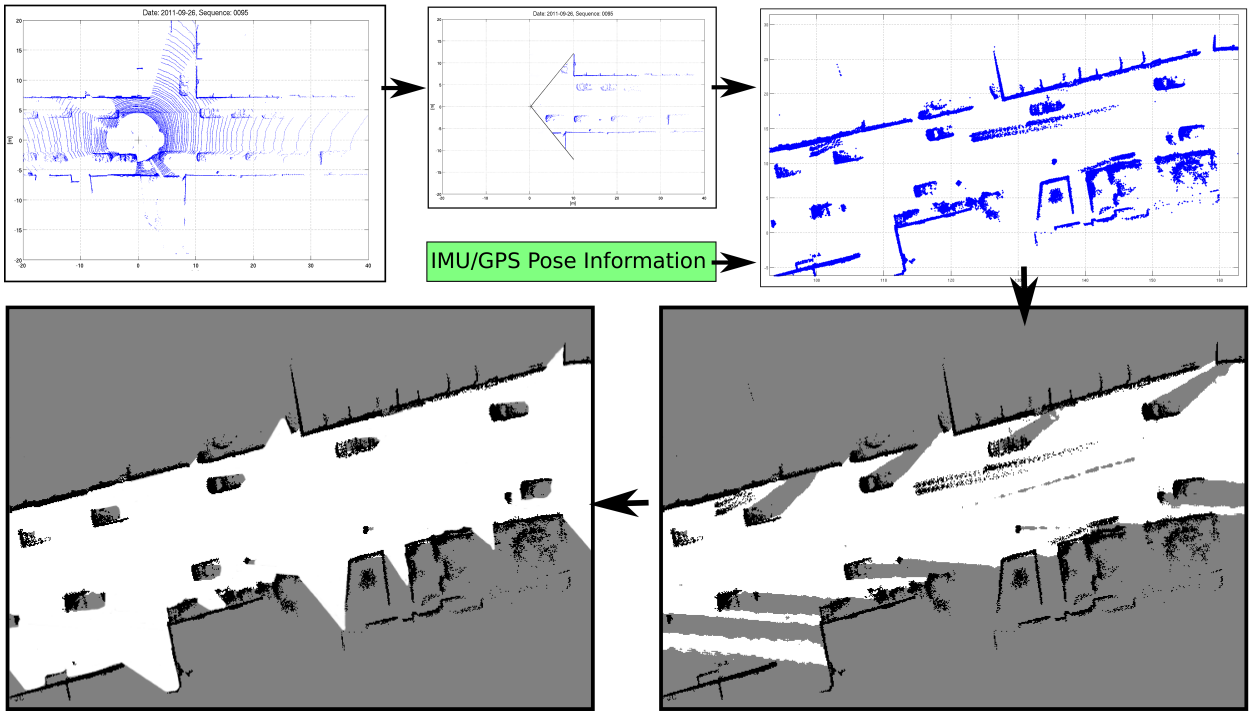
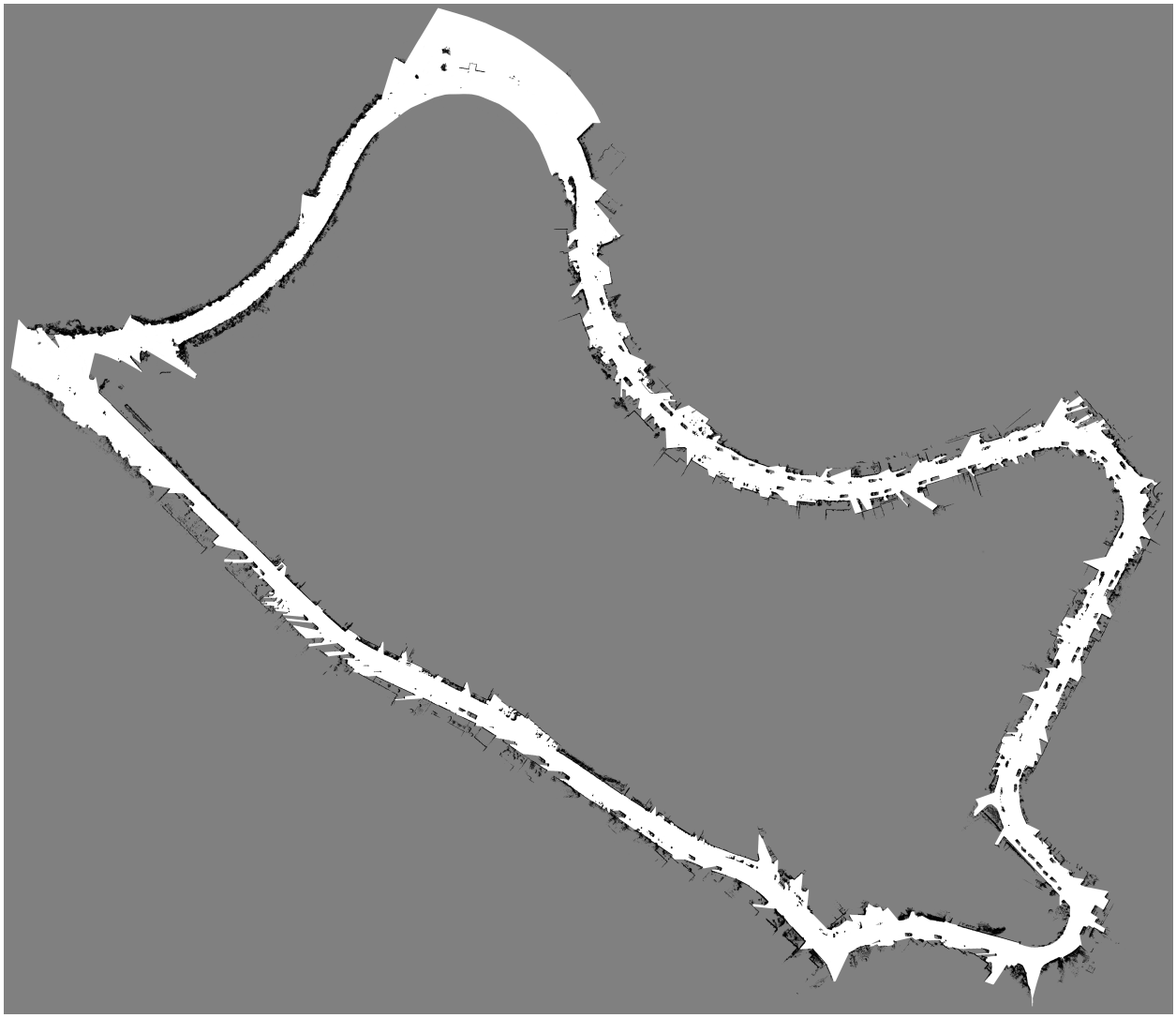
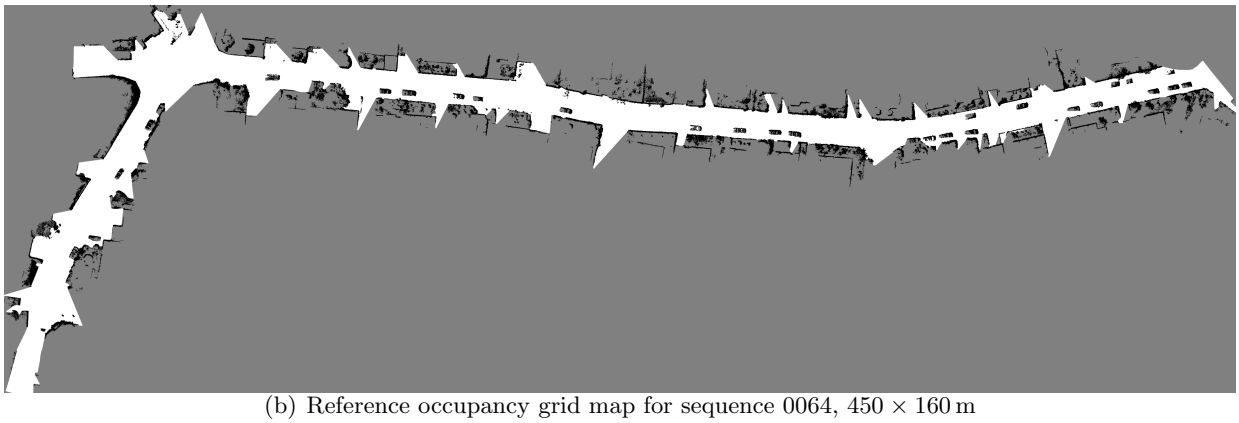


Figure 5.19: Process chain of reference occupancy grid map generation with the help of raw data sets of the real-world KITTI vision benchmark suite [Geiger et al., 2012, 2013]. The original Velodyne point clouds are used as input (top left). The preprocessed object points (top middle) are integrated over time using IMU/GPS position information (top right). The coarse reference occupancy grid map (bottom right) is generated by counting hits per cell and thresholding. In the final step, dynamic obstacles, outliers, and clutter are removed by hand. The free space is also adjusted (bottom left).

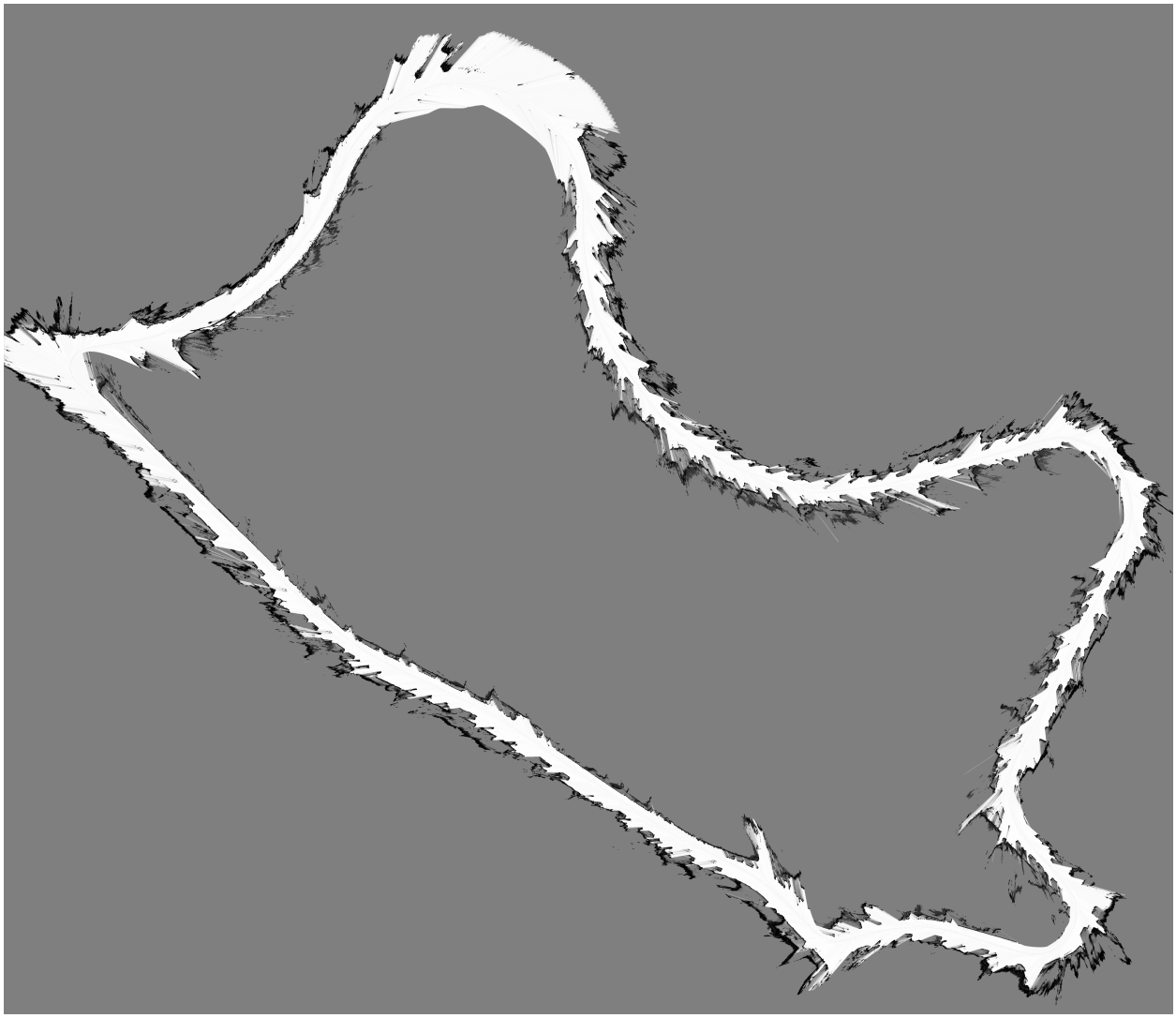


(a) Reference occupancy grid map for sequence 0033, 640 × 560 m

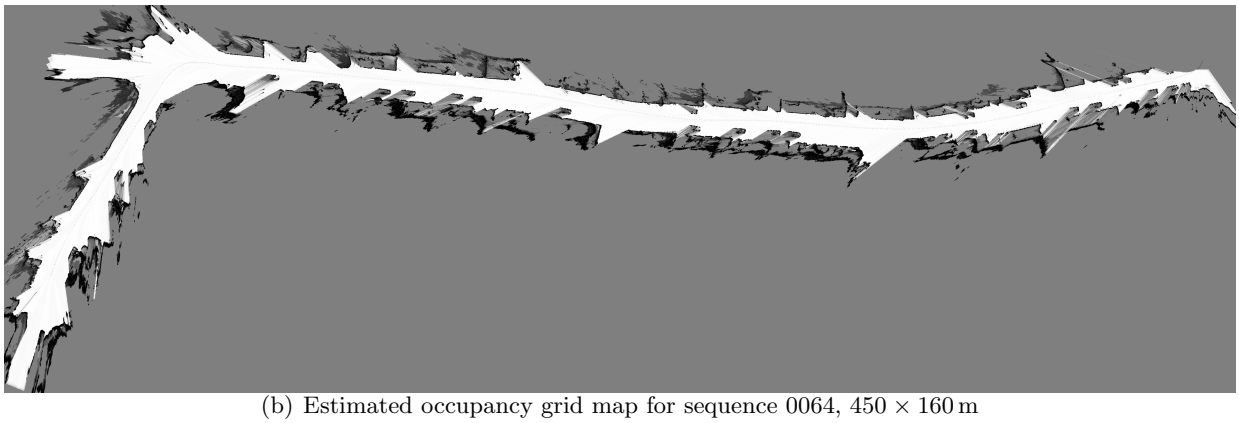


(b) Reference occupancy grid map for sequence 0064, 450 × 160 m

Figure 5.20: Reference occupancy grid maps for sequences 0033 (5.20(a)) and 0064 (5.20(b)) based on raw Velodyne point clouds and IMU/GPS data of the KITTI vision benchmark suite [Geiger et al., 2012, 2013]. The images are scaled to fit best on page.

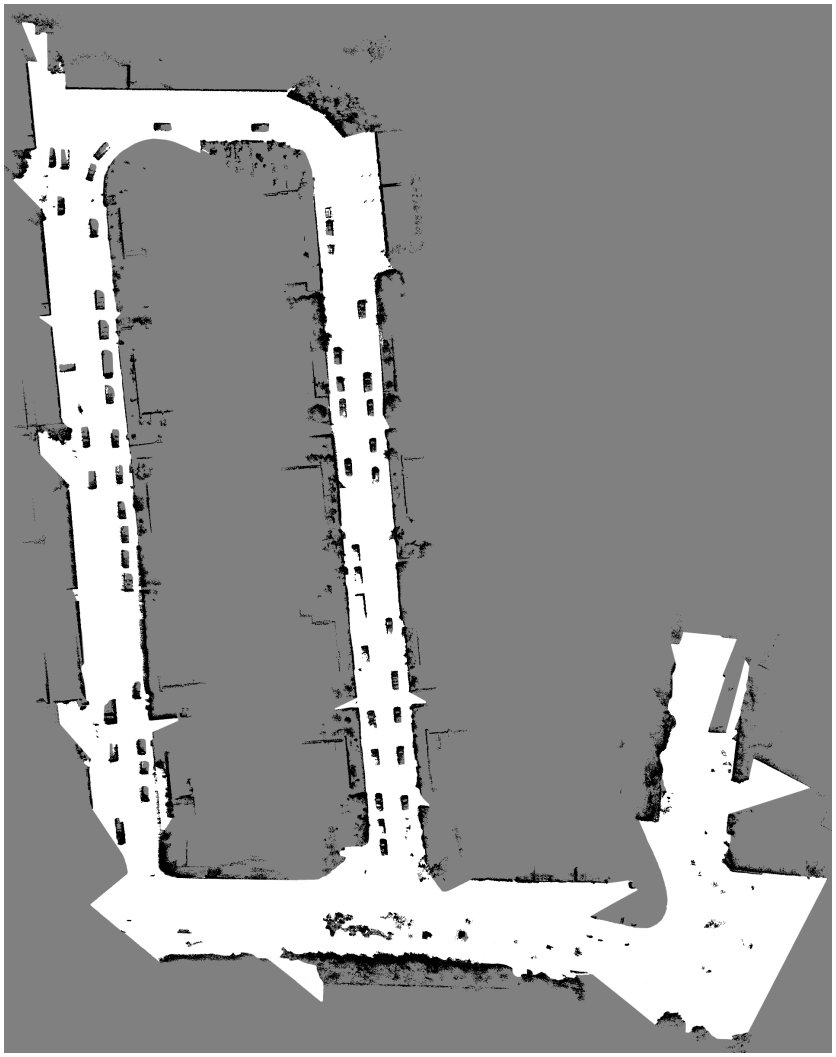


(a) Estimated occupancy grid map for sequence 0033, 640 × 560 m

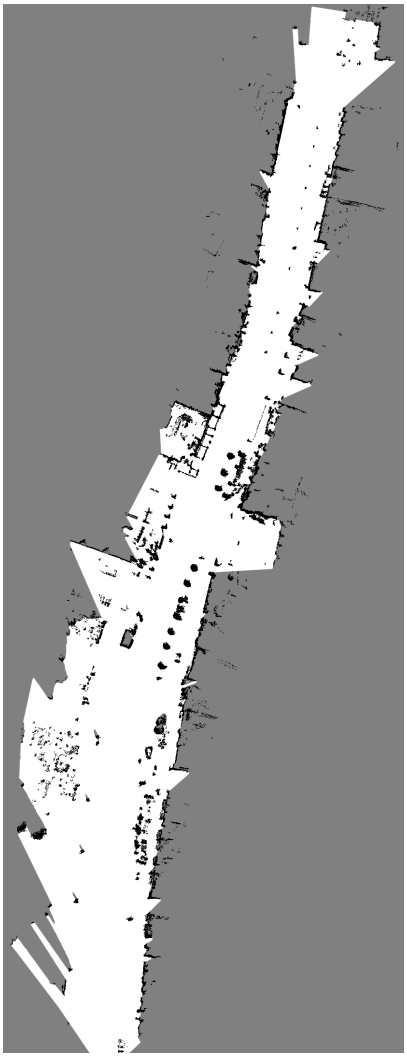


(b) Estimated occupancy grid map for sequence 0064, 450 × 160 m

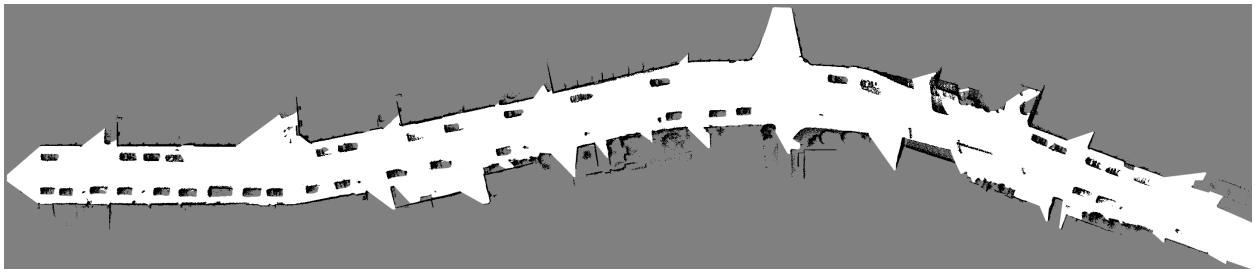
Figure 5.21: Estimated occupancy grid maps using the novel approach for sequences 0033 (5.21(a)) and 0064 (5.21(b)). As input stereo image sequences and IMU/GPS data of the KITTI benchmark suite are used. The images are scaled to fit best on page.



(a) Reference occupancy grid map for sequence 0022, 260 × 220 m

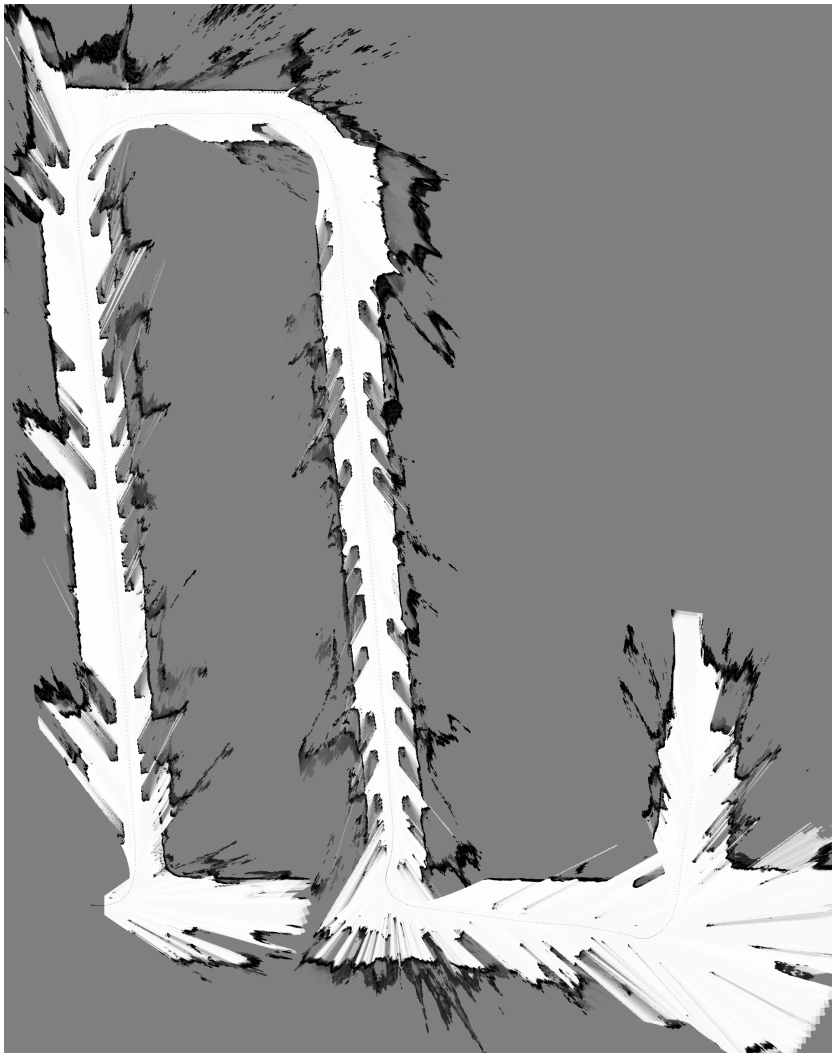


(b) Reference occupancy grid map for sequence 0091, 100 × 240 m

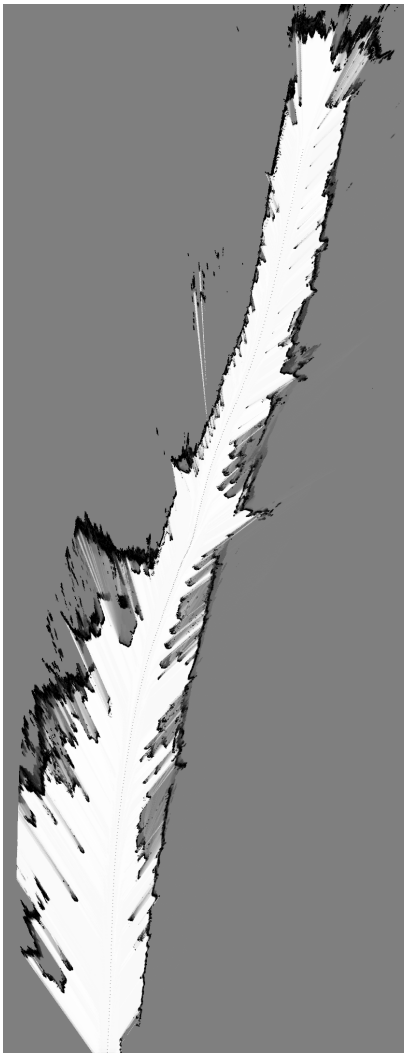


(c) Reference occupancy grid map for sequence 0095, 290 × 70 m

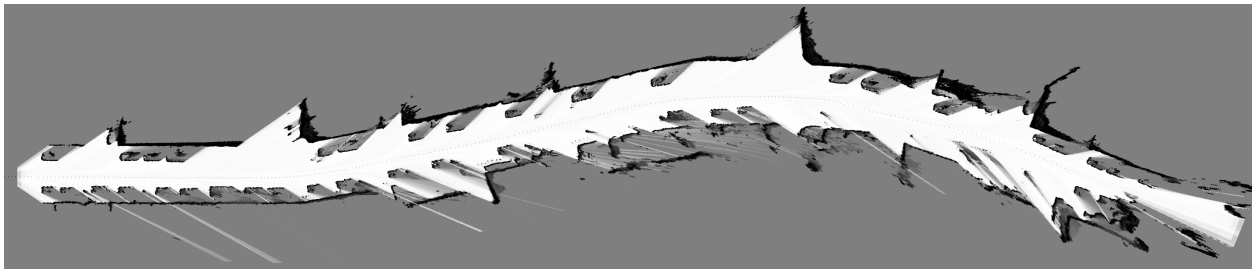
Figure 5.22: Reference occupancy grid maps for sequence 0022 (5.22(a)), 0091 (5.22(b)), and 0095 (5.22(c))). The images are scaled to fit best on page.



(a) Estimated occupancy grid map for sequence 0022, 260 × 220 m



(b) Estimated occupancy grid map for sequence 0091, 100 × 240 m



(c) Estimated occupancy grid map for sequence 0095, 290 × 70 m

Figure 5.23: Estimated occupancy grid maps using the novel approach for sequences 0022 (5.23(a)), 0091 (5.23(b)), and 0095 (5.23(c))). The images are scaled to fit best on page.

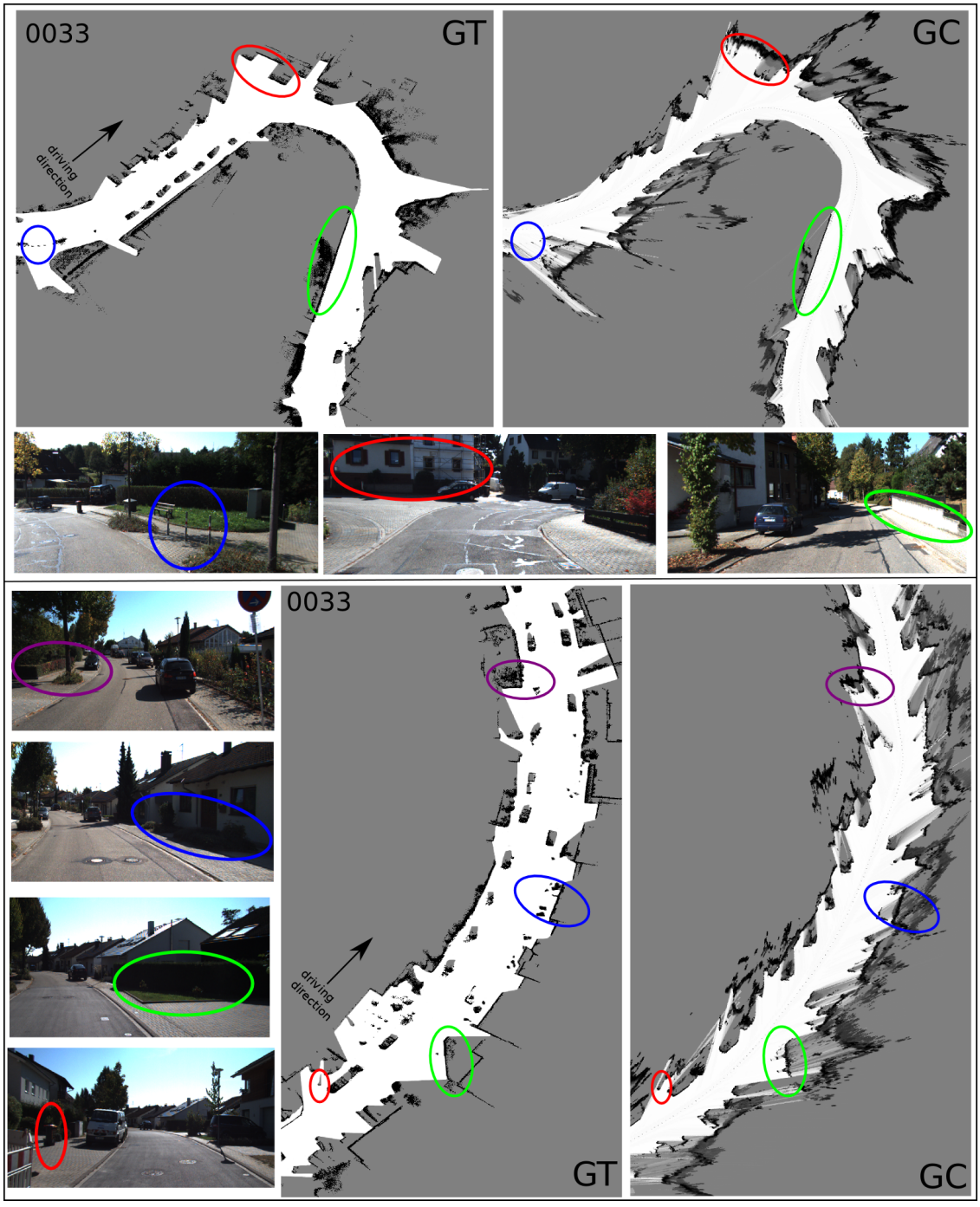


Figure 5.24: Close-ups of $\mathcal{M}_{\mathrm{GC}}$ and $\mathcal{M}_{\mathrm{RE}}$ for sequence 0033. The results for a sharp right turn in a residential are shown in the second example. As one can see, vertically orientated structures like walls (green marker) and house facades (red marker) are constructed correctly. Even objects with small extension like poles are mapped correctly (blue marker). In the example given below, hedges (purple and green marker), house facades (blue marker) and a trash bin (red) are highlighted. The situation marked by the green circle shows that a small bush is missing in the reference data.

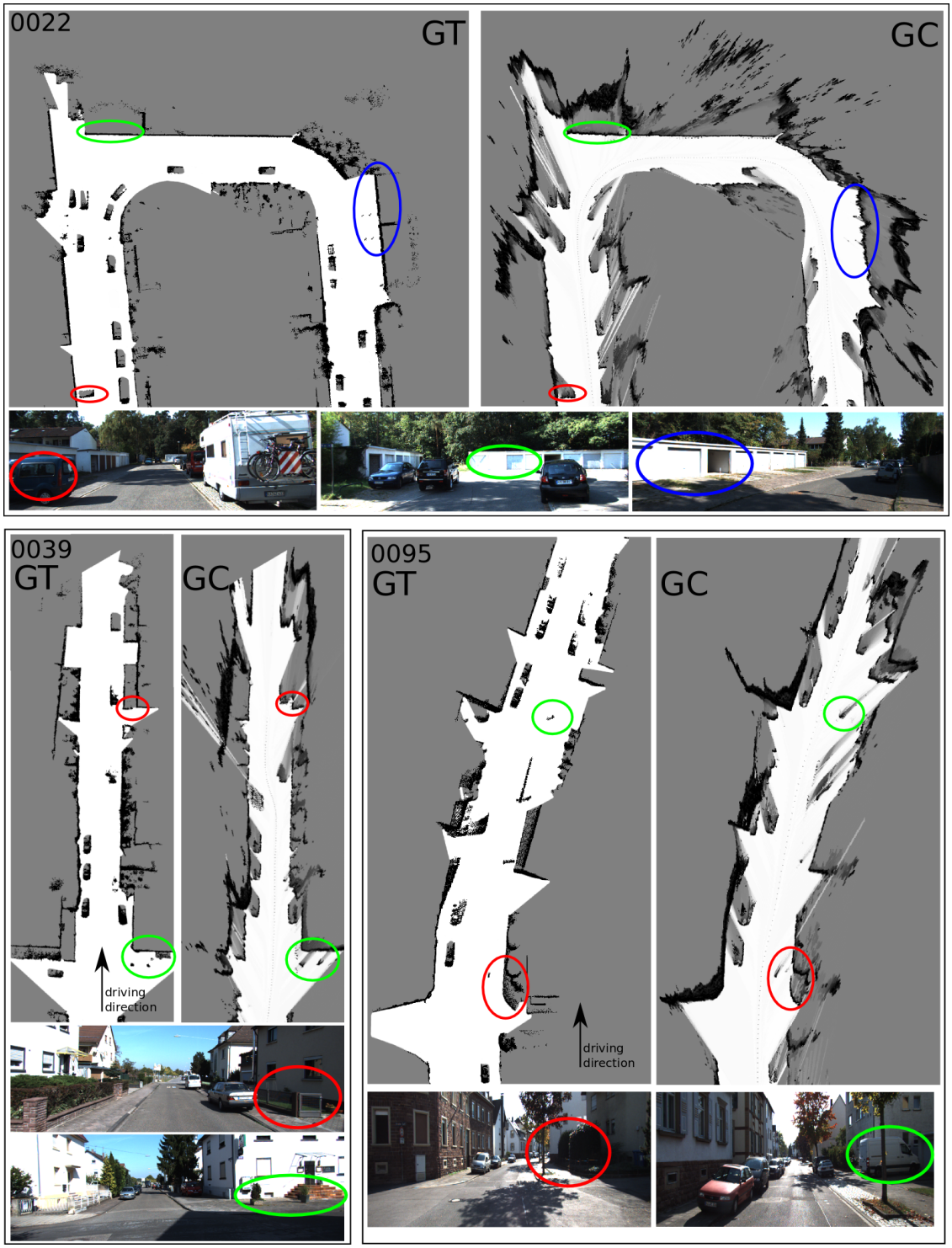


Figure 5.25: Close-ups of $\mathcal{M}_{\text{GC}}$ and $\mathcal{M}_{\text{RE}}$ for sequence 0022, 0039, and 0095. All close-ups show map results of residential areas. Although parked cars are not long-term static obstacles, they are still mapped precisely (see sequence 0022, red marker). Man-made obstacles like garages are mapped nicely, but tree rows and huge bushes produce a lot of clutter in the grid maps. The other close-ups illustrate how accurate small fences (0039, red marker), garden accessories (0039, green marker), and single trees/bushes close to buildings (0095, green and red marker) are mapped.

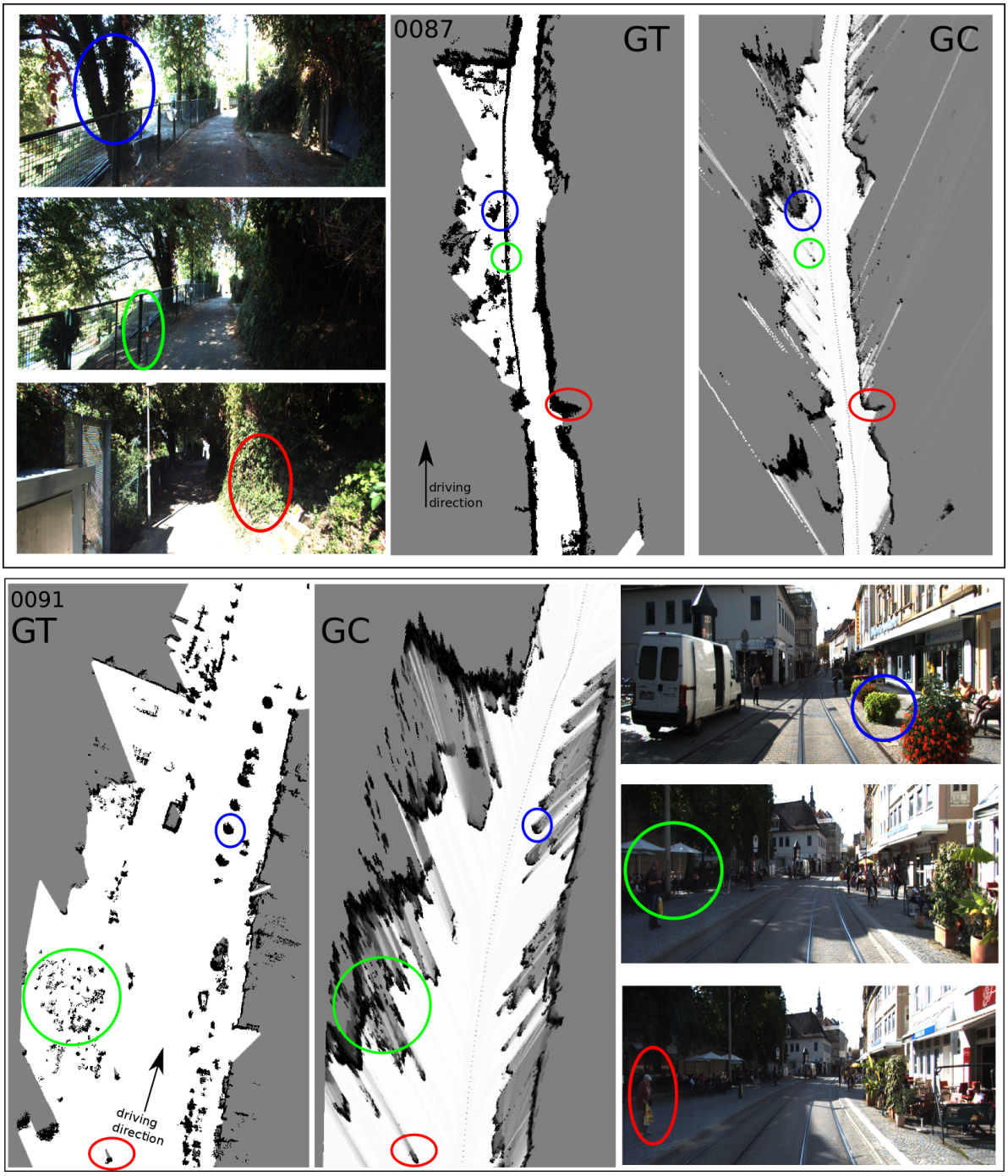


Figure 5.26: Close-ups of $\mathcal{M}_{\text{GC}}$ and $\mathcal{M}_{\text{RE}}$ for sequence 0087, and 0091. These two sequences are the most difficult ones. Unstructured, chaotic environment with lots of vegetation and trees was recorded in 0087. In sequence 0091, a busy inner city scenario with lots of pedestrians and street furniture was captured. Street furniture on the right is represented correctly in the occupancy grid map and have a high overlap with the reference data. On the other hand, tables, parasols and sitting people on the left side produce a lot of clutter and noise in the resulting grid map.

5.2.2 Qualitative Results

In this section qualitative results are presented to allow first insights how good the novel mapping approach performs against real-world reference data. In Sec. 5.2.2.1, the parameter settings and the conditions during the map building process are described. Afterwards, occupancy grid map results are shown and discussed in a qualitative way (see Sec. 5.2.2.2).

5.2.2.1 Parameter Settings

As already mentioned in Sec. 5.2.1.1, the provided rectified stereo image sequences and the IMU/GPS data of the chosen KITTI sequences (see Tab. 5.3) are used as input data for our novel mapping approach. For the occupancy grid building process we use nearly the same parameter settings than described in Sec. 5.1.1.3. Here, the major differences exist in the input data. The images have a size of 1242(W) $\times$ 375(H) pel and are recorded with 10 Hz. The baseline b is 0.53 m and the focal length is 721.53 pel. The horizontal field of view is 81 deg. The height of the camera is 1.65 m above ground. Based on the evaluation with artificial sequences (see Sec. 5.1.4) we chose a Stixel width of $s_w = 3$ pel and a disparity resolution of $d_s = 16$. Tab. 5.3 includes the different map sizes which are used to preallocate memory for the occupancy grid maps on the GPU. The grid cell resolution is 10 cm.

5.2.2.2 Map Results

Figure 5.21 presents the final results of the novel occupancy grid mapping approach for the sequences 0033 and 0064. The results for sequences 0022, 0091, and 0095 are shown in Fig. 5.23. The occupancy grid maps for 0039, 0023, and 0087 are illustrated in Appendix C.1. As defined in Sec. 5.1, the novel occupancy map results are named by $\mathcal{M}_{\mathrm{GC}}$. To allow a better qualitative comparison between reference and estimated maps we present close-ups of both maps in Figs. 5.24-5.26. These figures also include real-world images for a better interpretation. Important regions in the maps are also highlighted with colored circles. In Appendix C.1, Fig. C.4 additional close-ups are shown. A sample of the comparison between results of the novel mapping approach with results of the method $\mathcal{M}_{\mathrm{EX}}$ is illustrated in Fig. 5.27.

5.2.2.3 Discussion

The generated occupancy grid maps represent a huge variety of environment information since the chosen sequences include a lot of different situations and scenarios. This allows us to make qualitative assessments where the novel mapping algorithm performs well and where the algorithm reaches its limits. The close-ups in Figs. 5.24-5.26 give us first insights without any quantitative evaluation steps.

As one can see in all examples, man-made structures with vertically orientated surfaces like walls, house facades, small garden fences, or garages are constructed precisely. Although parked cars are not long-term static obstacles, they are still mapped correctly. They are represented in the occupancy grid maps as typical L-shaped forms. Even static obstacles with small extension, like poles, trash bins and garden accessories, are represented in the maps as static obstacles (see Fig. 5.24).

With regard to spatial dimension and quality of static obstacles, especially the representation of vegetation differs a lot. On the one hand, single trees, small bushes and hedges close to the vehicle are mapped correctly. These insights are shown in Fig. 5.25, sequence 0039 and 0095. On the other hand, unstructured environment like tree rows produces a lot of clutter in the grid maps. This is observed in parts of sequence 0022 (see Fig. 5.25), parts of sequence 0033 (see Fig. 5.24), and in sequence 0087. As already mentioned in Sec. 5.2.1.1, sequence 0087 includes challenging situations with unstructured, chaotic environment.

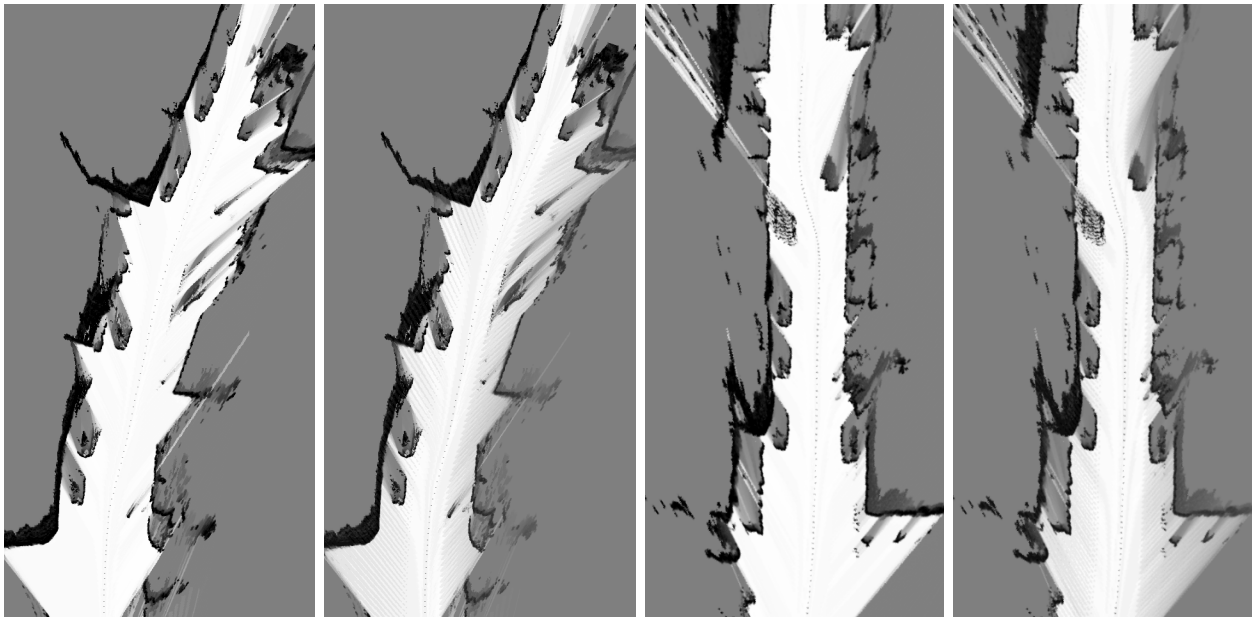
It seems that the algorithm reaches its limit in situations like these. Another difficult scenario is shown in sequence 0091, where a busy inner city scenario with lots of pedestrians and street furniture was captured. As seen in Fig. 5.26, street furniture close to the right side of the vehicle is represented correctly and has a high overlap with the reference data. On the other hand, tables, parasols and sitting people on the left side produce a lot of clutter and noise in the occupancy grid map. Here, the algorithm also reaches its limit. As already stated in Sec. 5.1.4, we observe that the quality of occupied areas strongly depends on the distances of obstacles with regard to the vehicle, and how often the corresponding map area was updated during the incremental mapping approach. Using real-world data, we also have to cope with difficult weather and light conditions, like e.g. reflections, over-, or underexpose, and image saturation, during stereo estimation which influence the Stixel precision, and consequently the map quality.

A brief qualitative comparison between the novel mapping approach $\mathcal{M}_{\mathrm{GC}}$ with the approach $\mathcal{M}_{\mathrm{EX}}$ results in the following statements. Taking the close-ups in Fig. 5.27 into account, both methods show similar results with regard to occupied areas, free space and unknown areas. But taking a closer look to the results, important differences occur.

It seems that the transition between free space and occupied areas is sharper and more precise using the novel mapping approach. Based on visual inspection, occupied areas are estimated more reliable than using the method $\mathcal{M}_{\mathrm{EX}}$ in some regions. Furthermore, the whole free space estimation looks more confident and smoother. Please take into account that these statements are based on eye balling. In the following Secs. 5.2.3-5.2.4, we make assessments with regard to classification and geometrical accuracies, and taking into considerations all statements and insights of the current and previous sections.

5.2.3 Classification Accuracies

Classification accuracies based on real-world data are presented in this section. This experiment follows exactly the same procedure described in Sec. 5.1.2.1. To estimate the detection rates for free space and obstacles, we use the threshold 0.6 to define occupied areas, and the threshold 0.3 to classify free space areas. Since we generated reference maps and not ground truth data, we state that $\mathcal{M}_{\mathrm{RE}}$ is very precise, but not perfect. This is caused by remaining errors in calibration, synchronization, and the uncertain behavior of the Velodyne laser scanner. Because of this fact, we define a confidence interval of 2×2 grid cells. As in Sec. 5.1, we also compare the novel mapping approach $\mathcal{M}_{\mathrm{GC}}$ with the approach without MRFs $\mathcal{M}_{\mathrm{EX}}$.



(a) $\mathcal{M}_{\mathrm{GC}}$ (0095) (b) $\mathcal{M}_{\mathrm{EX}}$ (0095) (c) $\mathcal{M}_{\mathrm{GC}}$ (0039) (d) $\mathcal{M}_{\mathrm{EX}}$ (0039)

Figure 5.27: Qualitative comparison between results of the novel mapping approach $\mathcal{M}_{\mathrm{GC}}$ with results of the method $\mathcal{M}_{\mathrm{EX}}$ for two close-ups of sequences 0095 and 0039.

5.2.3.1 Results of Detection Rates

In Tab. 5.4 we present the classification accuracies for the eight different KITTI sequences. The table also shows the results for method $\mathcal{M}_{\mathrm{EX}}$. In each case, we mark the best results in blue and the lowest results in red. In the last column the weighted mean for the detection of free space and obstacles is also shown. Each weight is based on the length of each sequence.

Similar to Fig. 5.7, the classification results are illustrated in a diagram where we plot the detection rates of free space and occupied areas against each other (see Fig. 5.28). This allows an easier interpretation of the results. In Fig. 5.28, each sequence is encoded by a different symbol. The mapping method $\mathcal{M}_{\mathrm{GC}}$ is visualized in bright red, and method $\mathcal{M}_{\mathrm{EX}}$ in gray. Similar to Fig. 5.6, the overlay of reference and estimated map data is presented in Figs. 5.29-5.31. The figures show close-ups of the grid maps and areas with "over-segmentation" (red) and "under-segmentation" (blue). The figures also show real-world images of the scenes to allow a better interpretation of the grid map data.

5.2.3.2 Discussion

Free Space Discussion. As one can clearly see in Tab. 5.4 and Fig. 5.28, our novel mapping approach performs considerably better than $\mathcal{M}_{\mathrm{EX}}$ with regard to free space detection. Taking the weighted mean into account, 87.05 % are correctly detected as free space using our novel approach. In comparison, the method $\mathcal{M}_{\mathrm{EX}}$ has a weighted mean detection rate of 84.32 %. The novel method

$\mathcal{M}_{\mathrm{GC}}$ achieves the best detection rate of free space in sequence 0087 (90.71 %), and the lowest in sequence 0091 (76.30 %). For method $\mathcal{M}_{\mathrm{EX}}$, the best free space detection rate exists in sequence 0033 (88.77 %), and the lowest also in sequence 0091 (72.38 %).

Based on these results we establish that the modeling of neighboring cell dependencies, which are represented by the binary terms $E_{i,j}(m_{i,t}, m_{j,t})$, has strong influence with regard to a better free space estimation. The use of MRFs during occupancy grid mapping allows the estimation of a smoother and cleaner free space. This reduce the number of outliers and clutter and, therefore, increase the detection rates of free space compared to methods which do not take the dependencies of grid cells into account. The effect of a smoother and cleaner free space was already observed in Sec. 5.2.2.2, Fig. 5.27. At this point, we also have a quantitative proof.

Compared to the results based on artificial sequences (see Sec. 5.1.2, Tab. 5.1), the absolute detection rates for free space are considerably lower using real-world data (96.92 % vs. 87.05 %). Without doubt, one reason is the free space correction and annotation by hand during the creation of reference data which was mentioned in Sec. 5.2.1.2. In difficult situations, it is challenging to define exactly the correct class type occupied area and free space, respectively.

Another reason is that the segmentation process into dynamic and static Stixels (see Sec. 3.2.3) faces problems with very slow-moving objects. This can be seen in the right example of Fig. 5.30 and in the left close-up of Fig. 5.31: slow moving cars or even trucks which let pass the vehicle, are classified as static obstacles. This results in false positives (FP) with regard to free space annotation.

Because of the high precision of the LIDAR scanner and the high horizontal angular resolution of 0.08 deg [Velodyne, 2010], even gaps between finely structured obstacles in complex environment situations can be detected. These gaps are defined as free space, in general. Because we define a Stixel width of $s_w = 3$, only a horizontal angular resolution of $\frac{81 \text{ deg}}{1242 \text{ pel}} \times 3 \text{ pel} \approx 0.195$ deg is reached with the current camera and Stixel setup. Since this resolution is by a factor of about 2.4 larger with regard to the reference sensor, our algorithm is not able to detect these mentioned gaps. This is another reason for lower free space detection rates.

If we take the differences of the detection rates between both methods $\mathcal{M}_{\mathrm{GC}}$ and $\mathcal{M}_{\mathrm{EX}}$ into account, we see a strong benefit using the novel approach: during evaluation of artificial sequences we were not able to see a benefit using the novel mapping approach. Here, the detection rates for free space are $\mathcal{M}_{\mathrm{GC}}$= 96.92 % and $\mathcal{M}_{\mathrm{EX}}$= 96.96 %. However, using real-world data a considerable difference in the over all detection rate of free space is observable, namely $\mathcal{M}_{\mathrm{GC}}$= 87.05 % vs. $\mathcal{M}_{\mathrm{EX}}$= 84.32 %. This shows us, that the power of the novel approach becomes visible using real-world data which includes challenging situations, outliers, and difficult lightning conditions.

Occupancy Discussion. Taking the mean detection rates of occupied areas into account, the novel approach also performs better than method $\mathcal{M}_{\mathrm{EX}}$ with 75.88 %, compared to 74.96 %. The best detection rates are achieved in sequence 0039 ($\mathcal{M}_{\mathrm{GC}}$ = 85.15 % vs. $\mathcal{M}_{\mathrm{EX}}$ = 84.95 %), and the lowest detection rates in sequence 0091 ($\mathcal{M}_{\mathrm{GC}}$ = 64.89 % vs. $\mathcal{M}_{\mathrm{EX}}$ = 63.35 %). In seven sequences the detection rates of obstacles are higher if we apply the novel mapping approach. Only in sequence 0087 the method $\mathcal{M}_{\mathrm{EX}}$ performs slightly better. In Sec. 5.2.1 and Sec. 5.2.2 we observed that sequence 0091 is difficult since a busy inner city scenario is captured. This statement is coherent with the results above: for both methods the detection rates are the lowest ones.

Table 5.4: Classification accuracies for the eight selected KITTI sequences for method $\mathcal{M}_{GC}$ and method $\mathcal{M}_{EX}$. The highest (blue) and lowest rates (red) are also marked. Weighted means are also estimated.

	detection rate of obstacles [%]		detection rate of free space [%]	
	$\mathcal{M}_{EX}$	$\mathcal{M}_{GC}$	$\mathcal{M}_{EX}$	$\mathcal{M}_{GC}$
26.09.2011				
0022	73.52	74.68	83.39	87.85
0023	65.99	68.34	77.55	79.99
0039	84.94	85.15	84.30	86.44
0064	74.66	75.43	85.08	86.87
0087	72.17	71.95	87.49	90.71
0091	63.35	64.89	72.38	76.30
0095	81.90	82.28	77.76	81.54
30.09.2011				
0033	78.59	79.66	88.77	90.52
weighted mean	**74.96**	**75.88**	**84.32**	**87.05**

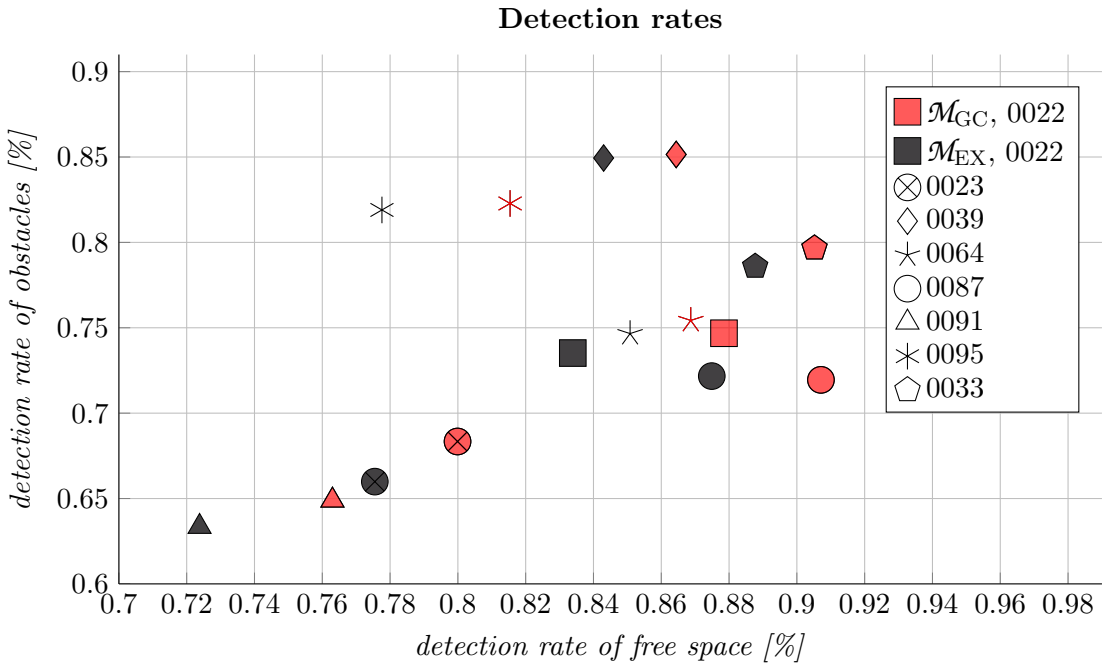


Figure 5.28: Classification accuracies for eight selected KITTI sequences for $\mathcal{M}_{GC}$ (bright red), and $\mathcal{M}_{EX}$ (gray). Different symbols represent different sequences.

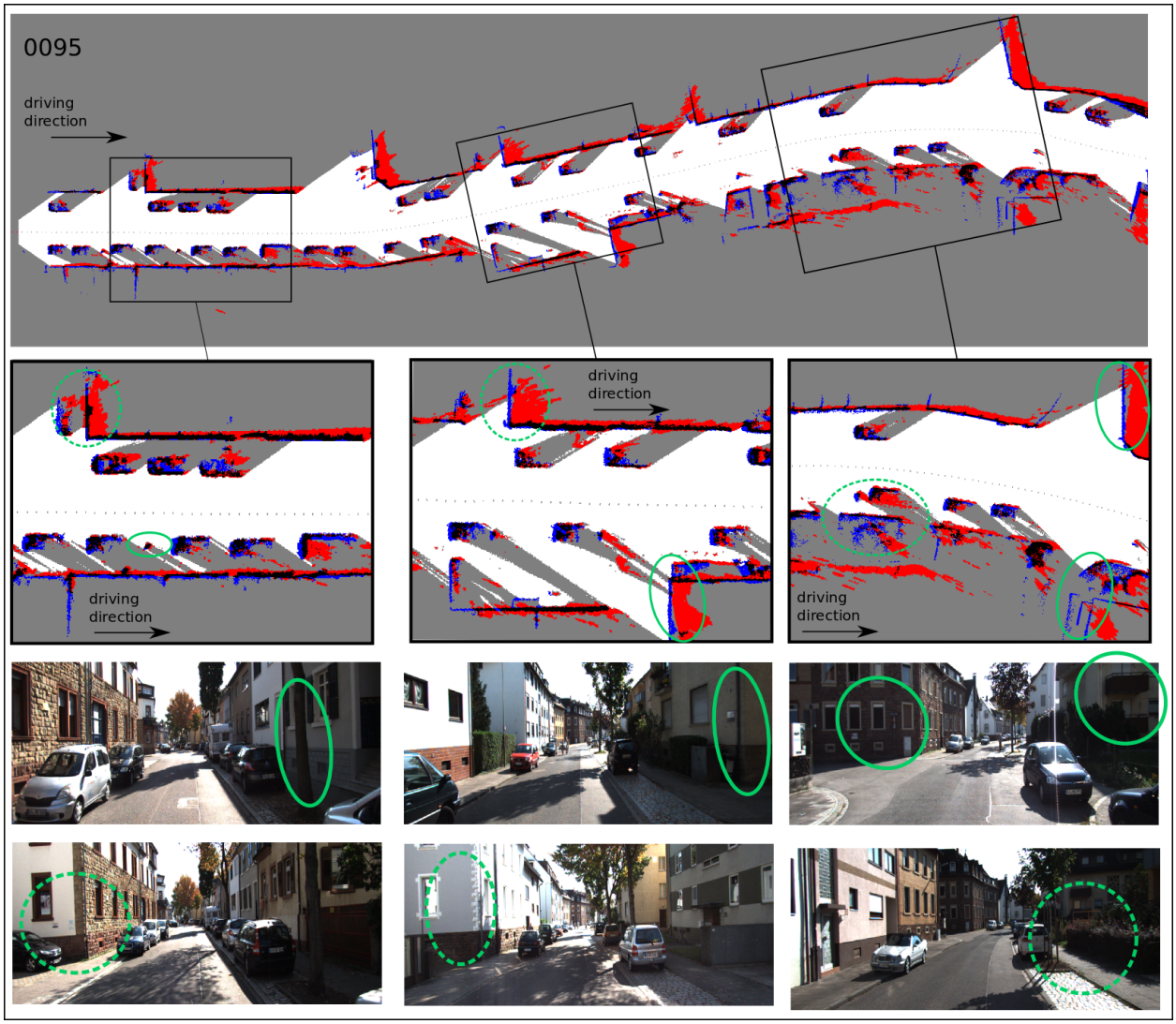


Figure 5.29: The overlay of reference data with $\mathcal{M}_{\mathrm{GC}}$ for parts of sequence 95. Three samples are illustrated which visualize over-segmentation (red areas) and under-segmentation (blue areas). Interesting regions are marked with green circles. In most of the areas a strong overlap is visible and even small objects like poles are matched correctly (left sample). The figure also shows strong over-segmentation in some regions.

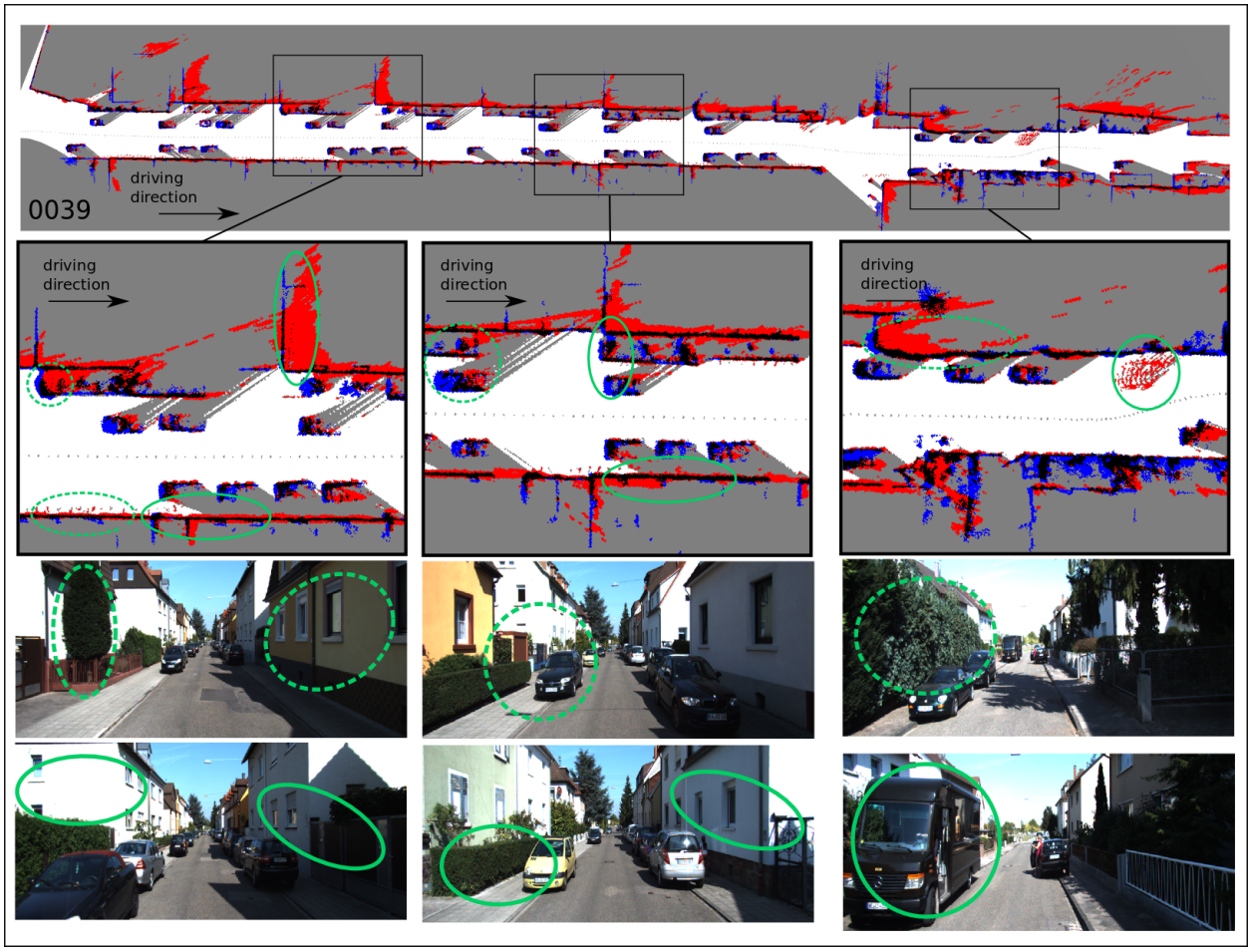


Figure 5.30: The overlay of reference data with $\mathcal{M}_{\mathrm{GC}}$ for parts of sequence 39. Facades parallel to driving direction of the vehicle are aligned well. Even small walls and singe bushes close to the vehicle are well aligned. Difficulties arrive if homogeneous, unstructured images are recorded (left example). This leads to a strong uncertainty and over-segmentation. Artifacts in free space occur if dynamic obstacles are not classified (right example).

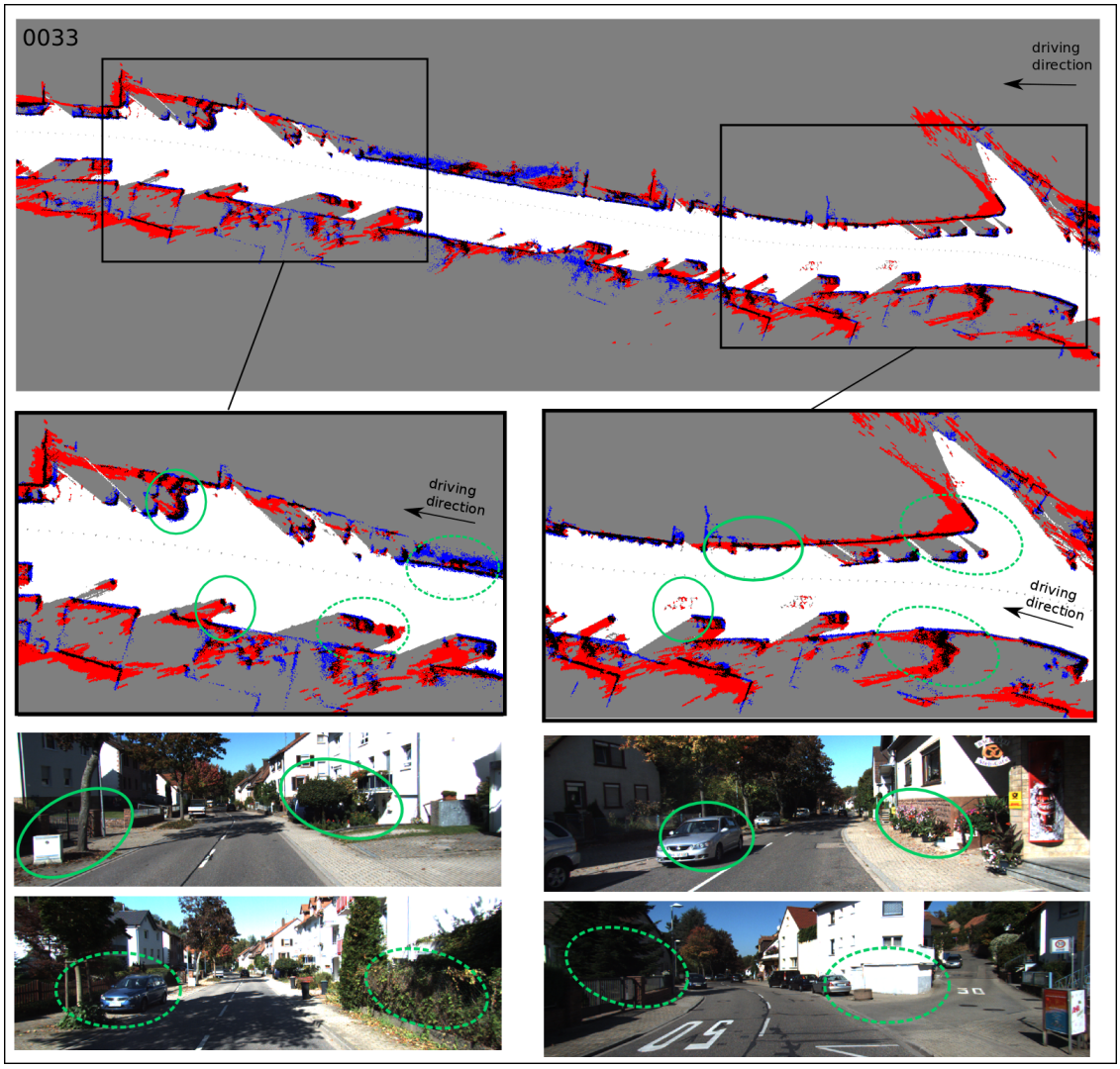


Figure 5.31: The overlay of reference data with map data of $\mathcal{M}_{\mathrm{GC}}$ for parts of sequence 33. In most of the cases a good alignment between reference data and grid maps based on MRFs are visible. Over-segmentation is caused by poor image conditions as well as by the uncertain behavior of stereo vision with regard to increasing distances.

The algorithm reaches its limits in situations like these, because the Stixel definition itself is no longer valid. In contrast to this scenario, sequence 0039 includes well-structured environment in residential areas which fits better to the Stixel World definition. Consequently, both mapping algorithms perform considerably better in these environments.

in Figs. 5.29-5.31, we observe much variation in the classification results for static obstacles: on the one hand, results of the novel approach are well aligned with the reference data. Especially parked cars, house facades parallel to driving direction, and poles produce a high overlap. On the other hand, moderate to strong over-segmentation is truly visible in the examples. In contrast to

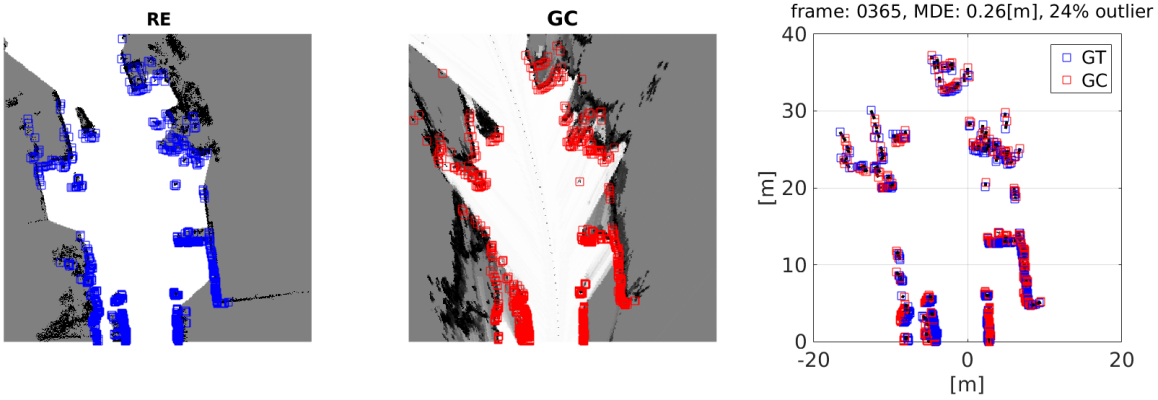
the results in Sec. 5.1.2.2, we have to deal with noisy and partly bad conditioned input data. Thus, the precision of Stixels is less accurate and the outlier probabilities are higher, in general. This behavior influences in turn the definition of the measurement model which finally results in more extended occupied areas. Reasons for noisy and partly bad conditioned data are e.g. difficult light conditions, homogeneous and unstructured areas, and observed objects far away with regard to the ego position. We already mentioned these facts in Sec. 5.2.2.3. We also observe that Stixels of the second row are more inaccurate than Stixels of the first row. These Stixels produce more clutter in the occupancy grid maps. The only meaningful reason is, that these Stixels mostly represent obstacles far away or vegetation, like bushes or trees. Huge under-segmentation is also determined. This means that the algorithm is not able to observe obstacles at all, or the measured distances of the Stixels are not accurate enough with regard to the reference data. An example of under-segmentation is presented in the left example of Fig. 5.30, where the algorithm is not able to map the bushes in front of houses. In Fig. 5.29 we can clearly see that house facades across the driving direction are more far away than reference data provides. We postulate that uncertain behavior of stereo-vision and bad image conditions are responsible for these situations. The non-detection of static obstacles as well as the misalignment leads to the fact, that the absolute detection rates of obstacles are considerably lower than the detection rates in Sec. 5.1.2.2.

5.2.4 Geometrical Accuracies

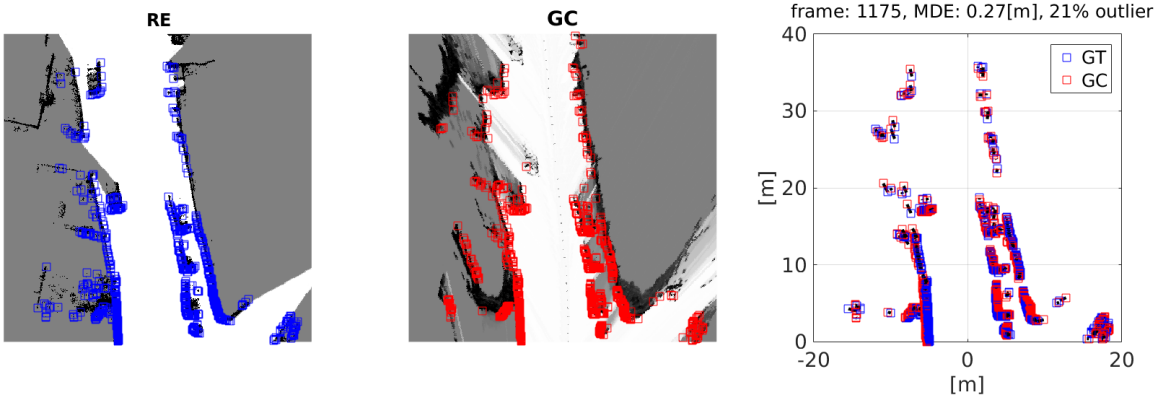
The geometrical accuracy of the estimated grid maps based on real-world data is evaluated in this section. Here, we apply the experimental setup which was already introduced in Sec. 5.1.3.

5.2.4.1 Results of Geometrical Map Errors

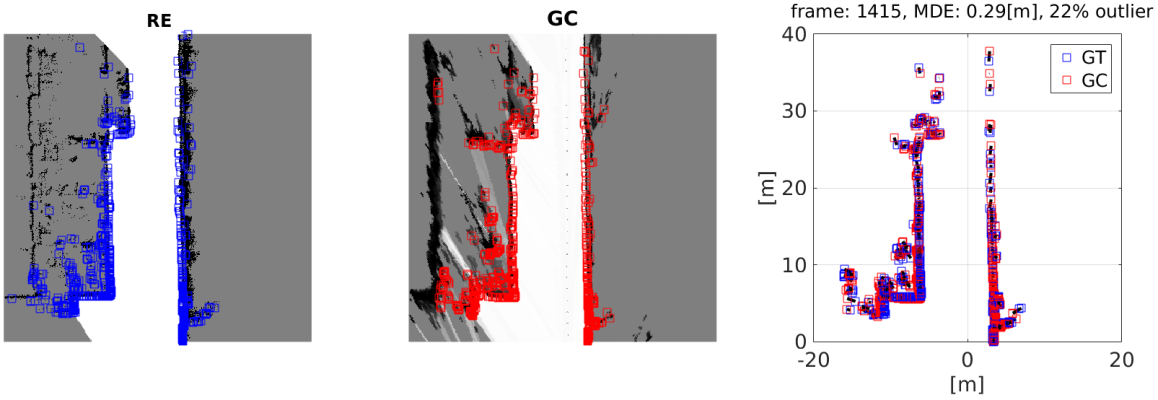
The estimation of geometrical errors was already described in Sec. 5.1.3.1. We simulate a laser scanner which scans the maps $\mathcal{M}_{\mathrm{RE}}$, $\mathcal{M}_{\mathrm{GC}}$, and $\mathcal{M}_{\mathrm{EX}}$ along the driven path. This procedure results in distances d^i_{GC}, d^i_{EX}, and d^i_{GT} for each single detected point i. Based on (5.1) we estimate the geometrical errors Δg^i_m, where $m \in \{\mathrm{GC}, \mathrm{EX}\}$ represents the chosen method. All geometrical errors are concatenated into the error vector $\Delta \boldsymbol{g}_m$ (see Sec. 5.1.3.2). In Sec. 5.2.3 we observed that the noise level of the estimated grid maps are higher than in the evaluation section with artificial sequences. Therefore, we increase the maximum radius for the correspondence search to 1.2 m. The Figs. 5.32-5.33 show the detected points and the correspondences for six sample scans of sequence 0033. These figures also include the mean distance errors (MDEs) for the specific frame as well as outlier rates. To estimate the weights $w^i_{\Delta g_m}$ we follow the procedure described in Sec. 5.1.3.3. The results of the sigmoid slopes $\lambda_{d^i_m}$ which are needed for the weight estimation, are presented in Fig. 5.34. Similar to Sec. 5.1.3.4, the MAEs and WMAEs are estimated for the chosen KITTI sequences and for both methods $\mathcal{M}_{\mathrm{GC}}$ and $\mathcal{M}_{\mathrm{EX}}$. The results are presented in Tab. 5.5. In comparison to Tab. 5.2, Tab. 5.5 also includes the outlier rates. We also mark the lowest and highest (W)MAEs in this table. The distributions of the absolute errors $\mid \Delta g^i_m \mid$ are shown in Fig. 5.35 for six KITTI sequences. Take into account that the number of bins and their widths vary, since the sequences have different length. In Fig. 5.36 we present the distribution over all absolute errors.



(a) sequence 0033, frame 0365.

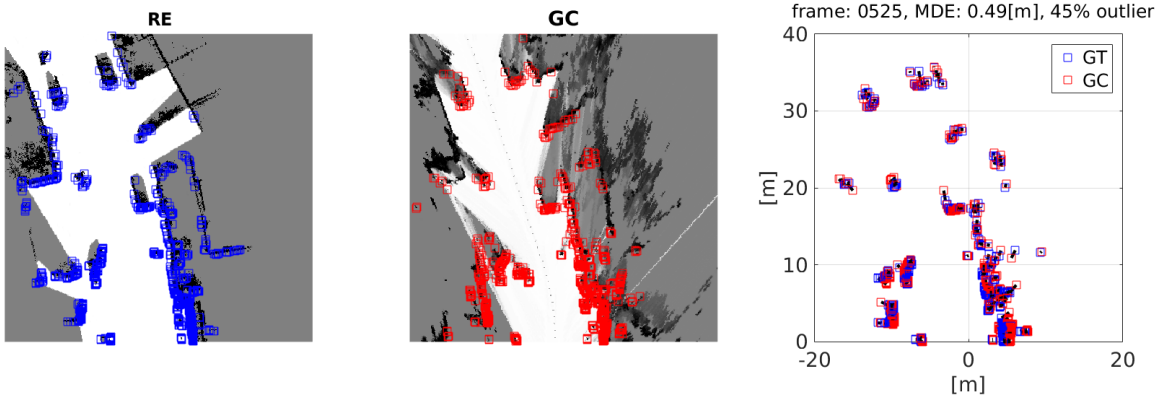


(b) sequence 0033, frame 1175.

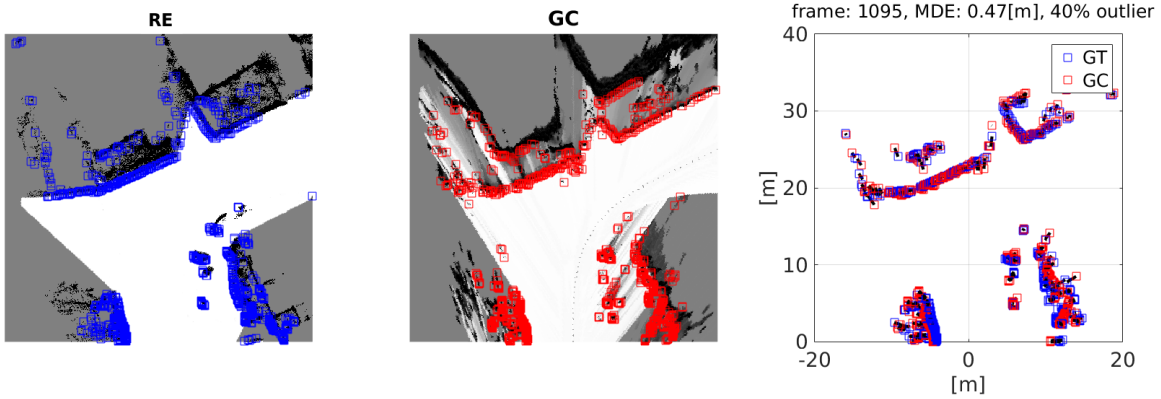


(c) sequence 0033, frame 1415.

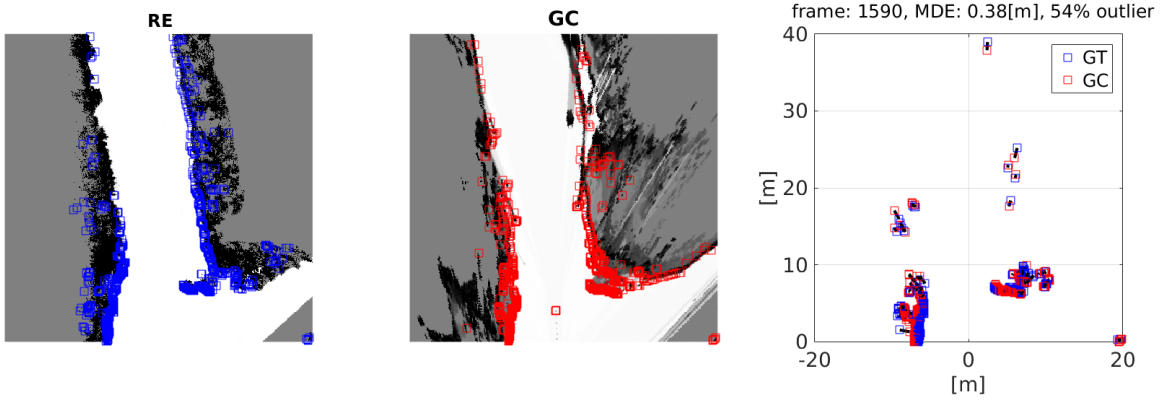
Figure 5.32: Detected points for $\mathcal{M}_{\text{RE}}$ and $\mathcal{M}_{\text{GC}}$ and their correspondences for frames 0365, 1175, and 1415 of sequence 0033. The first column includes the detected points of $\mathcal{M}_{\text{RE}}$, the second column the detected points of $\mathcal{M}_{\text{GC}}$, and the resulting correspondences are visualized in the last column. The visualization of the correspondences also includes the mean distance error (MDE), and the outlier rate for this specific frame number.



(a) sequence 0033, frame 0525.



(b) sequence 0033, frame 1095.



(c) sequence 0033, frame 1590.

Figure 5.33: Detected points for $\mathcal{M}_{\text{RE}}$ and $\mathcal{M}_{\text{GC}}$ and their correspondences for frames 0525, 1095, and 1590 of sequence 0033. The first column includes the detected point of $\mathcal{M}_{\text{RE}}$, the second column the detected points of $\mathcal{M}_{\text{GC}}$, and the resulting correspondences are visualized in the last column. The visualization of the correspondences also includes the distance error (MDE), and the outlier rate for this specific frame number.

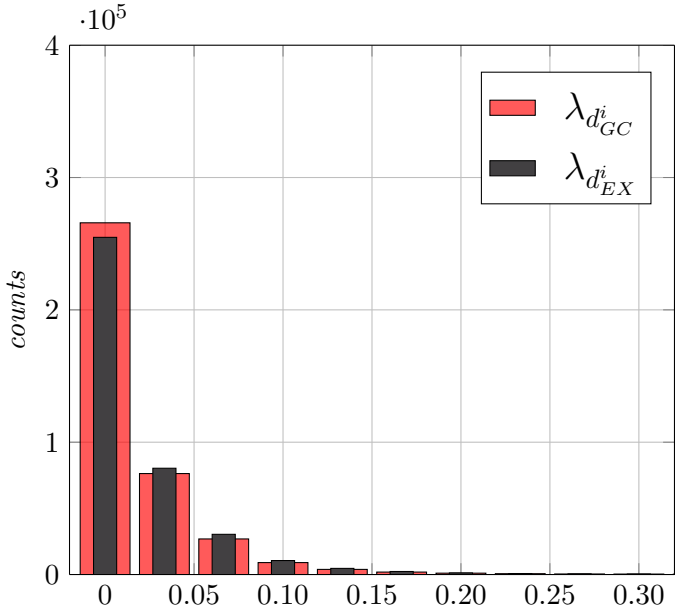


Figure 5.34: Distributions of the slope values $\lambda_{d^i_{\mathrm{GC}}}$ and $\lambda_{d^i_{\mathrm{EX}}}$ for both methods $\mathcal{M}_{\mathrm{GC}}$ (bright red) and $\mathcal{M}_{\mathrm{EX}}$ (gray). In this geometrical evaluation, almost 400 000 data points are used.

Table 5.5: The MAE_m and the WMAE_m of both methods $\mathcal{M}_{\mathrm{GC}}$ and $\mathcal{M}_{\mathrm{EX}}$ for the selected KITTI sequences. The table has the same structure than Tab. 5.2. The outlier rates are also presented.

	$WMAE_m$ [meter]/(outlier rate [%])		*MAE_m [meter]/(outlier rate [%])*	
	$m = GC$	$m = EX$	$m = GC$	$m = EX$
26.09.2011				
0022	0.37/(0.37)	0.38/(0.35)	0.37/(0.37)	0.38/(0.35)
0023	0.40/(0.32)	0.38/(0.31)	0.32/(0.32)	0.31/(0.31)
0039	0.37/(0.24)	0.31/(0.23)	0.28/(0.24)	0.27/(0.23)
0064	0.40/(0.29)	0.39/(0.28)	0.40/(0.29)	0.40/(0.28)
0087	0.43/(0.36)	0.43/(0.34)	0.46/(0.36)	0.44/(0.34)
0091	0.42/(0.35)	0.39/(0.34)	0.41/(0.35)	0.40/(0.34)
0095	0.30/(0.26)	0.29/(0.25)	0.30/(0.26)	0.29/(0.25)
30.09.2011				
0033	0.36/(0.31)	0.37/(0.31)	0.35/(0.31)	0.36/(0.31)
over all	**0.38**/(0.31)	**0.37**/(0.30)	**0.36**/(0.31)	**0.36**/(0.30)

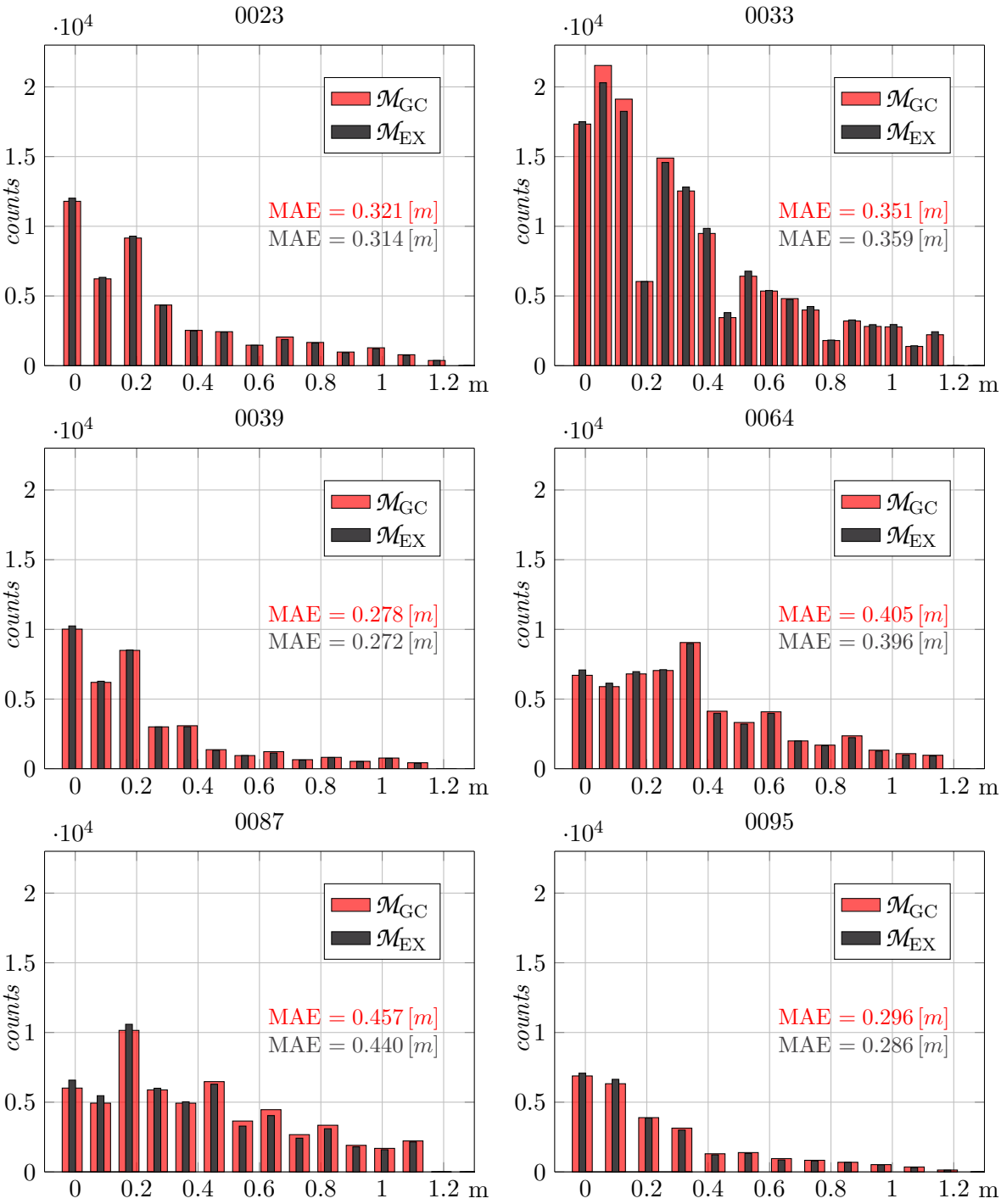


Figure 5.35: Histograms of the absolute geometrical errors Δg_m^i for the KITTI sequences 0023, 0033, 0039, 0064, 0087, and 0091. The distributions of method $\mathcal{M}_{\text{GC}}$ are visualized in bright red, and the distributions of $\mathcal{M}_{\text{EX}}$ are visualized in bright gray. The histograms also include the mean absolute errors (MAE_m).

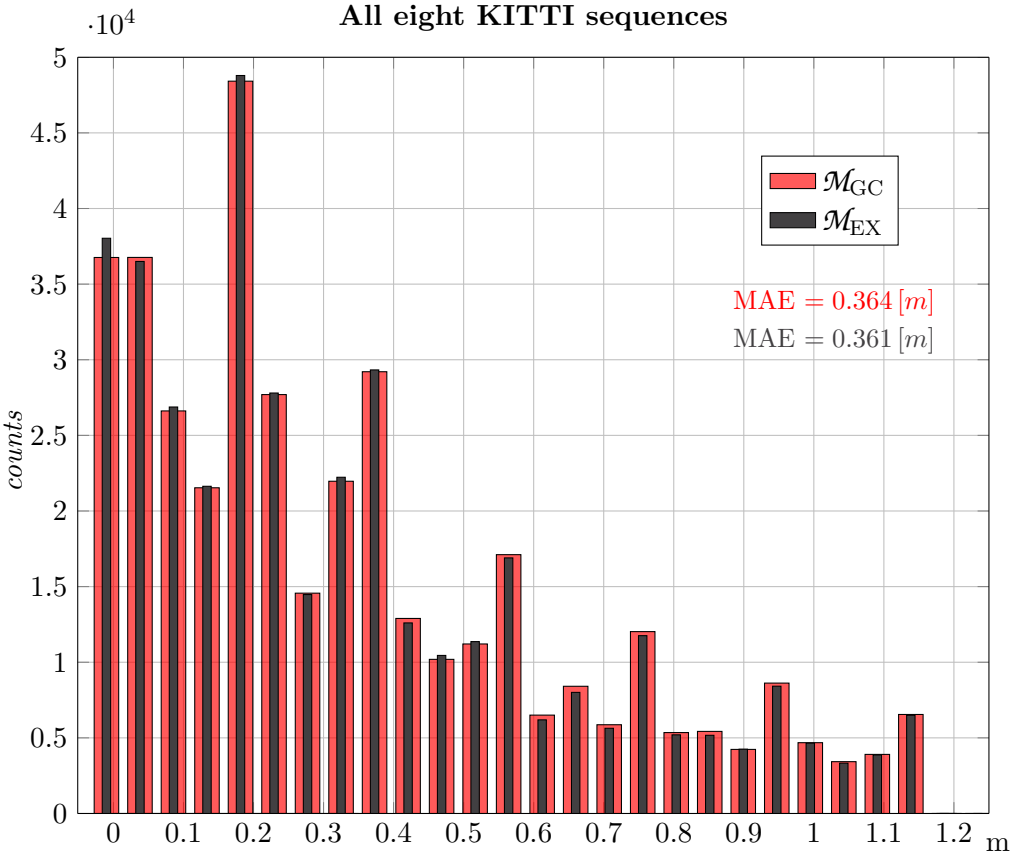


Figure 5.36: Histogram of all estimated absolute geometrical errors $|\ \Delta g_m^i\ |$ for both methods $\mathcal{M}_{GC}$ (bright red) and $\mathcal{M}_{EX}$ (gray). Almost 400 000 measurements are used.

5.2.4.2 Discussion

Taken the results of Tab. 5.5 into account we observe that the novel mapping approach as well as the approach $\mathcal{M}_{EX}$ have the same geometrical map accuracy with a MAE of 0.36 m. Consequently, and similar to the results in Sec. 5.1.3.5, we cannot observe a benefit of the novel mapping approach as long as we take the MAE into account. The histograms in the Figs. 5.35-5.36 substantiate this statement: we cannot observe considerable differences between the distributions of the absolute errors for both methods.

Tab. 5.5 also reveals that the MAE differs strongly between the different sequences. In sequence 0039 both methods perform best with $\text{MAE}_{GC} = 0.28\,\text{m}$ and $\text{MAE}_{EX} = 0.27\,\text{m}$. In comparison, sequence 0087 has a MAEs of 0.46 m ($\mathcal{M}_{GC}$) and 0.44 m ($\mathcal{M}_{EX}$). The differences between both sequences are over 15 cm. As already figured out in Sec. 5.2.3, the accuracy of the occupancy grid maps strongly depends on the captured environment and its distance with regard to the ego vehicle. In sequence 0087 mainly unstructured environment like bushes and vegetation were captured. The highest errors occur in this sequence. In comparison, residential areas with narrow streets and well

structured environment were recorded in sequence 0039 and 0095 instead. These sequences have the lowest errors. We also have to deal with outlier rates up to 37 % which is observed in sequence 0022. The lowest outlier rate occurs in sequence 0039 with 23 %. These rates are understandable since we observed in Sec. 5.2.3 that in some estimated map regions obstacles are not aligned with the reference data or that they are even missing in the estimated maps. This results in high outlier rates since no correspondences can be estimated in these regions.

The estimation of the WMAEs takes the weights $w^i_{\Delta g_m}$ of the errors $\mid \Delta g^i_m \mid$ into account. As one can see in Tab. 5.5, the WMAEs also vary strongly between the sequences. The highest error occurs in sequence 0087, the lowest WMAE is observed in sequence 0095. Taking the WMAEs over all sequences into account, the method $\mathcal{M}_{\mathrm{EX}}$ performs slightly better than the novel approach $\mathcal{M}_{\mathrm{GC}}$. The errors are $\mathrm{WMAE}_{\mathrm{GC}} = 0.38\,\mathrm{m}$ and $\mathrm{WMAE}_{\mathrm{EX}} = 0.37\,\mathrm{m}$. In Fig. 5.34 we observe that the distribution of estimated slopes $\lambda_{d^i_m}$ for both methods $\mathcal{M}_{\mathrm{GC}}$ and $\mathcal{M}_{\mathrm{EX}}$ are very similar. This means, the precision of the detected points is also nearly the same. This insight conflicts with the qualitative statements in Sec. 5.2.2.3 where we postulate that occupied areas are sharper and more precise when we use the novel mapping approach. Taking the presented results into account, this statement is no longer valid. We observe that the presented geometrical errors do not differ considerably between both mapping methods. This also means, that occupied areas have closely the same precision. We regret that we are not able to prove with the used experiments that the novel mapping approach achieves improvements with regard to geometrical accuracies in real word situations.

5.2.5 Summary and Final Discussion Using Real-World Data

5.2.5.1 Summary

In this section we evaluated the novel mapping approach based on real-world data using the raw data sets of the KITTI vision benchmark suite [Geiger et al., 2012, 2013]. Based on precise pose information of a IMU/GPS unit and Velodyne point clouds reference occupancy grid maps $\mathcal{M}_{\mathrm{RE}}$ were generated in a semi-automated way. The provided stereo image sequences were used to feed the novel mapping approach. Qualitative results were presented to allow first insights how good the novel mapping approach performs against reference data. Furthermore, we compare the novel approach against the mapping technique of Muffert et al. [2014] which does not take the dependency of neighboring grid cells into account .

We presented classification accuracies and the geometrical map errors in Sec. 5.2.3 and Sec. 5.2.4. Here, we used the experimental setups of Sec. 5.1.2 and Sec. 5.1.3. We observed that the novel mapping approach has strong overlaps with the reference data as long as obstacles are close to the ego vehicle and well structured environment was captured in the recordings. We also figured out that the algorithm reaches its limits in chaotic and unstructured environments. Especially vegetation which is observed in the outer areas of the field of view of the stereo camera produces a lot of clutter in the final grid maps.

Detection rates up to 90 % for free space, and detection rates in the range of 85 % for obstacles were achieved using the novel mapping approach. The results of the classification accuracies show that the use of MRFs during occupancy grid mapping produces a smoother and cleaner free space as well as slightly better detection rates of obstacles. We state that these insights are one of the

core achievements of this thesis. Unfortunately we were not able to show in a quantitative way, that occupancy grid maps based the novel mapping approach achieve higher geometrical accuracies than approaches which do not take the dependencies of neighboring grid cells into account. For both methods, the MAE was 0.36 m.

5.2.5.2 Final Discussion

At the end of this chapter we have a final discussion about the obtained insights from the implemented experiments based on real-world image sequences.

Influences of the defined Measurement Model. The results of the occupancy grid maps strongly depends on the quality of the input data and how their uncertain behavior is modeled in the measurement model $p(\mathcal{S}_t \mid m_{i,t}, {}^w\mathcal{X}_{0:t})$. In our case the measurement model is primary controlled by the Stixel uncertainty $\tilde{\sigma}^2_{d_{un}}$, the outlier probability p^{out}_{un}, the disparity interval d_s, and the Stixel width s_w. It depends on the captured situation and environment if the combination of these factors together results in a well conditioned measurement model or not.

In our evaluation steps we figured out that the measurement model is well conditioned when we observe well structured environments which fits to the assumptions made during the Stixel estimation. Here, the major assumption is that the captured surroundings are mainly controlled by man-made environment with either vertical or horizontal planar surfaces (see Sec. 2.2.3). As a result of this, house facades, well-aligned walls, parked cars and even small poles are precisely represented by the Stixels. Consequently, the measurement model is good conditioned which results in a precise global occupancy grid map and a high overlap with reference data at the end. On the other hand, the Stixel algorithm has its difficulties by representing non man-made environments. This results in a bad conditioned measurement model, and consequently in a global grid map with noise and clutter. This was observed in Sec. 5.2.3. As long as we use the Stixels as input for our measurement model we have to cope with these situations.

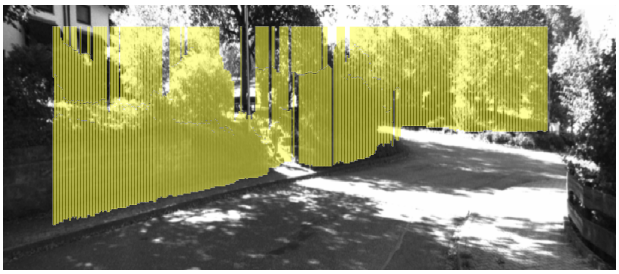
Since we define the measurement model in the disparity space we have to handle the projection into the regular Cartesian grid space (see Sec. 3.4.2.2). There, the most important factor is the disparity interval step d_s which defines the granularity of the discretization of the disparity space. The larger the value d_s, the smaller is the influence of projection and discretization errors. This was already observed and discussed in Sec. 5.1.4 in detail. The definition of the measurement model in the disparity space and taking the disparity uncertainties of Stixels $\tilde{\sigma}^2_{d_{un}}$ into account leads to the fact that the farther away obstacles are observed, the more Cartesian grid cells are influenced and occupied. Only if the distance to these obstacles is reduced over time, the contours get precise and crisper. This uncertain behavior is typical for stereo vision and was observed in both evaluation sections. Nevertheless, we do not have a quantitative proof that the geometrical error of obstacles depends on the observed distance. This should be done in future.

These three essential characteristics of the measurement model can also be observed in Fig. 5.37. The figure shows the original scene, the used Stixels, and the resulting measurement model $p(\mathcal{S}_t \mid m_{i,t}, {}^w\mathcal{X}_{0:t})$. Walls close to the ego vehicle are precisely mapped, whereas trees and bushes produce cluttered areas. One can also see that the farther away obstacles are observed, the more uncertain they become.

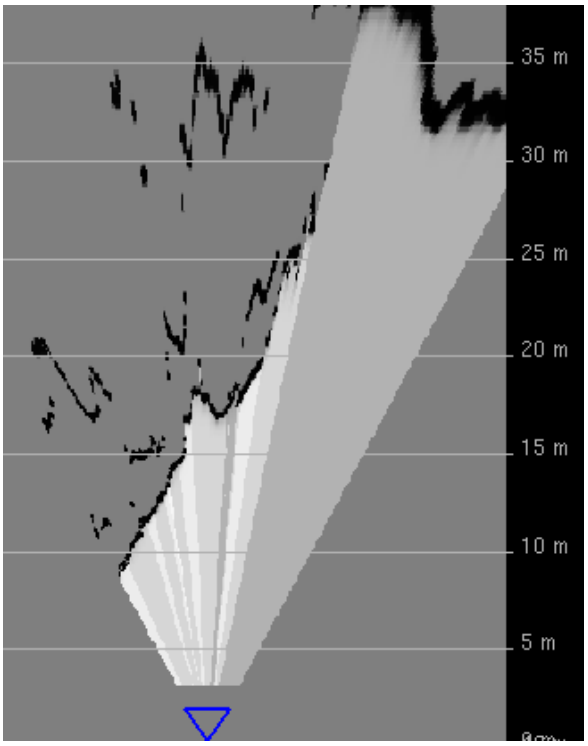
As stated in Sec. 5.2.3, the horizontal angular resolution influences the free space detection. To achieve a better horizontal angular resolution we suggest to increase the ratio of pixels per degree. A reduction of the Stixel width with $s_w < 3\,\text{pel}$ makes little sense for us.

As long as the current Stixel approach is used, we reach limits by the representation of obstacles with small heights. Especially curbs should be mentioned here which are neglected in the current mapping approach. The consideration of these obstacles is one of the most important steps we should carry out in future. Another important step of improvement would be the differentiation of temporary static and long-term static obstacles. A good example are parked cars which are not excluded at this time. To improve the quality of maps, they should exclude in future.

(a) Original image.



(b) The used Stixels.



(c) The results of the measurement model.

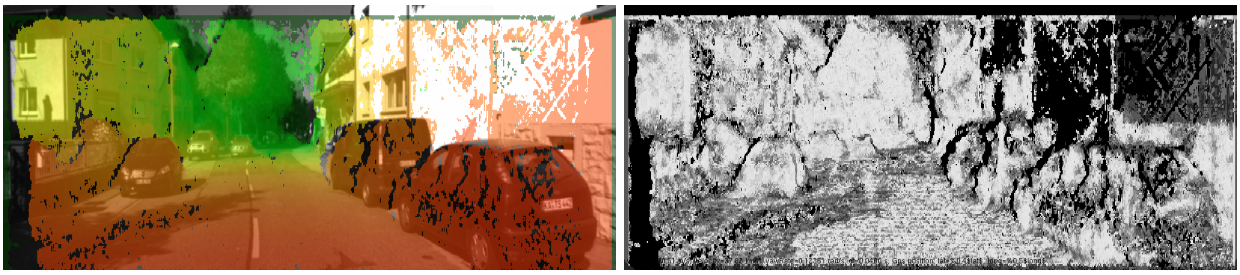
Figure 5.37: Example result of the measurement model for a complex scene (KITTI sequence 0033, image number 954). The essential characteristics of the measurement model can be observed. Details are described in the text above.

Influence of the Prediction Step. The prediction term $p(m_{i,t} \mid \mathcal{S}^{lab}_{0:t-1}, {}^{w}\mathcal{X}_{0:t})$ represents the conditional probability of a single grid cell without taken current measurements $\mathcal{S}^{lab}_{t}$ into account. It is controlled by the transition probabilities $p(m_{i,t} \mid m_{i,t-1})$ and $p(\neg m_{i,t} \mid m_{i,t-1})$ which were defined by 0.95 and 0.05, respectively. The influence of these parameters was not evaluated in this thesis and should be done in future.

Influence of the Use of MRFs during Occupancy Grid Mapping. The evaluation based on real-world data shows that the use of MRFs during occupancy grid mapping increases especially the detection rate of free space and slightly the detection rate of obstacles. With the above described experiments we were not able to prove in a qualitative way, that the novel approach also achieve better geometrical accuracies with regard to static obstacles. This statement is contradictory to the qualitative validation of the two map methods and their comparison (see Sec. 5.2.2.3). Based on visual inspection we stated that the maps of the novel approach have sharper and more precise occupied areas than the results of using method $\mathcal{M}_{\text{EX}}$. In future we should improve our experiments to prove these insights based on pure eye balling.

The use of MRFs includes the use of the binary terms $\Phi(m_{i,t}, m_{j,t})$ which was defined in Sec. 3.5. The binary terms are controlled by two parameters which were defined with $k_{ij} = 0.08$ and $\lambda_b = 2$ in Sec. 5.1.1.3. The influence of these tuning factors was not evaluated yet and should be done in future. The binary terms are realized by a data independent Potts model. In future we should also incorporate observations into this term to allow additional smoothness constraints.

Influence of the Quality of Stereo Vision. The quality of stereo vision influences the Stixel results. Using real-world data, stereo estimation has to handle difficult weather and light conditions which can result in e.g. in the images. These effects were neglected in the evaluation with rendered image sequences (see Sec. 5.1.1) to provide best conditioned scenarios. Reflections or overexpose can lead to less accurate disparity images, and consequently to Stixels with low precision and higher outlier rates. An example of a less accurate disparity image is shown in Fig. 5.38. It shows a strong overexposure in the upper right part of the image which results in bad or no stereo information and low stereo confidence values. Images with a relatively low image bit depth are more sensitive to these effects. In this evaluation the KITTI images have a bit depth of 8.



(a) Disparity image. (b) Confidence image.

Figure 5.38: Example of overexposure and the resulting disparity image (left image) with its stereo confidences (right image). Strong overexposure is visible in the upper right part of the left image which results in bad or no stereo information and low stereo confidence values. The color encoding is described in Sec. 2.2.2.

Influence of Errors in the Reference Map. The generation of the reference occupancy grid maps rely on a semi-automated technique where an inspection by hand is carried out in the final step. This inspection includes smoothing free space areas as well as removing dynamic obstacles,

and occupied areas which are defined as clutter by the user. This inspection by hand can lead to really minor errors in the reference maps which is shown in Fig. 5.39. As long as these errors are really rare, they influence the detection rates and geometrical errors only in a very moderate way, but they should be avoided in future.

Figure 5.39: Detected error in the reference map. The green circles show that a small bush is missing in the reference data (left), but is clearly mapped using our new mapping approach (middle).

Runtime Behavior. The current implementation of the novel mapping approach needs about 180 ms per image cycle using only one kernel of the CPU. Here, the image preprocessing steps were excluded. 85 % of the runtime are used for the computation of the marginal probabilities using dynamic graph cuts. Since we use a foreign, open source implementation for this task, we consider a runtime optimization as challenging for this part. The use of the GPU for the global grid map update is effective if not to many downloads/uploads between GPU and CPU are carried out within one cycle. The realization of the measurement model can be parallelized using OpenMP[2] in future. In conclusion, we consider it as difficult to improve the current implementation in a manner that the approach runs in a range of 10 Hz or even faster.

[2]http://www.openmp.org/

Chapter 6

Evaluation with Uncertain Poses

In this chapter the introduced SLAM approach is evaluated based on real-world data. In Sec. 6.1 the used data set, as well as the parameter settings of the SLAM approach based on RBPFs is presented. Afterwards, it is explained how we generate reference data, and how the experiments look like (see Sec. 6.2). In Sec. 6.3 and Sec. 6.4 the results are presented and discussed. We finalize this chapter with a brief summary and a conclusion.

6.1 Data Acquisition and Parameter Definition

This section includes the description of data acquisition (Sec. 6.1.1), and the definition of the parameters which are used during the RBPF approach (Sec. 6.1.2). We also explain the experiments in Sec. 6.2.3.

6.1.1 Data Acquisition

In this evaluation we use real-world image sequences which were recorded with the test vehicle *S 500 Intelligent Drive* [Ziegler et al., 2014]. An image of the research car was already presented in Fig. 3.3(b) of Sec. 3.2.1. The used stereo camera rig is shown in Fig. 3.3(a). The stereo system has a baseline of 0.23 m and a horizontal field of view of ≈ 44 deg. The cameras are mounted 1.25 m above ground behind the windshield of the test vehicle. The 12 bit grayscale images have a size of 1024(W) $\times$ 440(H) pel, and the system runs with a frame rate of 25 Hz. Just like in Sec. 5.2.2.1, we chose a Stixel width of $s_w = 3$ pel and a disparity resolution of $d_s = 16$. With this setup a horizontal angular resolution of 0.13 deg per Stixel is achieved. Compared to the raw data provided by the KITTI Vision Benchmark Suite [Geiger et al., 2012, 2013], the presented stereo setup of the research vehicle provides a higher angular resolution and a better image quality which results in an improvement of the disparity and Stixel quality. This is our major reason why we use the data of the research vehicle. Furthermore, a comprehensive evaluation of the motion behavior of the research car was already realized in [Dömötör, 2014]. We discussed these results already in Sec. 4.3 and we want to benefit from these insights. Thus, it allows us to model the noise of the motion model precisely. The drawback of using the recorded image sequences of the Mercedes-Benz

research car is, that no GT or reference data is directly available since the system does not include a deeply couple IMU/GPS system to provide these data.

However, we describe in Sec. 6.2 how we can generate reference data only based on the recorded image sequences. As test scenario we chose a residential area in Böblingen, Germany. The area was already shown in Sec. 4.1.1, Fig. 4.1(c). We recorded a sequence with a length of 11 400 images. The total driven path is 1 900 m long and the area covers a region of 200 m × 200 m. In order to detect loop closures later on, we attached importance to record regions multiple times.

6.1.2 Parameter Definition

In this section we define the parameters for the RBPF. For the mapping procedure in the particle filter approach we use the settings which were already defined in Sec. 5.1.1.3. As in Sec. 5.2.2.1, we chose a Stixel width of $s_w = 3\,\text{pel}$ and a disparity resolution of $d_s = 16$. The grid cell resolution is 0.1 m. In the following, we define all important parameters for the particle filter which were already introduced and described in Sec. 4.2. Since each particle i includes its own global grid map $\mathcal{M}^{(i)}$ and consequently produces much memory requirements, we want to keep the number of particles as low as possible. Since we allocate the global maps on the GPU, we are limited by the memory of the used NVIDIA GeForce GTX 480 unit which provides 1.5 GB RAM in total. Therefore, the number of particles is limited by $I = 60$. The estimation of the model parameter $\sigma^2_{\dot{\varphi}_t}$ for the yaw rate $\dot{\varphi}_t$ of the motion model was already mentioned in Sec. 4.3. Figure 4.4 shown the distribution of the observed yaw rate with and without offset correction. The studies result in a systematic yaw rate offset of $1.3 \times 10^{-3} \frac{\text{rad}}{s}$ and an estimated yaw rate precision of $\sigma_{\dot{\varphi}_t} = 0.008\ \frac{\text{rad}}{\text{s}}$. The standard deviation of the velocity v is tuned by hand and is defined by $\sigma_{v_t} = 0.15\ \frac{\text{m}}{\text{s}}$. The sampling rate is running with 25 Hz and 0.04 sec respectively. As mentioned in Sec. 4.4, the weight estimation runs with a lower frequency then the pose sampling. Here, we choose a frequency of 1.6 Hz which allows an intermediate map integration of 15 observation steps. This provides a larger field of view of the local environment around the ego vehicle for the grid matching process. The tuning parameter α of the observation model $p(\mathcal{S}_t|\mathcal{M}^{(i)}_{t-1}, \boldsymbol{x}^{(i)}_t)$ during grid matching is set by 1.5×10^{-4}. For adaptive resampling scheme, we use the effective number of particles N_{eff}. Only if $N_{\text{eff}} < 0.5$, we carry out the resampling procedure.

6.2 Generation of Reference Data and Experiment Description

As mentioned in the previous section the research vehicle does not include a deeply coupled IMU/GPS system to provide precise GT or reference data. Therefore, we generate the reference data in a semi-automated way based on the recorded image sequence data and the vehicle odometry information. The key idea is to use a full SLAM technique, which was introduced in Sec. 2.6, to generate the reference trajectory $\mathcal{X}_{0:t}$ for the recorded image sequence. The reason why we use these results as reference is explained in the following: as stated in Sec. 2.6, full SLAM approaches take the whole history into account which results in more stable and consistent solutions. To estimate the best probable solution for $\mathcal{X}_{0:t}$, we rely on a graph based solution with feature maps. In this context, we solve the challenging task of loop close detection and outlier reduction by hand. Next, we describe the construction of the graph $G(\mathcal{X}_{0:t}, \mathcal{M})$.

6.2.1 Construction of the Graph

The construction of the graph is illustrated in Fig. 6.1. Following (2.50), we have to define the constraints between consecutive poses using the motion model $g(.,.)$ with its noise behavior Ω^m_{t-1}, and the constraints between poses and landmarks using the observation model $h(.,.)$ with its noise behavior $\Omega^o_{t,i}$. Just like in Sec. 4.3, odometry information of the ego vehicle with the recorded velocities $\boldsymbol{v}_{1:t}$ and the yaw rates $\dot{\boldsymbol{\varphi}}_{1:t}$ is used to feed the motion model $g(.,.)$. The same uncertain behavior as described in Sec. 6.1 is applied to model Ω^m_{t-1}.

To define the constraints between poses and map features we rely on the Stixel World and use static Stixels as map features. Since Stixels are tracked over time (see Sec. 2.2.3), these Stixel tracklets are used for the definition of global map features (see Fig. 6.1). Only static Stixels which are tracked over a period of more than 50 frames are considered in the graph. The noise behavior of a single feature constraint is defined by the theoretical precision of the triangulated Cartesian 2D point of each Stixel using (2.4)-(2.6) of Sec. 2.2.2.3. Since the precision of a triangulated 2D point is decreasing quadratically with regard to the distance (see also Fig. 2.6), we only consider static Stixels up to a distance of 40 m. These Stixel tracklets and their theoretical precision define the observation model $h(.,.)$ with their noise behavior $\Omega^o_{t,i}$. How Stixel tracklets and their precision look like is also visualized in Fig. 6.1, top.

The challenging part of automated loop close detection is carried out by hand. Temporary different static Stixel tracklets which represent the same static feature, are fused together and get the same feature ID. An example is shown in Fig. 6.1, bottom right. As anchoring constraint, the first position of the ego vehicle is chosen to $\boldsymbol{x}_0 = [0\ 0\ 0]^\mathsf{T}$.

6.2.2 Graph Optimization

After the construction of the graph $G(\mathcal{X}_{0:t}, \mathcal{M})$ we want to optimize in a way that we get the best probable results for the pose $\mathcal{X}^*_{0:t}$ and the map $\mathcal{M}^*$, respectively. As already described in Sec. 2.8, the graph $G(\mathcal{X}_{0:t}, \mathcal{M})$ represents a sum of non-linear constraints which are formulated as a NLS problem. Here, we exploit the power of the open source library *g2o* [Kümmerle et al., 2011] to solve the NLS problem. We rely on the Levenberg-Marquardt (LM) procedure which is used in the *g2o* software. 64 iteration steps were carried out to find the optimum solution for $\mathcal{X}^*_{0:t}$ and $\mathcal{M}^*$, respectively. Figure 6.2 shows the initialized, none optimized graph $G(\mathcal{X}_{0:t}, \mathcal{M})$(top) and the optimized graph $G(\mathcal{X}^*_{0:t}, \mathcal{M}^*)$ (bottom). For evaluation we define the reference trajectory with $\mathcal{X}_{0:t,\mathrm{RE}} = \mathcal{X}^*_{0:t}$ and the reference map with $\mathcal{M}_{\mathrm{RE}} = \mathcal{M}^*$.

6.2.3 Experiment Description

The implementation of the RBPF of Sec. 4.2 results in the maps $\mathcal{M}^{(i)}_{\mathrm{GC}}$, the trajectories $\mathcal{X}^{(i)}_{0:t,\mathrm{GC}}$, and the weights $\mathcal{W}^{(i)}_{0:t,\mathrm{GC}}$ for each particle i. Here, GC symbolizes that we use the novel mapping approach based on MRFs and dynamic graph cuts during the RBPF. From now, we defined this method as RBPF_GC. Based on the single trajectories and their corresponding weights we decided to estimate the weighted trajectory over all particles $\widehat{\mathcal{X}}_{0:t,\mathrm{GC}}$ which we finally evaluate against $\mathcal{M}_{\mathrm{RE}}$. The trajectory $\widehat{\mathcal{X}}_{0:t,\mathrm{GC}}$ is also used to (re)estimate the final map $\widehat{\mathcal{M}}_{\mathrm{GC}}$ using "mapping with known poses".

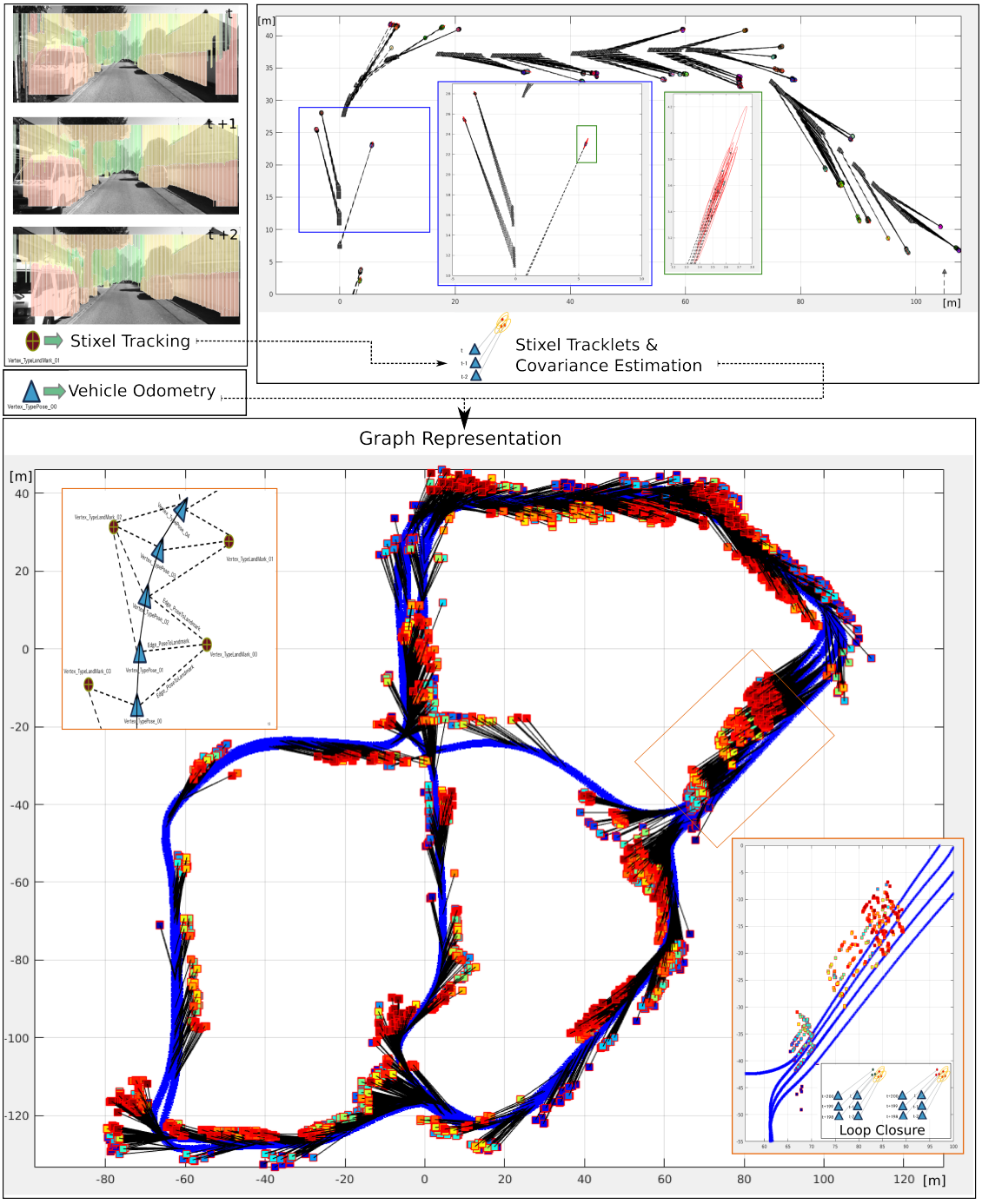


Figure 6.1: The Construction of the Graph $G(\mathcal{X}_{0:t}, \mathcal{M})$ for the recorded image sequences. The Stixel tracking is used to define Stixel tracklets which define static map features. Based on these tracklets and Covariance/Information estimation, the observation constraints are defined in the graph structure (top). The odometry information of the ego vehicle is applied to define the motion constraints. With both constraint definitions the global graph $G(\mathcal{X}_{0:t}, \mathcal{M})$ is constructed (bottom). An example of loop close detection is shown in a close-up bottom right. The blue colored map features as well as the red colored map features represent the same obstacles. These features represent the same obstacle and, therefore, they get the same IDs.

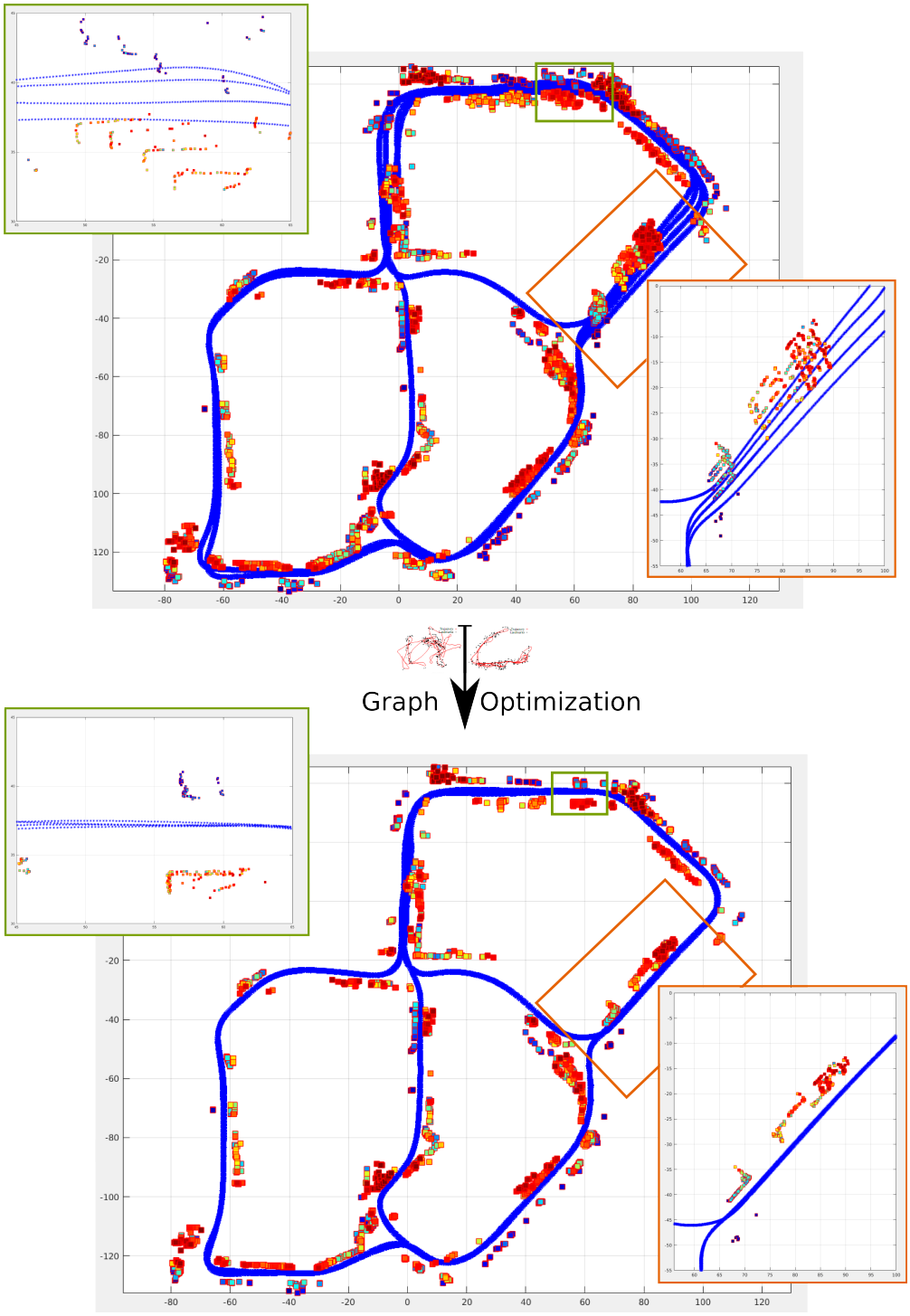


Figure 6.2: The initialized graph $G(\mathcal{X}_{0:t}, \mathcal{M})$(top) and the result $G(\mathcal{X}^*_{0:t}, \mathcal{M}^*)$ (bottom) after optimization. Two close-ups show how well aligned the trajectory as well as features are after the optimization step.

We also run the implementation of the RBPF with the mapping method described in [Muffert et al., 2014] to make a comparison between both mapping techniques. Therefore, the results $\widehat{\mathcal{X}}_{0:t,\mathrm{EX}}$ and $\widehat{\mathcal{M}}_{\mathrm{EX}}$ are also estimated. This method is defined as RBPF_EX. We focus on the evaluation of the trajectories $\widehat{\mathcal{X}}_{0:t,\mathrm{GC}}$ and $\widehat{\mathcal{X}}_{0:t,\mathrm{EX}}$ against the reference trajectory $\mathcal{M}_{\mathrm{RE}}$. We neglect the full evaluation of the maps since comprehensive evaluation assessments with regard to map quality were already carried out in Sec. 5.1 and Sec. 5.2. Nevertheless, we present the map results in Sec. 6.3. Here, we briefly discuss the results in a qualitative way. Since the trajectories are synchronized to each other, we are able to estimate the pose error $\Delta\boldsymbol{x}_{m,t}$ for each time step t and both methods $m \in \{\mathrm{GC}, \mathrm{EX}\}$ directly using the inverse compositional operator $\ominus$ [Geiger et al., 2012]:

$$\Delta\boldsymbol{x}_{m,t} = \widehat{\boldsymbol{x}}_{m,t} \ominus \boldsymbol{x}_{\mathrm{RE},t} \ . \tag{6.1}$$

The pose error includes the components of the lateral $\Delta x_{m,t}$, the longitudinal $\Delta y_{m,t}$, and the heading error $\Delta\phi_{m,t}$ with regard to the ego vehicle coordinate system:

$$\Delta\boldsymbol{x}_{m,t} = [\Delta x_{m,t},\ \Delta y_{m,t},\ \Delta\phi_{m,t}]^{\mathsf{T}}. \tag{6.2}$$

We also estimate the translation error $\Delta t_{m,t} = \sqrt{\Delta x_{m,t}^2 + \Delta y_{m,t}^2}$. The results of the pose errors are presented and discussed in Sec. 6.4.

6.3 Qualitative Evaluation of the Map Results

As stated in Sec. 6.2.3, we only present a brief qualitative evaluation of map results in this section. Figure 6.3 shows four different occupancy maps, namely the reference map $\mathcal{M}_{\mathrm{RE}}$ (Fig. 6.3(a)), the map based on pure odometry (Fig. 6.3(b)), the map $\widehat{\mathcal{M}}_{\mathrm{GC}}$ (Fig. 6.3(c)), and the map $\widehat{\mathcal{M}}_{\mathrm{EX}}$ (Fig. 6.3(d)). All maps were generated based on their corresponding trajectories $\mathcal{X}_{0:t,\mathrm{RE}}$, $\widehat{\mathcal{X}}_{0:t,\mathrm{GC}}$, $\widehat{\mathcal{X}}_{0:t,\mathrm{EX}}$, and pure odometry information respectively. In Sec. 4.1.1 we already motivated that the lack of precise pose information results in ambiguities and inconsistencies in the map. This is clearly seen in Fig. 6.3(b), where only odometry information is used. We observe that the maps $\mathcal{M}_{\mathrm{RE}}$, $\widehat{\mathcal{M}}_{\mathrm{GC}}$, and $\widehat{\mathcal{M}}_{\mathrm{EX}}$ do not have these misalignments. No considerable differences between the reference map $\mathcal{M}_{\mathrm{RE}}$ and the map $\widehat{\mathcal{M}}_{\mathrm{GC}}$ are observable by pure eyeballing. The comparison between $\widehat{\mathcal{M}}_{\mathrm{GC}}$ and $\widehat{\mathcal{M}}_{\mathrm{EX}}$ shows that the free space in $\widehat{\mathcal{M}}_{\mathrm{GC}}$ is much cleaner and smoother. This insight is coherent to the quantitative results of Sec. 5.1 and Sec. 5.2. In the following section the accuracy of the estimated pose is discussed in detail.

6.4 Evaluation of the Pose Accuracy

6.4.1 Results of Pose Errors

The pose errors are estimated using (6.1) of Sec. 6.2.3. In Fig. 6.4 the translation errors $\Delta t_{m,t}$ and the absolute heading errors $|\ \Delta\phi_{m,t}\ |$ are presented with regard to their global positions $\widehat{\boldsymbol{x}}_{m,t}$. The pose errors are color encoded where red symbolizes large errors, and green stands for small

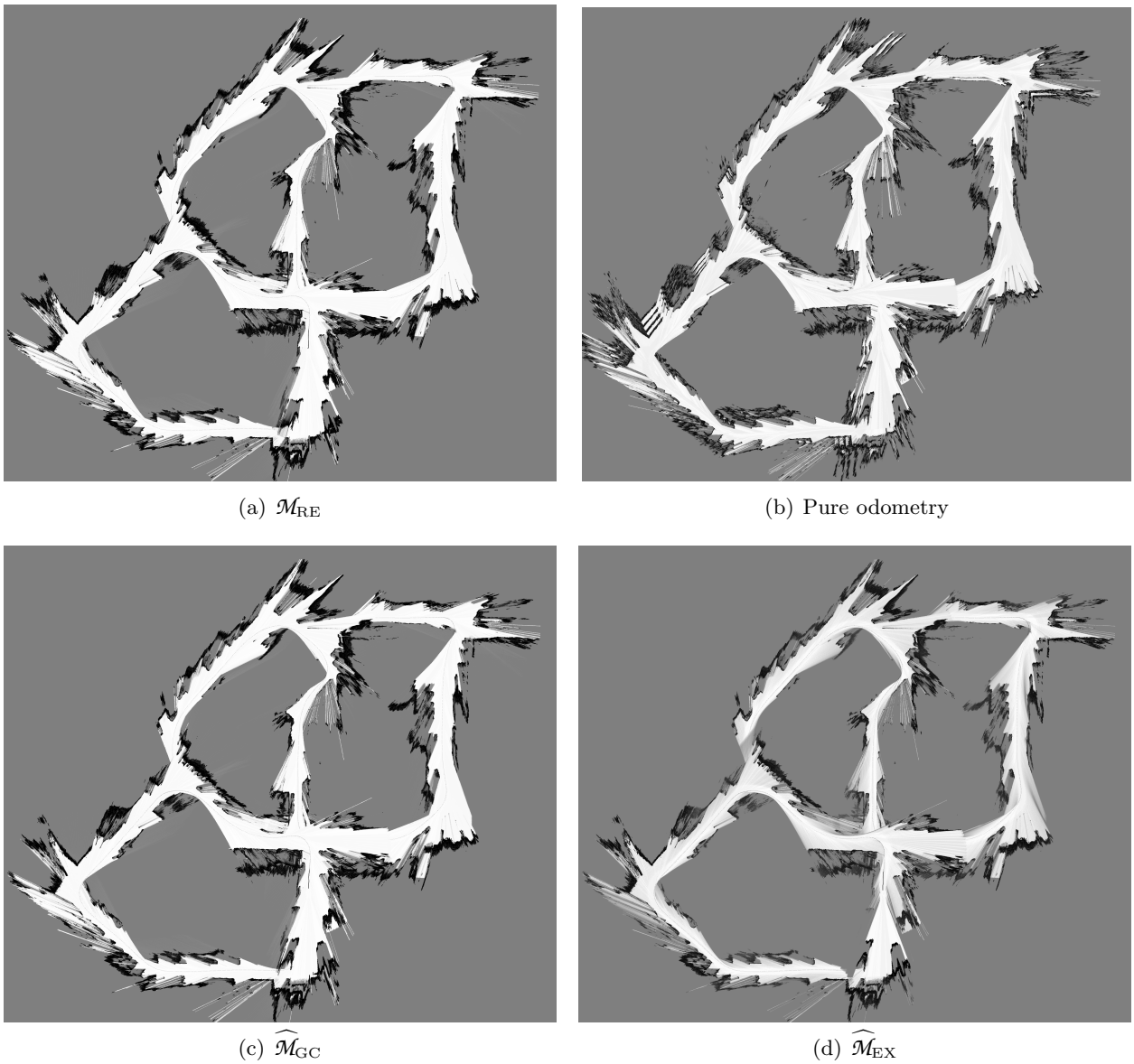
errors. Figure 6.4 includes the results of both methods RBPF_GC and RBPF_EX. Next to these heat maps, the distributions of the absolute lateral errors $\mid \Delta x_{m,t} \mid$, the absolute longitudinal errors $\mid \Delta y_{m,t} \mid$, and the absolute heading errors $\mid \Delta \phi_{m,t} \mid$ are presented in Fig. 6.5. These histograms also include the MAEs for RBPF_GC as well as for RBPF_EX.

6.4.2 Discussion

As clearly shown in Fig. 6.4(a), the translation error $\Delta t_{GC,t}$ is under 0.5 m for most situations. The method RBPF_GC also reaches translation errors under 0.1 m which we state as very precise in terms for on-line SLAM approaches. The translation error is not increasing over time which means that we are able to compensate the drift behavior of the ego vehicle's odometry information with the concept of RBPFs in a correct way. Take into account, that the method RBPF_GC only runs with 60 particles. Errors larger than 0.7 m are not visible. This shows us that the implemented version of the RBPF is robust against huge drifts or outliers.

By comparing the Fig. 6.4(b) with Fig. 6.4(a), we observe, that method RBPF_GC has a better performance than method RBPF_EX. This is especially visible in the upper part of the driven path. The distributions of the lateral and longitudinal errors $\Delta x_{m,t}$ and $\Delta y_{m,t}$, respectively, also confirm this statement (see Fig. 6.5). As an example, the amount of absolute lateral errors larger than 0.4 m is much higher for method RBPF_EX than for method RBPF_GC. Consequently, the MAEs show also a better performance for method RBPF_GC: for the lateral component, the MAE_{GC} is 0.21 m and the MAE_{EX} is 0.24 m. The MAE_{GC} for the longitudinal component is 0.20 m and the MAE_{EX} is 0.23 m. The MAEs of the lateral component do not differ much from the MAEs of the longitudinal component. This means, that the implemented RBPF shows no weaknesses in the estimation one of these components. Based on these facts, we observe that the novel mapping approach improves the performance of RBPFs in terms of the translation accuracy. Since the sampling procedure of the RBPF is independent from the observations, and therefore independent from the chosen mapping method, the reason for a better performance can only be a more accurate weight estimation. This statement is reasonable, since the weight estimation is based on map matching in which more accurate maps are used, as long as the novel approach is chosen. That the novel mapping approach is able to generate more accurate maps was already shown in Sec. 5.2.3.

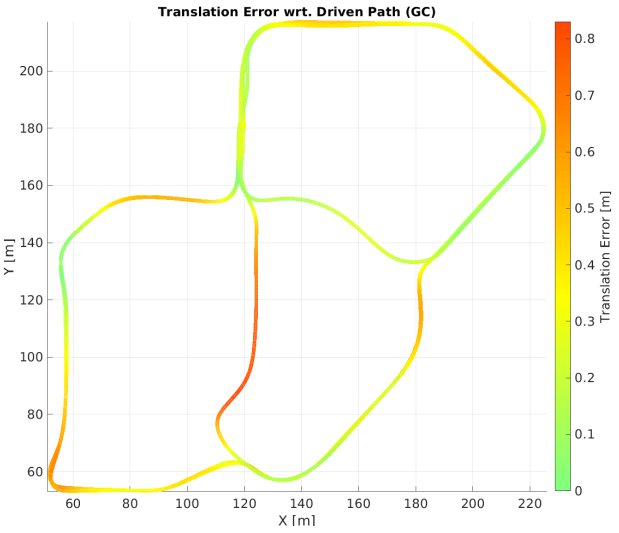
The performance of the heading errors must be regarded as more critical than the translation errors. The comparison of the heading errors, visualized in Fig. 6.4(c) and Fig. 6.4(d), shows that method RBPF_EX has a slightly better performance than the novel method RBPF_GC. This also becomes clear by taking the histograms and the MAEs in Fig. 6.5 into account: the MAE_{GC} is 0.54 deg, and the MAE_{EX} performs slightly better with 0.51 deg. The reason, why the novel approach RBPF_GC shows no benefits with regard to the heading error, is unclear at this point. In most situations, the heading errors are under 0.5 degree, but we also observe errors up to 2 deg for both methods. Furthermore, we observe that the heading errors increase especially in narrow turns (see Fig. 6.4(c)-6.4(d)). We state, that heading errors within the range of 0.5 deg and higher are critical for autonomous driving applications. Therefore, improvements have to be done to increase the heading accuracy. It should be noted that the achieved evaluation assessments relate to a comparison with the results of a full SLAM approach.



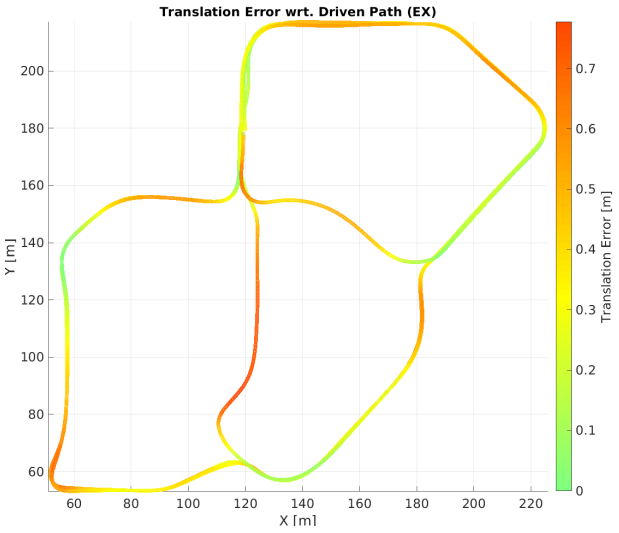
(a) $\mathcal{M}_{\mathrm{RE}}$ (b) Pure odometry

(c) $\widehat{\mathcal{M}}_{\mathrm{GC}}$ (d) $\widehat{\mathcal{M}}_{\mathrm{EX}}$

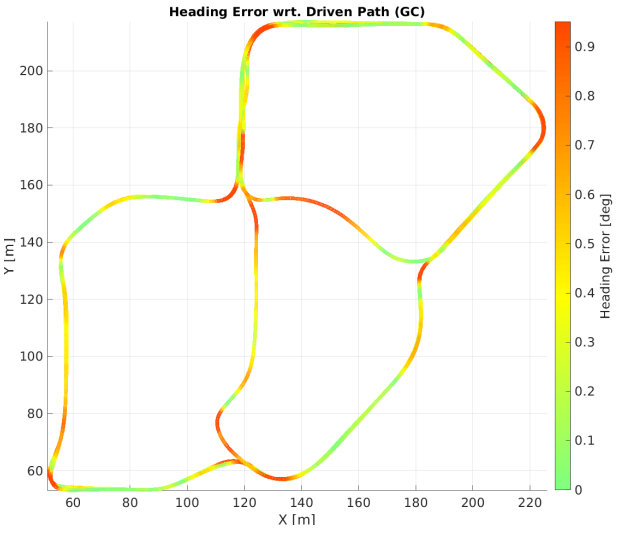
Figure 6.3: Map results using reference pose data (Fig. 6.3(a)), pure odometry information (Fig. 6.3(b)), and the results of RBPFs (Figs. 6.3(c)-6.3(d)). The maps were estimated based on their corresponding trajectories $\mathcal{X}_{0:t,\mathrm{RE}}$, $\widehat{\mathcal{X}}_{0:t,\mathrm{GC}}$, $\widehat{\mathcal{X}}_{0:t,\mathrm{EX}}$, and pure odometry information respectively. Ambiguities and inconsistencies in the map are clearly seen by using only odometry information. A considerable difference between $\mathcal{M}_{\mathrm{RE}}$ and $\widehat{\mathcal{M}}_{\mathrm{GC}}$ is not observable by eye balling. The free space estimation of $\widehat{\mathcal{M}}_{\mathrm{GC}}$ is cleaner and smoother compared to $\widehat{\mathcal{M}}_{\mathrm{EX}}$ which is coherent with the insights of Sec. 5.1 and Sec. 5.2.



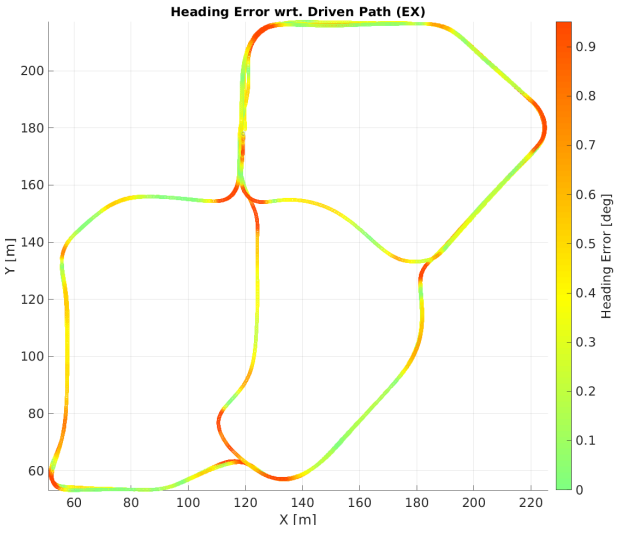
(a) Translation error $\Delta t_{GC,t}$.



(b) Translation error $\Delta t_{EX,t}$.



(c) Absolute heading errors $|\ \Delta\phi_{GC,t}\ |$.



(d) Absolute heading errors $|\ \Delta\phi_{EX,t}\ |$.

Figure 6.4: Translation and absolute heading error with regard to the position of the vehicle. Figure 6.4(a) and Fig. 6.4(b) show the translation error for the methods $\mathcal{M}_{\mathrm{GC}}$ and $\mathcal{M}_{\mathrm{EX}}$, respectively. Figure 6.4(c) and Fig. 6.4(d) show the absolute heading error for the two methods. The comparison between Fig. 6.4(b) and Fig. 6.4(a) shows, that the RBPF based on method $\mathcal{M}_{\mathrm{GC}}$ has a better performance than using $\mathcal{M}_{\mathrm{EX}}$.

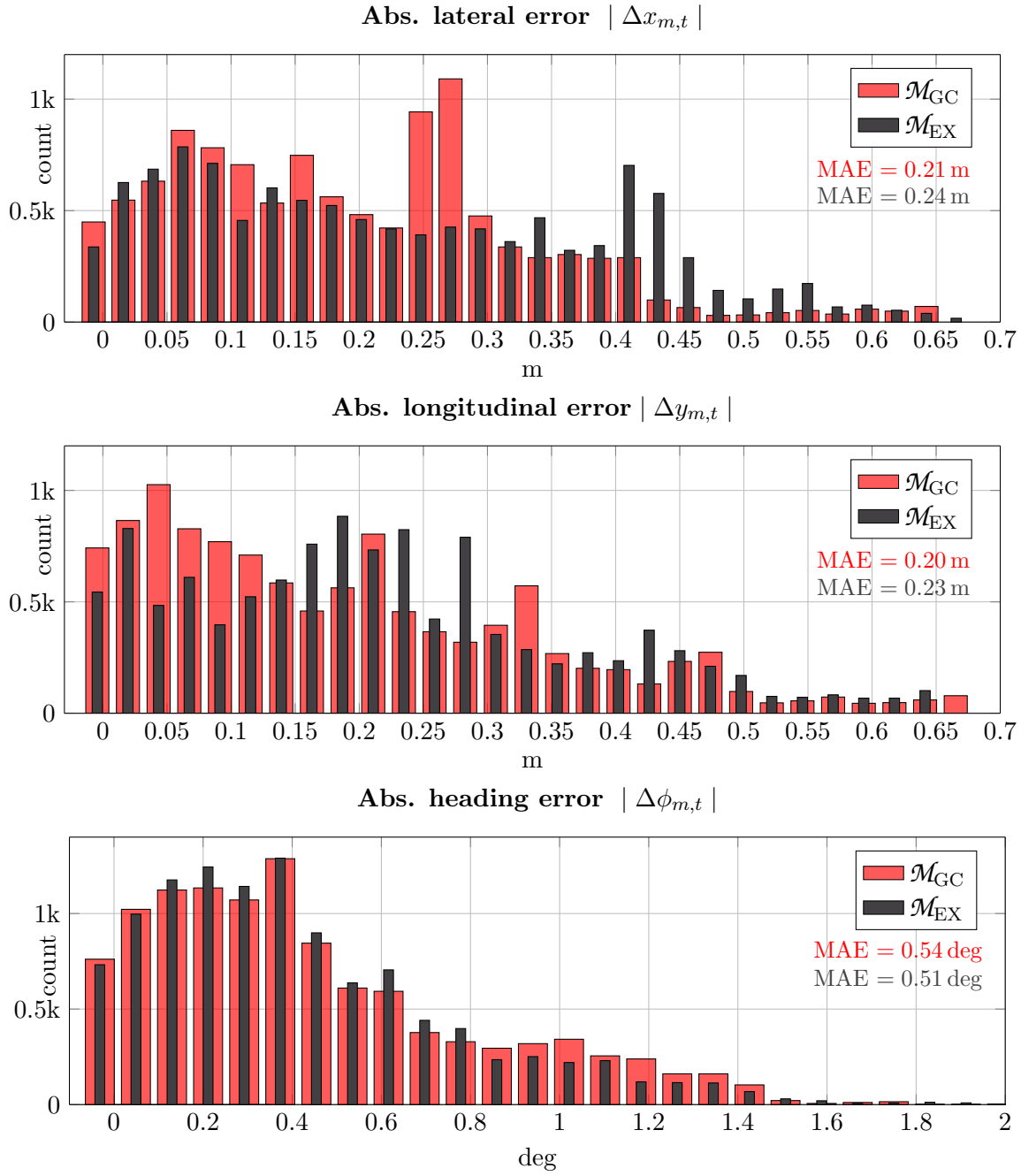


Figure 6.5: Histograms of absolute lateral errors | $\Delta x_{m,t}$ | (top), absolute longitudinal errors | $\Delta y_{m,t}$ | (middle), and absolute heading errors | $\Delta \phi_{m,t}$ | (bottom). The MAE_{GC} for the lateral component is 0.21 m and MAE_{EX} is 0.24 m (top). The MAE_{GC} for the longitudinal component is 0.20 m and MAE_{EX} is 0.23 m (middle). The MAE_{GC} for the heading component is 0.54 deg and MAE_{EX} is 0.51 deg (bottom).

6.5 Summary and Conclusion

6.5.1 Summary

In this chapter we evaluated the realization of the RBPF based on real-world data which were recorded with a Mercedes-Benz research vehicle. A full SLAM approach (see Sec. 2.6, and Sec. 2.8) was applied to generate reference poses and reference occupancy grid map data in a semi-automated way. Our focus was on the evaluation of the lateral, the longitudinal, and the heading component of pose errors between the reference poses and the estimated poses. With regard to the lateral and longitudinal component, we observed that method RBPF_GC outperforms method RBPF_EX which does not take the dependencies of neighboring grid cells into consideration. The $\mathrm{MAE}_{\mathrm{GC}}$ was about 20 cm for the longitudinal and lateral component. In comparison, the $\mathrm{MAE}_{\mathrm{EX}}$ was about 24 cm for both components. In contrast to these insights, it surprised us that no improvements for heading accuracies were achieved using the novel mapping approach during the RBPF. Here, the $\mathrm{MAE}_{\mathrm{GC}}$ was 0.54 deg.

6.5.2 Conclusion

The current implementation of the RBPFs using the novel grid mapping technique is able to correct the odometry drift of the ego vehicle's motion behavior in a correct way although the number of 60 particles is quite small. To enhance the pose accuracy in the future, one possibility is to increase the number of particles. Here, one solution would be to improve the hardware settings of our system. For example, current state-of-the-art graphic cards like the NVIDIA GeForce GTX TITAN Z which provides 12 GB RAM would allow a immense performance boost. It would allow us to raise the number of particles by almost a factor of 10 which would result in a finer sampling of the state space. Based on this fact, we assume a better estimation of the poses, especially for the heading component. We would also be able to represent larger map areas, if we improve the hardware settings.

Another possibility is the change of the definition of the proposal distribution which is governed by the motion model in our current situation. As mentioned in [Grisetti et al., 2007], incorporating recent sensor observations into the proposal distribution can massively improve the performance of the filter as long as the observation model performs considerable better than the motion model.

The improvement of the estimation of our control parameters should also happen in future work. At this point, the velocity noise behavior σ_{v_t} in the motion model as well as the scaling factor α in the observation model are tuned by hand. Like the estimation of the yaw rate noise behavior, empirical studies should be carried out to give us more insights with regard to the noise behavior of the velocity. Similar to the evaluation steps in Sec. 5.1, where we evaluated different configuration settings against GT, we should validate a set of different scaling factors to find the best solution for α. Another possibility would be to estimate the internal model parameters on-line using schemes like the EM algorithm as described in [Thrun et al., 2005, chapter 6.3.2]. We should also validate the influence of the frequency of the weight estimation step which is not done yet. Furthermore, we should compare the results of other observation models during the weight estimation step with our current performance in the future. One example could be to use correlation based approaches instead of score based approaches. This idea was also mentioned in Dömötör [2014].

In this evaluation chapter, we did not take the precision of the reference data as well as the precision of the RBPFs into account. Since both methods theoretically provide this information we should take these precisions into account for further evaluation steps.

At this point, we did not use GT data in this evaluation chapter. If we were able to improve our hardware settings and our RAM memory management in a way that larger maps and/or more particles can be realized, we could use raw data sets from the KITTI vision benchmark suite [Geiger et al., 2012, 2013]. These GT data sets contain recordings of large scale environments and loop closures. If this would be the case, an ego motion estimator based on stereo vision [Badino, 2004] would be used for the motion model instead of using the pure odometry information of the car. Furthermore, we would also be able to evaluate the mentioned full SLAM approach described in Sec. 6.2 against GT data.

Chapter 7

Conclusion and Outlook

7.1 Conclusion

In this thesis, a novel, incremental occupancy grid mapping approach is presented and evaluated which takes the dependencies of neighboring grid cells into account. The generated occupancy grid maps represent static obstacles as well as free space area in a probabilistic way. Here, we address the field of autonomous driving and advanced driving assistance systems where such occupancy grid maps are utilized for planning or localization purposes. As input data, we rely on the Stixel World, a highly compact environment representation which is generated based on dense disparity images. The Stixel World describes obstacles as vertically oriented, adjacent rectangles which are defined by its distance, width and height. The current free space is also represented in this super pixel representation. However, the concept of the novel mapping approach can also deal with other sensor data which provides spatial environment information.

The novelty of the presented system lies in the fact that we model explicitly dependencies between neighboring grid cells which is not a common practice in occupancy grid mapping approaches. It is well known that neighboring grid cells influence each other which shows that they are dependent from each other. This fact is often neglected in order to allow efficient and straight forward on-line occupancy grid mapping. Here the major challenge is to realize a system which keeps the dependencies of neighboring grid cells into account and simultaneously allows an incremental framework with real time requirements. In this thesis we have specifically focused on the realization of such system which allows to run on-line in autonomous research vehicles.

The novel mapping approach is formulated as an optimization problem in a probabilistic fashion. We have exploited the power of MRFs which allows us to interpret the occupancy grid map as an undirected graph where each node represents a grid cell and the dependencies between them are modeled as undirected edges. Under the assumption that the observations and the vehicle's pose is given, the posterior of the map is described as a product of unary and binary terms. The unary terms include the measurement model in which we define the uncertain behavior of the current observations in the column-disparity space. The unary terms also involve the prediction step, where the occupancy grid mapping results of the previous time step are taken into account. Based on this, the incremental map update formula is realized. The binary constraints are modeled as a data independent Potts model.

To allow an incremental mapping scheme, it is necessary that marginal probabilities are estimated which are used during the prediction step. This is realized with the help of dynamic graph cuts, an efficient inference scheme, to solve dynamically gradually changing MRFs. Only with the help of dynamic graph cuts we are able to estimate marginal probabilities in a feasible computation time.

In this thesis we have also taken into account the uncertainty of the pose of the vehicle which leads to the well-known SLAM problem. To solve the on-line SLAM problem we applied state-of-the-art Rao-blackwellized particle filters (RBPF) which separates the estimation of the trajectory from the estimation of the map. For the estimation of the map we used our novel mapping approach. The estimation of the trajectory is solved via a particle filter, where we used a precise motion model for the sampling step and a score based map matching technique for the weight estimation.

To evaluate the novel occupancy grid mapping approach artificial image sequences as well as real-world data were taken into account. Detection rates and geometrical errors of obstacles were presented in this thesis. We also compared the novel approach against a grid mapping method which does not take the dependency of neighboring grid cells into account. It was shown that the approach based on MRFs has a better performance with regard to detection rates for artificial and real-world data. We have proven that the occupancy grid mapping approach with the use of MRFs produces a much cleaner and smoother free space. Under real-world conditions, detection rates up to 90 % for free space were achieved using the novel mapping approach. Detection rates for obstacles are in the range of 85 %.

Under optimal conditions, the geometrical accuracy of obstacles are in the range of 10 cm or less if a disparity sampling rate of at least 8 is chosen. For real-world data, the mean geometrical accuracy of static obstacles is 0.36 m. We figured out that the quality of obstacles in the occupancy maps depends on the distance relatively to the car and how often map regions were updated. Static obstacles, which are close to the vehicle and appear over a long time period, are precisely mapped and have sharp contours.

The developed on-line SLAM approach based on RBPFs was evaluated against a full SLAM approach. We were able to correct the uncertainty behavior of the odometry information of the vehicle using only 60 particles. For the longitudinal and lateral pose component, we observed that the RBPF with the novel mapping approach performs better compared to RBPFs which do not take the dependencies of neighboring grid cells into account. A mean absolute pose error of 20 cm was achieved.

The grid mapping system reaches its limits in unstructured environments where the Stixel definition is no longer valid. Based on the current Stixel definition, we are not able to model obstacles with low height like traffic islands and curbs. Using the chosen evaluation measures, it was not possible to show, that the novel mapping approach achieves higher geometrical accuracies for obstacles than approaches which do not model the dependencies of neighboring grid cells. In the evaluation of the SLAM approach it was surprising that no improvements for the heading estimation were achieved. A mean absolute heading error of 0.5 deg was obtained.

7.2 Outlook

Improvements of the Evaluation Techniques. We evaluated the occupancy grid mapping approach based on MRFs with artificial and real-world data in a comprehensive way. Nevertheless, we were not to able to show for real-world data, that the mapping technique based on MRFs produces grid maps with higher geometrical accuracies than grid map approaches without MRFs. We are optimistic that further experiments and evaluation measures can prove the stated assumption.

Moreover, the validation of the influence of additional control parameters should also be considered in future. Especially the impact of the change of the transition probabilities in the prediction step, as well as changing the control parameters of the binary terms should be evaluated in detail.

In our evaluation we observed that the quality of obstacles is dependent on the distance relatively to the car. This observation is based on qualitative results and therefore, it should also be proven in a quantitative way.

We validated the developed on-line SLAM approach against a reference off-line SLAM algorithm. It is necessary that we also make accuracy assessments based on GT data. Here, the KITTI Vision Benchmark Suite [Geiger et al., 2012, 2013] can also be used for the SLAM evaluation.

Improving the SLAM Approaches. The results of the evaluation of the on-line SLAM approach showed, that it reaches its limits in the estimation of high accurate heading angles. One major reason is that a relatively small number of particles were used. Therefore, we should optimize the hardware settings in the future to allow larger numbers of particles. Furthermore, an optimization of the RAM memory management should be carried out.

At this time we assume that the odometry motion behavior of the vehicle has a small drift behavior. To be more independent of the odometry motion model, we should incorporate recent sensor observations during the sampling procedure of the RBPF as suggested by Grisetti et al. [2007]. This would lead to be more independent against the motion model of the vehicle.

We applied a graph-based off-line SLAM approach as reference in which the loop closure problem was solved by hand. We should also be able to manage the loop close issue automatically in order to improve the maturity of this approach.

Incorporating Height Information. The current system is modeled in the 2D space and neglects the height of the environment. Height information is provided by many sensor data and should be considered in future versions. One possibility is to model the full 3D space which would result in a voxel grid map scheme. This step would be very expensive with regard to the computation time, especially in the estimation of the marginal probabilities. It would be more efficient to include an additional height attribute for each grid cell. It would result in an additional height map layer and a 2.5D representation. The consideration of the height information in the current, probabilistic framework is also a topic for future work.

Consideration of additional Semantic Information. In the current mapping approach we used the segmented Stixel World to exclude dynamic obstacles which increased the quality of our maps. Nevertheless, obstacles which are not long-term static, like parked cars, still standing pedestrians or trash bins, should also be excluded from the occupancy grid maps. Additional

semantic information of the environment would help to detect and excluded these types. One possibility would be the use of the semantic Stixel World [Schneider et al., 2016] which provides precise semantic information of the environment. Semantic information would also help us to weight long-term static obstacles in the measurement model. As an example, class types like buildings and poles should have higher importance than classes like vegetation or terrain. An example of an overlay of semantics with static environment information is shown in Fig. 7.1.

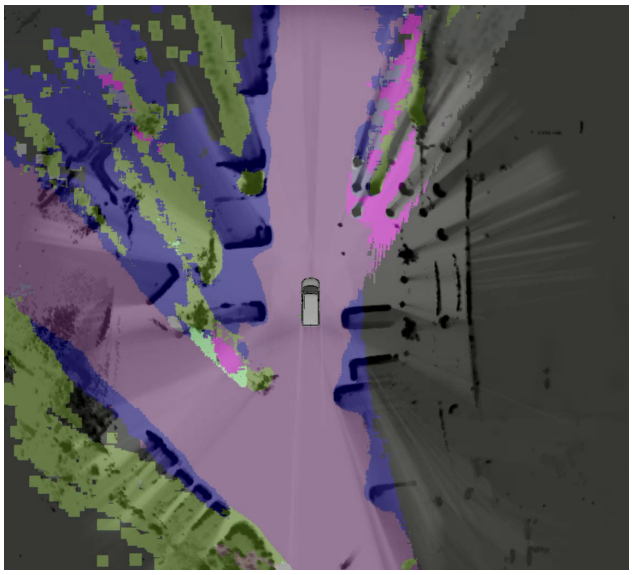


Figure 7.1: Example of an overlay of static information with semantic labels. Here, LIDAR data is used for the generation of the occupancy grid map. Grid cells are labeled as road (purple), sidewalks (pink), vehicles (blue), vegetation/terrain (green), and buildings (bright gray). Additional semantic information, like the work of Schneider et al. [2016], would help us to label static environment information in a precise way. This information can be considered in the measurement update scheme of incremental mapping techniques.

Life-Long Occupancy Grid Mapping Our approach is focused on precise incremental occupancy grid mapping to represent current free space and obstacles. However, changes in the long-term static environment should also be considered in future to maintain and increase the map quality over a long time period. Life-long (occupancy grid) mapping handles this issue, where the environment is learned over time periods based on chronological data recordings and by adding explicitly a temporal dimension to the map learning process. As an example, the work of Santos et al. [2016] deals with this issue. In future, our current approach should be extended by the idea of life-long map learning.

Altogether, the thesis demonstrated the potential of using MRFs in combination with incremental occupancy grid mapping approaches. We are able to produce highly accurate digital maps which can be used for autonomous driving applications in future.

Appendices

Appendix A

Derivation of the Particle Filter

During this thesis particle filters are used to solve the well-known SLAM problem. Here, we describe the general derivation of the particle filter approach, particularly the *importance sampling step*. This derivation is the basis of the SIR particle filter which is described in Sec. 2.7.1. We also refer to [Thrun et al., 2005, chapter 4.3.2] for more detailed information.

Target and Proposal Distribution The particle filter approach deals with the challenge to compute the expectation $E_p(\delta(x \in A))$ of an arbitrary *target density function* $p(x)$ over the state $x \in A$. Here, $\delta()$ is the Diract function which returns 1 if its argument is true and otherwise 0; A is an arbitrary region. The expectation can be expressed by

$$E_p\big(\delta(x \in A)\big) = \int p(x)\ \delta(x \in A) dx \tag{A.1}$$

$$= \int \underbrace{\frac{p(x)}{\pi(x)}}_{=:w(x)}\ \pi(x)\ \delta(x \in A) dx \tag{A.2}$$

$$= E_\pi\big(w(x)\delta(x \in A)\big). \tag{A.3}$$

We introduce the *proposal distribution* $\pi(x)$ and we define that $p(x) > 0 \longrightarrow \pi(x) > 0$. The weight $w(x)$ represents the "offset" between the target $p(x)$ and the proposal distribution $\pi(x)$.

Introducing Samples By following the ideology of the particle filter, we would like to sample from the target distribution $p(x)$, which is, unfortunately, impossible. Because of this fact, we draw from the proposal distribution $\pi(x)$ to achieve a particle filter set, which represents the distribution of $\pi(x)$. Now, the integral of $\pi(x)$ over the region A can approximately described by the sum over all particles:

$$\int_A \pi(x)\ dx\ \approx \frac{1}{M} \sum_{i=1}^{M}\ \delta(x^{(i)} \in A). \tag{A.4}$$

Here, $x^{(i)}$ represents an individual particle i of the total number of M samples. We now introduce the individual importance weights for each sample

$$w^{(i)} = \frac{p(x^{(i)})}{\pi(x^{(i)})} \ , \tag{A.5}$$

which corresponds to the importance weighting step of Sec. 2.7.1. With the definition of the weights the integral of $p(x)$ over A could be expressed as:

$$\int_A p(x)\ dx \ \approx \eta \sum_{i=1}^{M}\ w^{(i)}\ \delta(x^{(i)} \in A) \tag{A.6}$$

$$= \lim_{M\to\infty} \eta \sum_{i=1}^{M}\ w^{(i)}\ \delta(x^{(i)} \in A) \ , \text{ with } \eta = \left[\sum_{i=1}^{M} w^{(i)}\right]^{-1} . \tag{A.7}$$

This derivation implies the following statements:

- The density of the target distribution $p(x)$ could be approximately described by a weighted particle filter set which is drawn from the proposal distribution $\pi(x)$.
- The more samples are used, the better is the approximation of the target distribution.

Appendix B

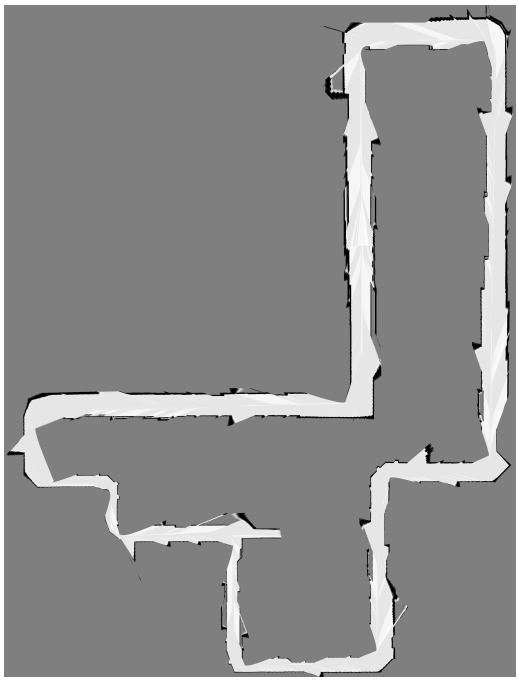
Additional Results for the Evaluation with an Artificial Ground Truth Data Set

B.1 Occupancy Grid Maps for different Configurations

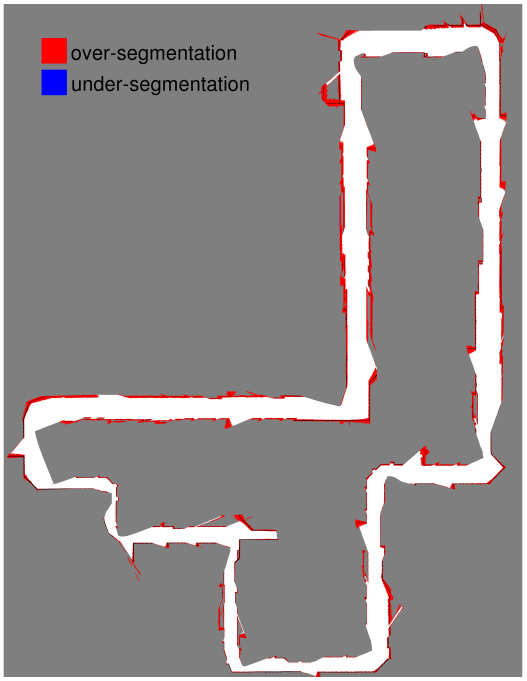
In Sec. 5.1.1.3 the global grid map $\mathcal{M}_{\mathrm{GC}}$ was shown using a configuration of $s_w = 3\,\mathrm{pel}$ and $d_s = 08$. Here, we present two additional grid map results with (1) a setup of $s_w = 9\,\mathrm{pel}$ and $d_s = 02$ (Fig. B.1(a)) and (2) with a setup of $s_w = 1\,\mathrm{pel}$ and $d_s = 16$ (Fig. B.1(c)). The comparison shows how much influence the disparity intervals d_s have with regard to occupied areas. The Fig. B.1(b) and Fig. B.1(d) show the overlay with the GT map data $\mathcal{M}_{\mathrm{GT}}$.

B.2 Close-ups for all Configurations

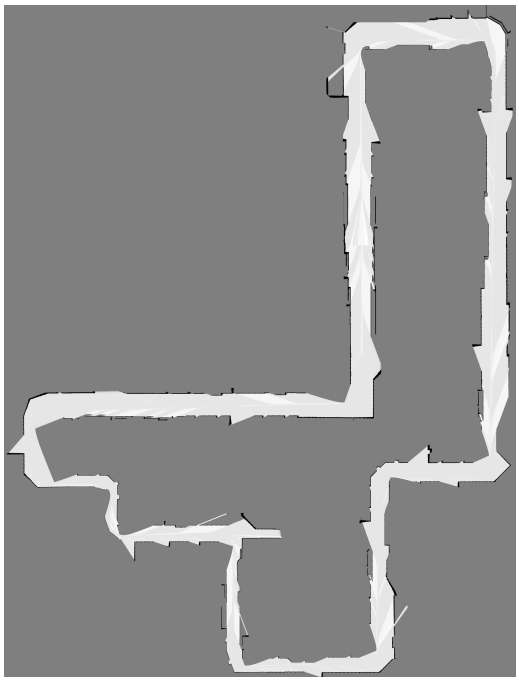
In Fig. 5.15 three different map samples with different configurations were shown to visualize how the disparity interval influences the wedge effect. As regards the completeness, we also present all samples in Fig. B.2.



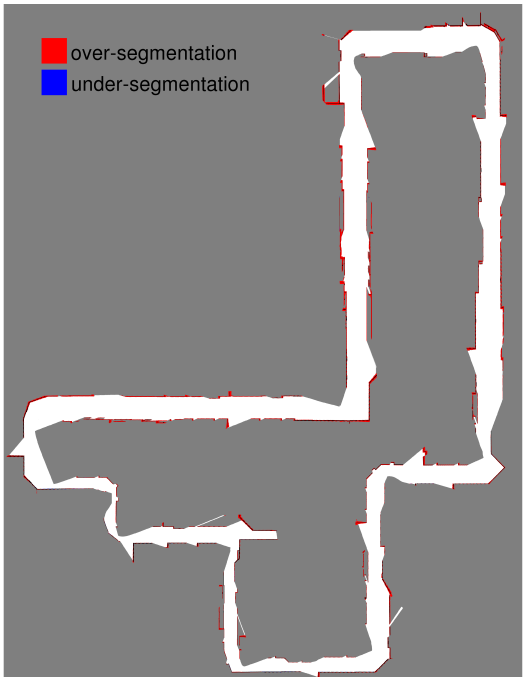
(a) Setup of $s_w = 1\,\text{pel}$ and $d_s = 16$



(b) Overlay with $\mathcal{M}_{\text{GT}}$

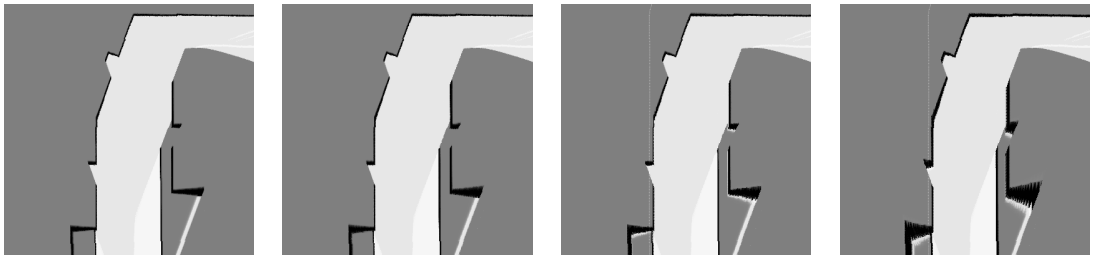


(c) Setup of $s_w = 9\,\text{pel}$ and $d_s = 02$

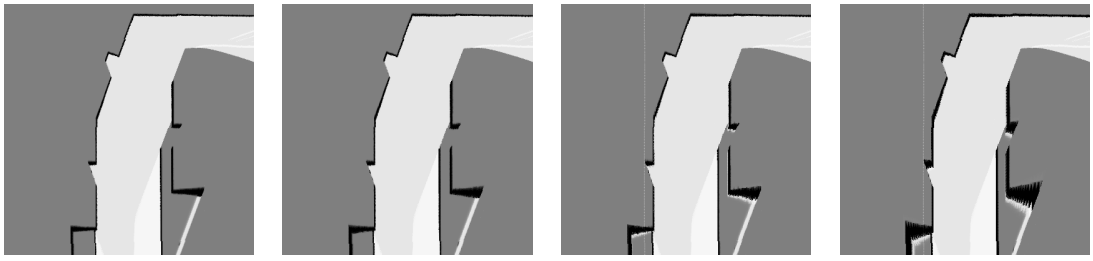


(d) Overlay with $\mathcal{M}_{\text{GT}}$

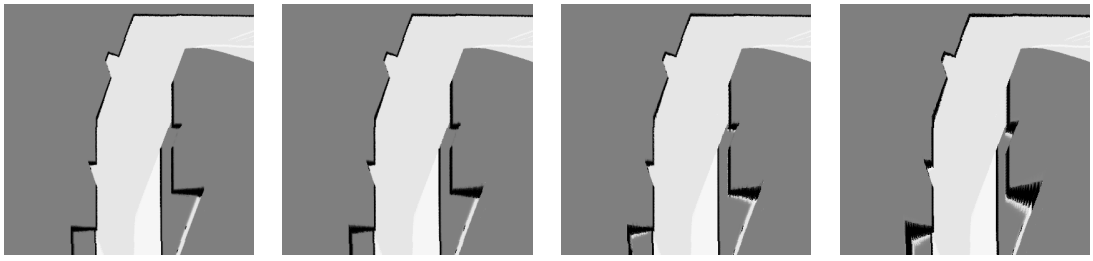
Figure B.1: The global occupancy grid map for two additional setups. In B.1(a) a setup of $s_w = 9\,\text{pel}$ and $d_s = 02$ was chosen. In B.1(c) a setup of $s_w = 1\,\text{pel}$ and $d_s = 16$ was applied. This figure helps to make the huge influence of the disparity interval clear. The figures B.1(b) and B.1(d) show the overlay with GT map data $\mathcal{M}_{\text{GT}}$.



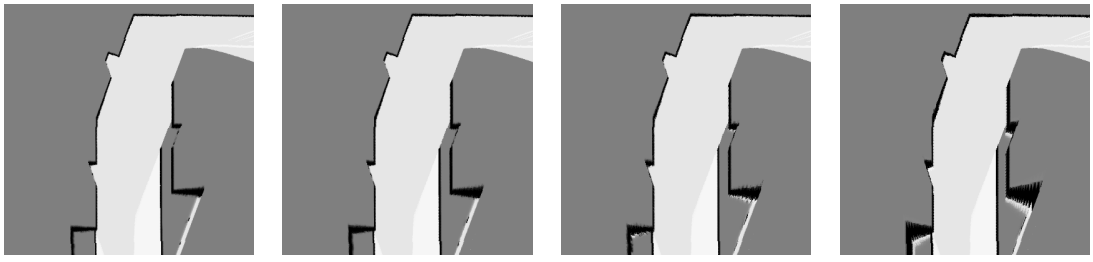
(a) $s_w = 1\,\text{pel}$, $d_s = 16$ (b) $s_w = 1\,\text{pel}$, $d_s = 08$ (c) $s_w = 1\,\text{pel}$, $d_s = 04$ (d) $s_w = 1\,\text{pel}$, $d_s = 02$



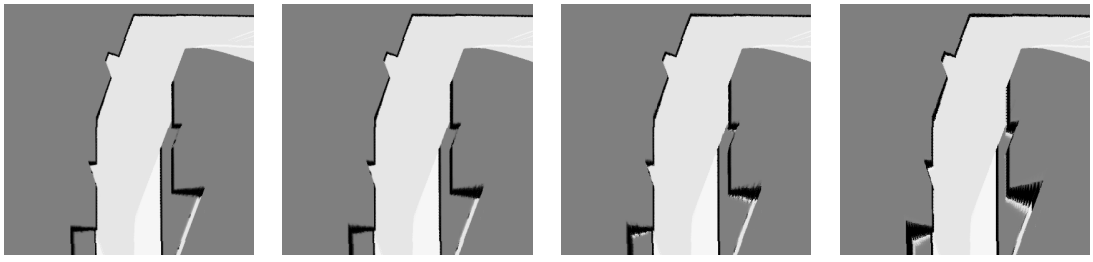
(e) $s_w = 3\,\text{pel}$, $d_s = 16$ (f) $s_w = 3\,\text{pel}$, $d_s = 08$ (g) $s_w = 3\,\text{pel}$, $d_s = 04$ (h) $s_w = 3\,\text{pel}$, $d_s = 02$



(i) $s_w = 5\,\text{pel}$, $d_s = 16$ (j) $s_w = 5\,\text{pel}$, $d_s = 08$ (k) $s_w = 5\,\text{pel}$, $d_s = 04$ (l) $s_w = 5\,\text{pel}$, $d_s = 02$



(m) $s_w = 7\,\text{pel}$, $d_s = 16$ (n) $s_w = 7\,\text{pel}$, $d_s = 08$ (o) $s_w = 7\,\text{pel}$, $d_s = 04$ (p) $s_w = 7\,\text{pel}$, $d_s = 02$



(q) $s_w = 9\,\text{pel}$, $d_s = 16$ (r) $s_w = 9\,\text{pel}$, $d_s = 08$ (s) $s_w = 9\,\text{pel}$, $d_s = 04$ (t) $s_w = 9\,\text{pel}$, $d_s = 02$

Figure B.2: Sample maps for all configurations

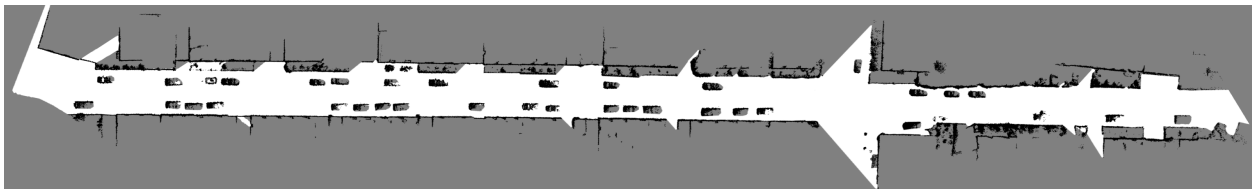
Appendix C

Additional Results for the Evaluation with Real World Data

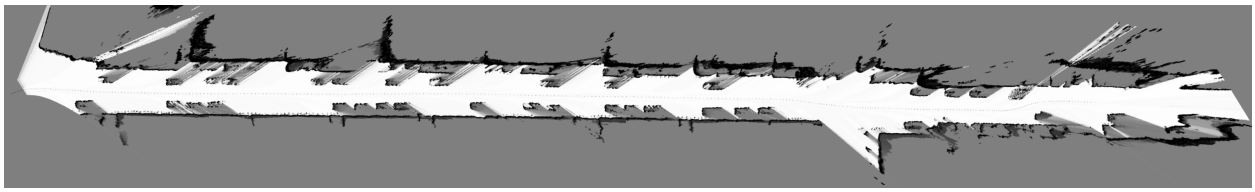
C.1 Further Results of Reference Grid Maps $\mathcal{M}_{\mathbf{RE}}$ and Estimated Grid Maps $\mathcal{M}_{\mathbf{GC}}$

In Sec. 5.2.1.2 the results for the reference occupancy grid maps and of $\mathcal{M}_{\mathrm{GC}}$ were shown. Here, we show additional results of the remaining three sequences 0039, 0023, and 0087 in Fig. C.2 and Fig. C.3.

Additional close-ups of reference and estimated maps are shown in Fig. C.4. This figure also includes real world images where interesting areas are highlighted with colored circles.

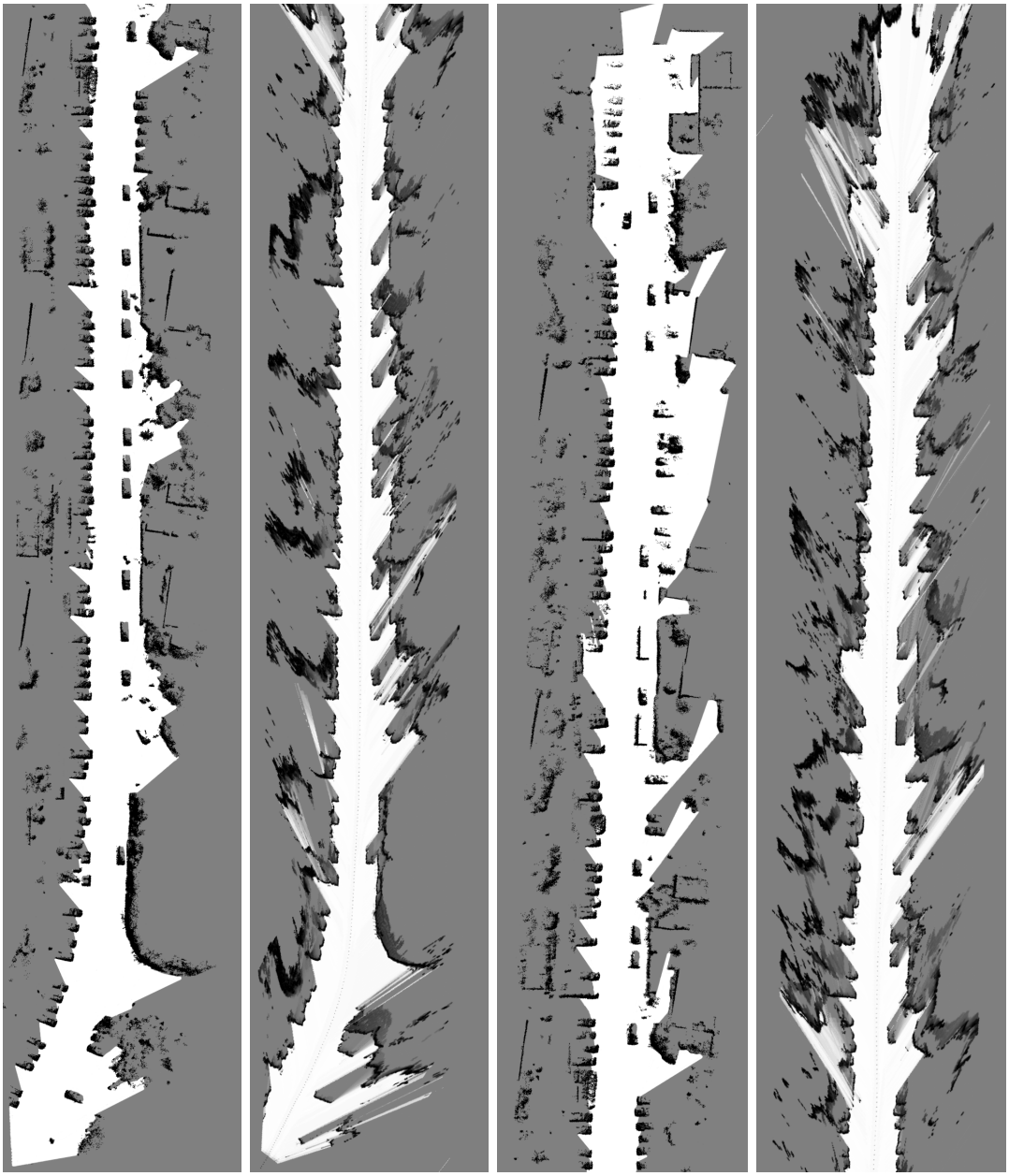


(a) $\mathcal{M}_{\mathrm{RE}}$



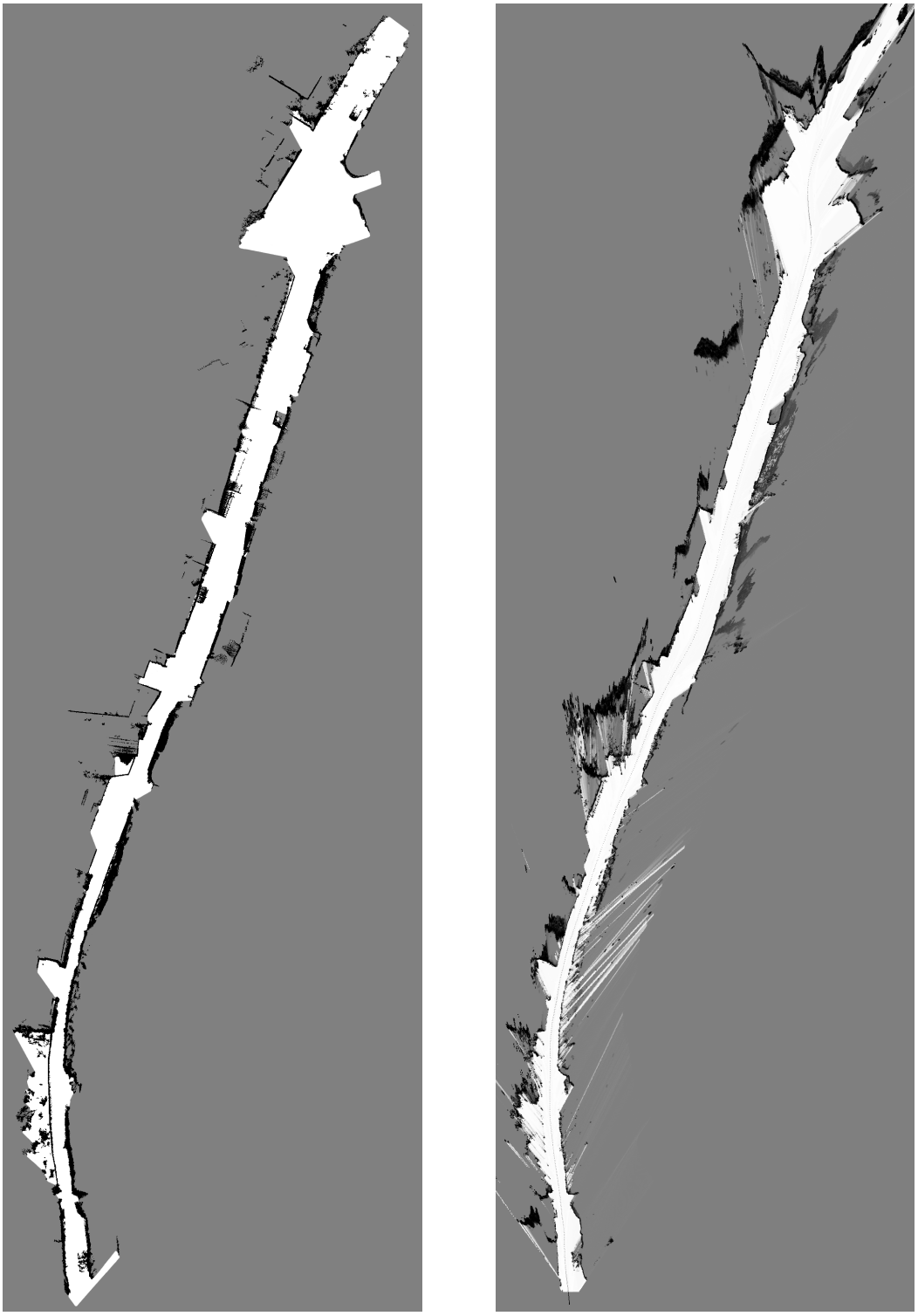
(b) $\mathcal{M}_{\mathrm{GC}}$

Figure C.1: Estimated occupancy grid maps for sequence 0039, $310 \times 70\,m$.



(a) $\mathcal{M}_{\mathrm{RE}}$, part I (b) $\mathcal{M}_{\mathrm{GC}}$, part I (c) $\mathcal{M}_{\mathrm{RE}}$, part II (d) $\mathcal{M}_{\mathrm{GC}}$, part II

Figure C.2: Estimated occupancy grid maps for sequence 0023, $460 \times 70\,m$.



(a) $\mathcal{M}_{\mathrm{RE}}$ (b) $\mathcal{M}_{\mathrm{GC}}$

Figure C.3: Estimated occupancy grid maps for sequence 0087,$320 \times 130\,m$.

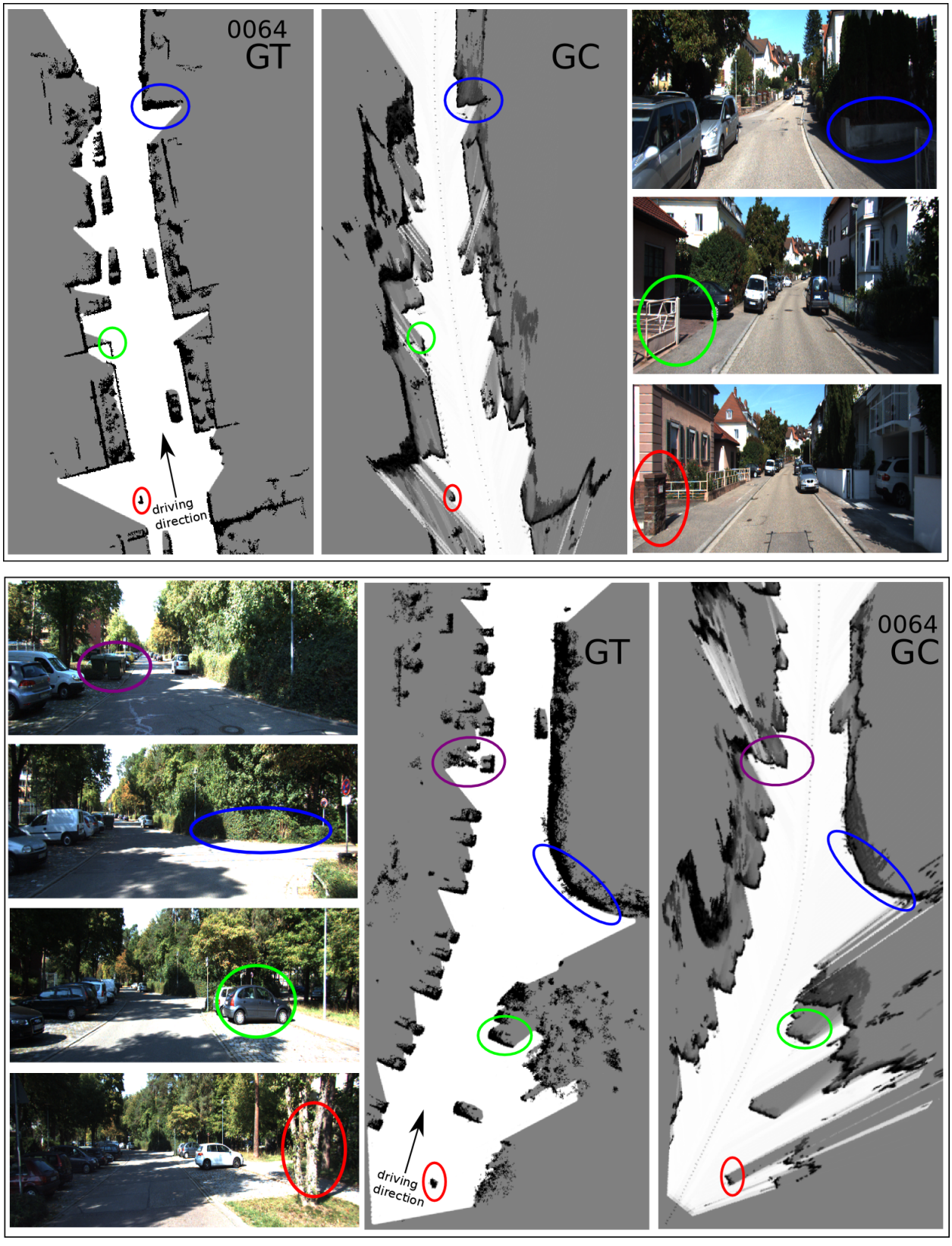


Figure C.4: Close-ups of $\mathcal{M}_{\mathrm{GC}}$ and $\mathcal{M}_{\mathrm{RE}}$ for sequence 0064.

Bibliography

Achanta, R., A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk (2012). Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 34:2274–2282.

Aeberhard, M., S. Rauch, M. Bahram, G. Tanzmeister, J. Thomas, Y. Pilat, F. Homm, W. Huber, and N. Kaempchen (2015). Experience, results and lessons learned from automated driving on germany's highways. *IEEE Intelligent Transportation Systems Magazine 7:42-57.*

Altendorfer, R. and S. Matzka (2010). A confidence measure for vehicle tracking based on a generalization of bayes estimation. In *IEEE Intelligent Vehicles Symposium (IV)*, San Diego, USA, pp. 30–41.

Badino, H. (2004). A robust approach for ego-motion estimation using a mobile stereo platform. In *International Workshop on Complex Motion (IWCM)*, Günzburg, Germany, pp. 198–208.

Badino, H., U. Franke, and R. Mester (2007). Free space computation using stochastic occupancy grids and dynamic programming. In *International Conference on Computer Vision (ICCV), Workshop*, Rio de Janeiro, Brazil.

Badino, H., U. Franke, and D. Pfeiffer (2009). The stixel world - a compact medium level representation of the 3d-world. In *German Association for Pattern Recognition (DAGM)*, Jena, Germany, pp. 51–60.

Badino, H., A. Yamamoto, and T. Kanade (2013). Visual odometry by multi-frame feature integration. In *International Workshop on Computer Vision for Autonomous Driving (ICCV)*, Sydney, Australia.

Barth, A. (2010). *Vehicle Tracking and Motion Estimation Based on Stereo Vision Sequences.* Ph. D. thesis, Institut für Geodäsie und Geoinformation (IGG), Bonn, Germany.

Barth, A., J. Siegemund, A. Meißner, U. Franke, and W. Förstner (2010). Probabilistic multi-class scene flow segmentation for traffic scenes. In *German Association for Pattern Recognition (DAGM)*, Darmstadt, Germany, pp. 503–512.

Bauer, M. (2011). *Vermessung und Ortung mit Satelliten. Globale Navigationssysteme (GNSS) und andere satellitengestützte Navigationssysteme.* Wichmann.

Bellman, R. (1954). The theory of dynamic programming. *Bulletin of the American Mathematical Society*, 6:503–515.

Benenson, R., M. Mathias, R. Timofte, and L. J. V. Gool (2012). Pedestrian detection at 100 frames per second. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Providence, USA, pp. 2903–2910.

Benenson, R., M. Mathias, R. Timofte, and L. Van Gool (2012). Fast stixel computation for fast pedestrian detection. In *European Conference on Computer Vision (ECCV)*, Florence, Italy, pp. 11–20.

Bengler, K., K. Dietmayer, B. Färber, M. Maurer, C. Stiller, and H. Winner (2014). Three decades of driver assistance systems: Review and future perspectives. *IEEE Intelligent Transportation Systems Magazine*, 6:6–22.

Bergmann, N. (1999). *Recursive Bayesian Estimation*. Ph. D. thesis, Studies in Science and Technology, Department of Electrial Engineering, Linköping, Schweden.

Besl, P. J. and N. D. McKay (1992). A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 14:239–256.

Biswas, R., B. Limketkai, S. Sanner, and S. Thrun (2002). Towards object mapping in non-stationary environments with mobile robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, Lausanne, Switzerland, pp. 1014–1019.

Blake, A. and A. Zisserman (1987). *Visual Reconstruction*. MIT Press.

Blanco, J.-L. (2009). *Contributions to Localization, Mapping and Navigation in Mobile Robotics*. Ph. D. thesis, Dpt. de Ingenieria de Sistemas y Automatica, University of Malaga, Malaga, Spain.

BMW (2017). Enjoy every trip. intelligent driving. http://www.bmw.com/com/en/insights/technology/connecteddrive/2013/driver_assistance/intelligent_driving.html#lanekeeping.

Bosse, M., P. Newman, J. Leonard, M. Soika, W. Feiten, and S. Teller (2003). An atlas framework for scalable mapping. In *IEEE International Conference on Robotics and Automation (ICRA)*, Taipei, China, pp. 1899–1906.

Bouguet, J. (2000). Matlab camera calibration toolbox. http://www.vision.caltech.edu/bouguetj/index.html. 2017-05-10.

Boykov, Y., O. Veksler, and R. Zabih (2001). Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 23:1222–1239.

Brechtel, S., T. Gindele, and R. Dillmann (2010). Recursive importance sampling for efficient grid-based occupancy filtering in dynamic environments. In *IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, USA, pp. 3932–3938.

Brostow, G. J., J. Fauqueur, and R. Cipolla (2009). Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30:88–97.

Brown, M. Z., D. Burschka, and G. D. Hager (2003). Advances in computational stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 25:993–1008.

Chen, L., G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille (2014). Semantic image segmentation with deep convolutional nets and fully connected crfs. *CoRR*, abs/1412.7062.

Choi, J. (2014). Hybrid map-based slam using a velodyne laser scanner. In *IEEE Conference on Intelligent Transportation Systems (ITSC)*, Quingdao, China, pp. 3082–3087.

Clifford, P. (1990). Markov random fields in statistics. In *Disorder in Physical Systems: A Volume in Honour of John M. Hammersley*, pp. 19–32. Oxford University Press.

Continental (2011). Automotive industrial sensors: Ars 300 long range radar sensor 77 ghz. http://www.conti-online.com/generator/www/de/en/continental/industrial_sensors/themes/ars_300/ars_300_en.html. 2017-05-16.

Cordts, M., M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele (2016). The cityscapes dataset for semantic urban scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, USA, pp. 3213–3223.

Dalal, N. and B. Triggs (2005). Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Washingto, USA, pp. 886–893. IEEE Computer Society.

Danescu, R., F. Oniga, and S. Nedevschi (2011). Modeling and tracking the driving environment with a particle-based occupancy grid. *IEEE Transactions on Intelligent Transportation Systems*, 12:1331–1342.

Dellaert, F., D. Fox, W. Burgard, and S. Thrun (1999). Monte carlo localization for mobile robots. In *IEEE International Conference on Robotics and Automation (ICRA)*, Detroit, USA, pp. 1322–1328.

Dhiman, V., A. Kundu, F. Dellaert, and J. J. Corso (2014). Modern MAP inference methods for accurate and fast occupancy grid mapping on higher order factor graphs. In *IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, pp. 2037–2044.

Dömötör, N. (2014). Stixslam: Optimierung von sterebasierten belegtheitskarten mit hilfe von mehrfachbefahrungen. Master's thesis, Fakultät für Maschinenbau, Karlsruher Institut für Technologie.

Doucet, A., N. D. Freitas, and N. Gordon (2001). *Sequential Monte Carlo methods in practice*. Springer.

Doucet, A., N. d. Freitas, K. P. Murphy, and S. J. Russell (2000). Rao-blackwellised particle filtering for dynamic bayesian networks. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, San Francisco, USA, pp. 176–183.

Durrant-Whyte, H. (2001). Introduction to estimation and the kalman filter. Technical report, Australian Centre for Field Robotics. The University of Sydney.

Durrant-Whyte, H. and T. Bailey (2006). Simultaneous localization and mapping (slam): Part I the essential algorithms. *IEEE Robotics and Automation Magazine*, 13:99–110.

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

Enzweiler, M., M. Hummel, D. Pfeiffer, and U. Franke (2012). Efficient stixel-based object recognition. In *IEEE Intelligent Vehicles Symposium (IV)*, Alcal de Henares, Spain, pp. 1066–1071.

Erbs, F., B. Schwarz, and U. Franke (2012). Stixmentation - probabilistic stixel based traffic scene labeling. In *British Machine Vision Conference (BMVC)*, Surrey, England, pp. 71.1–71.12.

Ess, A., B. Leibe, K. Schindler, and L. J. V. Gool (2009). Moving obstacle detection in highly dynamic scenes. In *IEEE International Conference on Robotics and Automation (ICRA)*, Kobe, Japan, pp. 56–63.

EUROSTAT (2015). 230/2016, News Release. http://ec.europa.eu/eurostat/documents/2995521/7734698/7-18112016-BP-EN.pdf. 2017-03-28.

Felzenszwalb, P. F. and D. P. Huttenlocher (2004). Efficient graph-based image segmentation. *International Journal of Computer Vision (IJCV)*, 59:167–181.

Fleming, W. J. (2008). New automotive sensors: A review. *IEEE Sensors Journal*, 8:1900–1921.

Förstner, W. and B. P. Wrobel (2016). *Photogrammetric Computer Vision – Statistics, Geometry, Orientation and Reconstruction*. Springer.

Frahm, J.-M., P. F. Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-H. Jen, E. Dunn, B. Clipp, and S. Lazebnik (2010). Building rome on a cloudless day. In *European Conference on Computer Vision (ECCV)*, Heraklion, Greece, pp. 368–381.

Franke, U., S. Mehring, A. Suissa, and S. Hahn (1994). The daimler-benz steering assistant: a spin-off from autonomous driving. In *IEEE Intelligent Vehicles Symposium (IV)*, Paris, France, pp. 79–85.

Franke, U., D. Pfeiffer, C. Rabe, C. Knöppel, M. Enzweiler, F. Stein, and R. G. Herrtwich (2013). Making bertha see. In *International Conference on Computer Vision (ICCV), Workshop*, Sydney, Australia, pp. 214–221.

Franke, U., C. Rabe, H. Badino, and S. Gehrig (2005). 6d-vision: Fusion of stereo and motion for robust environment perception. In *German Association for Pattern Recognition (DAGM)*, Vienna, Austria, pp. 216–223.

Gehrig, S., F. Eberli, and T. Meyer (2009). A real-time low-power stereo vision engine using semi-global matching. In *International Conference on Computer Vision Systems (ICVS)*, pp. 134–143.

Gehrig, S. and C. Rabe (2010). Real-Time Semi-Global Matching on the CPU. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Workshop*, San Francisco, USA, pp. 85–92.

Geiger, A., P. Lenz, C. Stiller, and R. Urtasun (2013). Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 1231–1237.

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

Girshick, R. (2015). Fast r-cnn. In *International Conference on Computer Vision (ICCV)*, pp. 1440–1448.

Google (2010). Google. what we're driving at. https://googleblog.blogspot.de/2010/10/what-were-driving-at.html. 2017-03-30.

Grisetti, G., C. Stachniss, and W. Burgard (2007). Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, 23:34–46.

Hähnel, D., W. Burgard, D. Fox, and S. Thrun (2003). An efficient fastslam algorithm for generating cyclic maps of large-scale environments from raw laser range measurements. In *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, Las Vegas, USA, pp. 206–211.

Hähnel, D., D. Schulz, and W. Burgard (2002). Map building with mobile robots in populated environments. In *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, Lausanne, Switzerland, pp. 496–501.

Haller, I. and S. Nedevschi (2010). Gpu optimization of the sgm stereo algorithm. In *IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, Washington, USA, pp. 197–202.

Hammersley, J. M. and P. E. Clifford (1971). Markov random fields on finite graphs and lattices. Unpublished manuscript.

Hartley, R. I. and A. Zisserman (2004). *Multiple View Geometry in Computer Vision*. Cambridge University Press.

Hermes, C., J. Einhaus, M. Hahn, C. Wöhler, and F. Kummert (2010, June). Vehicle tracking and motion prediction in complex urban scenarios. In *IEEE Intelligent Vehicles Symposium (IV)*, San Diego, USA, pp. 26–33.

Hirschmüller, H. (2005, June). Accurate and efficient stereo processing by semi-global matching and mutual information. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Diego, USA, pp. 807–814.

Hirschmüller, H. (2008). Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 30:328–341.

Homm, F., N. Kaempchen, J. Ota, and D. Burschka (2010, June). Efficient occupancy grid computation on the gpu with lidar and radar for road boundary detection. In *IEEE Intelligent Vehicles Symposium (IV)*, San Diego, USA, pp. 1006–1013.

Hornung, A., K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard (2013). Octomap: an efficient probabilistic 3d mapping framework based on octrees. *Autonomous Robots*, 34:189–206.

iMAR (2017). Navigation and control. http://www.imar-navigation.de/index.php/de/. 2017-05-01.

Kaess, M., A. Ranganathan, and F. Dellaert (2008). isam: Incremental smoothing and mapping. *IEEE Transactions on Robotics*, 24:1365–1378.

Kammel, S., J. Ziegler, B. Pitzer, M. Werling, T. Gindele, D. Jagzent, J. Schröder, M. Thuy, M. Goebl, F. v. Hundelshausen, O. Pink, C. Frese, and C. Stiller (2008). Team annieway's autonomous system for the 2007 darpa urban challenge. *Journal of Field Robotics*, 25:615–639.

Kerl, C., J. Sturm, and D. Cremers (2013). Dense visual slam for rgb-d cameras. In *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, Tokyo, Japan, pp. 2100–2106.

Kohli, P. and P. H. S. Torr (2007). Dynamic graph cuts for efficient inference in markov random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29:2079–2088.

Kohli, P. and P. H. S. Torr (2008). Measuring uncertainty in graph cut solutions. *Journal on Computer Vision and Image Understanding*, 112:30–38.

Kolmogorov, V. and R. Zabih (2004). What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 26:147–159.

Konolige, K. and K. Chou (1999). Markov localization using correlation. In *International Joint Conference on Artificial Intelligence*, San Francisco, USA, pp. 1154–1159.

Kümmerle, R., G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard (2011). g2o: A general framework for graph optimization. In *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, pp. 3607–3613.

Kweon, I. S. and T. Kanade (1992). High-resolution terrain map from multiple sensor data. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 14:278–292.

L. R. Ford, J. and D. R. Fulkerson (1962). *Flows in Networks*. Princeton University Press.

Lacroix, S., I. kyun Jung, and A. Mallet (2002). Digital elevation map building from low altitude stereo imagery. *Robotics and Autonomous Systems*, 41:119–127.

Lafferty, J. D., A. McCallum, and F. C. N. Pereira (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning (ICML)*, San Francisco, USA, pp. 282–289.

Lategahn, H., T. Graf, C. Hasberg, B. Kitt, and J. Effertz (2011). Mapping in dynamic environments using stereo vision. In *IEEE Intelligent Vehicles Symposium (IV)*, Baden-Baden, Germany, pp. 150–156.

Lategahn, H. and C. Stiller (2014). Vision-only localization. *IEEE Transactions on Intelligent Transportation Systems*, 15:1246–1257.

Levinson, J. and S. Thrun (2010). Robust vehicle localization in urban environments using probabilistic maps. In *IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, USA, pp. 4372–4378.

Limketkai, B., R. Biswas, and S. Thrun (2002). Learning occupancy grids of non-stationary objects with mobile robots. In *Experimental Robotics VIII*, Sant'Angelo d'Ischia, Italy, pp. 222–231.

Lindgren, A. and F. Chen (2006). State of the art analysis: An overview of advanced driver assistance systems (adas) and possible human factors issues. In *Human factors and economics aspects on safety*, Linköping, Schweden, pp. 38–51.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60:91–110.

Lui, J. S. (1996). Metropolized independent sampling with comparisons to rejection sampling and importance sampling. *Statistics and Computing*, 6:113–119.

Mackay, D. J. C. (2003). *Information theory, inference, and learning algorithms.* Cambridge University Press.

Matthies, L. and A. E. Elfes (1988). Integration of sonar and stereo range data using a grid-based representation. In *IEEE International Conference on Robotics and Automation (ICRA)*, Philadelphia, USA, pp. 727–733.

McGlone, J., E. Mikhail, J. Bethel, and R. Mullen (2004). *Manual of Photogrammetry.* American Society for Photogrammetry and Remote Sensing.

Merali, R. and T. Barfoot (2013). Occupancy grid mapping with markov chain monte carlo gibbs sampling. In *IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, pp. 3183–3189.

Merali, R. S. and T. D. Barfoot (2012). Patch map: A benchmark for occupancy grid algorithm evaluation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, Algarve, Portugal.

Mercedes-Benz (2016a). Mercedes-Benz at the NAIAS. https://media.mbusa.com/releases/mercedes-benz-at-the-2016-naias?firstResultIndex=0&sortOrder=PublishedDescending/. 2017-03-30.

Mercedes-Benz (2016b). Mercedes-Benz. DISTRONIC PLUS. https://www.mbusa.com/mercedes/technology/videos/detail/title-claclass/videoId-caf758b451127410VgnVCM100000ccec1e35RCRD. 2017-03-30.

Montemerlo, M., S. Thrun, D. Koller, and B. Wegbreit (2002). Fastslam: A factored solution to the simultaneous localization and mapping problem. In *National Conference on Artificial Intelligence*, Edmonton, Canada, pp. 593–598.

Moravec, H. P. (1996). Robot spatial perception by stereoscopic vision and 3d evidence grids. Technical report, Carnegie Mellon University.

Moravec, H. P. and A. E. Elfes (1985). High resolution maps from wide angle sonar. In *IEEE International Conference on Robotics and Automation (ICRA)*, St. Louis, USA, pp. 116–121.

Muffert, M., S. Anzt, and U. Franke (2013). An incremental map building approach via static stixel integration. In *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Antalya, Turkey, pp. 55–60.

Muffert, M., D. Pfeiffer, and U. Franke (2013). A stereo-vision based object tracking approach at roundabouts. In *IEEE Intelligent Transportation Systems Magazine*, pp. 5:22–23.

Muffert, M., N. Schneider, and U. Franke (2014). Stix-fusion: A probabilistic stixel integration technique. In *Canadian Conference on Computer and Robot Vision (CRV)*, Montreal, Canada, pp. 16–23.

Munz, M., M. Mählisch, and K. Dietmayer (2010). Generic centralized multi sensor data fusion based on probabilistic sensor and environment models for driver assistance systems. In *IEEE Intelligent Transportation Systems Magazine*, pp. 2:6–17.

Murphy, K. P. (1999). Bayesian map learning in dynamic environments. In *Neural Information Processing Systems (NIPS)*, Denver, USA, pp. 1015–1021.

Nalpantidis, L., A. Gasteratos, and G. Sirakoulis (2008). Review of stereo vision algorithms: From software to hardware. *International Journal of Optomechatronics*, 2:435–462.

Nuss, D., T. Yuan, G. Krehl, M. Stübler, S. Reuter, and K. Dietmayer (2015). Fusion of laser and radar sensor data with a sequential monte carlo bayesian occupancy filter. In *IEEE Intelligent Vehicles Symposium (IV)*, Seoul, South Kore, pp. 1074–1081.

O'Callaghan, S. T. and F. T. Ramos (2012). Gaussian process occupancy maps. *I. J. Robotics Res.*, 31:42–62.

O'Callaghan, S. T. and F. T. Ramos (2014). Gaussian process occupancy maps for dynamic environments. In *nternational Symposium on Experimental Robotics, ISER*, Marrakech and Essaouira, Morocco, pp. 791–805.

Perrollaz, M., J.-D. Yoder, A. Spalanzani, and C. Laugier (2010). Using the disparity space to compute occupancy grids from stereo-vision. In *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, Taipei, Taiwan, pp. 2721–2726.

Pfeiffer, D. (2011). *The Stixel World: a compact medium-level represantation for efficiently modeling dynamic three-dimensional environments*. Ph. D. thesis, Mathematisch-Naturwissenschaftliche Fakultät Humboldt-Universität, Berlin, Germany.

Pfeiffer, D. and U. Franke (2010). Efficient representation of traffic scenes by means of dynamic Stixels. In *IEEE Intelligent Vehicles Symposium (IV)*, San Diego, USA, pp. 217–224.

Pfeiffer, D. and U. Franke (2011). Towards a global optimal multi-layer stixel representation of dense 3D data. In *British Machine Vision Conference (BMVC)*, Dundee, Scotland, pp. 51.1–51.12.

Pfeiffer, D., S. Gehrig, and N. Schneider (2013). Exploiting the power of stereo confidences. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Portland, USA, pp. 297–304.

Pfeiffer, D., S. Morales, A. Barth, and U. Franke (2010). Ground truth evaluation of the stixel representation using laser scanners. In *IEEE Conference on Intelligent Transportation Systems (ITSC)*, Maideira Island, Portugal, pp. 1091–1097.

Potts, R. B. (1952). Some generalized order-disorder transformations. *Mathematical Proceedings of the Cambridge Philosophical Society*, 48:106–109.

Rabe, J., M. Necker, and C. Stiller (2016). Ego-lane estimation for lane-level navigation in urban scenarios. In *IEEE Intelligent Vehicles Symposium (IV)*, Gotenburg,Sweden, pp. 896–901.

Rapp, M., K. Dietmayer, M. Hahn, and B. D. andJürgen Dickmann (2016). Hidden markov model-based occupancy grid maps of dynamic environments. In *International Conference on Information Fusion (FUSION)*, Heidelberg, Germany, pp. 1780–1788.

Roewekaemper, J., C. Sprunk, G. Tipaldi, C. Stachniss, P. Pfaff, and W. Burgard (2012). On the position accuracy of mobile robot localization based on particle filters combined with scan matching. In *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, Vilamoura, Portugal, pp. 3158–3164.

Sameer Agarwal, Keir Mierle, O. (2016). Ceres solver. `http://ceres-solver.org/`. 2017-04-28.

Santos, J., T. Krajník, J. P. Fentanes, and T. Duckett (2016). Lifelong information-driven exploration to complete and refine 4d spatio-temporal maps. *IEEE Robotics and Automation Letters*, 1:684–691.

Scharstein, D. and R. Szeliski (2002). Middlebury online stereo evaluation. `http://vision.middlebury.edu/stereo`. 2017-05-28.

Scharwächter, T. (2013). Stixel-based target existence estimation under adverse conditions. In *German Conference on Pattern Recognition (GCPR)*, Graz, Austria, pp. 225–230.

Scharwächter, T., M. Enzweiler, U. Franke, and S. Roth (2013). Efficient multi-cue scene segmentation. In *German Conference on Pattern Recognition (GCPR)*, Saarbrücken, Germany, pp. 435–445.

Scharwächter, T. and U. Franke (2015). Low-level fusion of color, texture and depth for robust road scene understanding. In *IEEE Intelligent Vehicles Symposium (IV)*, Seoul, South Korea, pp. 599–604.

Schmid, M., M. Maehlisch, J. Dickmann, and H.-J. Wünsche (2010, June). Dynamic level of detail 3d occupancy grids for automotive use. In *IEEE Intelligent Vehicles Symposium (IV)*, San Diego, USA, pp. 269–274.

Schneider, L., M. Cordts, T. Rehfeld, D. Pfeiffer, M. Enzweiler, U. Franke, M. Pollefeys, and S. Roth (2016). Semantic stixels: Depth is not enough. In *IEEE Intelligent Vehicles Symposium (IV)*, Gotenburg, Sweden, pp. 110–117.

Schreiber, M., C. Knöppel, and U. Franke (2013). Laneloc: Lane marking based localization using highly accurate maps. In *IEEE Intelligent Vehicles Symposium (IV)*, Gold Coast City, Australia, pp. 449–454.

Schröter, C., H. Böhme, and H. Gross (2007). Memory-efficient gridmaps in rao-blackwellized particle filters for slam using sonar range sensors. In *European Conference on Mobile Robots (EMCR)*, Freiburg, Germany, pp. 143–149.

Schwing, A. G. and R. Urtasun (2015). Fully connected deep structured networks. *Computing Research Repository(CoRR) abs/1503.02351*.

Senanayake, R., L. Ott, S. T. O'Callaghan, and F. T. Ramos (2016). Spatio-temporal hilbert maps for continuous occupancy representation in dynamic environments. In *Annual Conference on Neural Information Processing Systems*, Barcelona, Spain, pp. 3918–3926.

Shelhamer, E., J. Long, and T. Darrell (2016). Fully convolutional networks for semantic segmentation. *Computing Research Repository(CoRR) abs/1605.06211*.

Siegemund, J. (2013). *Street Surfaces and Boundaries from Depth Image Sequences using Probabilistic Models*. Ph. D. thesis, Institut für Geodäsie und Geoinformation (IGG), Bonn, Germany.

Smith, R. C. and P. Cheeseman (1986). On the representation and estimation of spatial uncertainly. *International Journal of Robotics Research (IJRR)*, 5:56–68.

Stachniss, C. (2006). *Exploration and Mapping with Mobile Robots*. Ph. D. thesis, Department of Computer Science, Freiburg, Germany.

Stachniss, C. and W. Burgard (2005). Mobile robot mapping and localization in non-static environments. In *National Conference on Artificial Intelligence (AAAI)*, Pittsburgh, USA, pp. 1324–1329.

Stein, F. (2012). The challenge of putting vision algorithms into a car. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Workshop*, pp. 89–94.

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

Thrun, S., W. Burgard, and D. Fox (2005). *Probabilistic Robotics*. The MIT Press.

Thrun, S. and M. Montemerlo (2006). The graph slam algorithm with applications to large-scale mapping of urban structures. *International Journal of Robotics Research (IJRR)*, 25:403–429.

Thrun, S., M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney (2006). Stanley: The robot that won the DARPA Grand Challenge: Research articles. *Journal of Field Robotics*, 23:661–692.

Tomasi, C. and T. Kanade (1991). Detection and tracking of point features. Technical report, School of Computer Science, Carnegie Mellon University.

Torabi, L., M. Kazemi, and K. K. Gupta (2007). Configuration space based efficient view planning and exploration with occupancy grids. In *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, San Diego, USA, pp. 2827–2832.

TOYOTA (2017). Lane keeping assist. helps keep drivers within lanes. http://www.toyota-global.com/innovation/safety_technology/safety_technology/technology_file/active/lka.html. 2017-03-30.

van der Merwe, R., A. Doucet, N. de Freitas, and E. Wan (2000). The unscented particle filter. In *Neural Information Processing Systems (NIPS)*, Denver, Colorado, pp. 584–590.

Čikeš, M., M. Seder, and I. Petrović (2011). The path planning algorithms for a mobile robot based on the occupancy grid map of the environment- a comparative study. In *International Symposium on Information, Communication and Automation Technologies (ICAT)*, Sarajevo, Bosnia and Herzegovina, pp. 1–8.

Veksler, O., Y. Boykov, and P. Mehrani (2010). Superpixels and supervoxels in an energy optimization framework. In *European Conference on Computer Vision (ECCV)*, Heraklion, Greece, pp. 211–224. Springer-Verlag.

Velodyne (2010). High definition lidar hdl-64e s2. http://www.velodyne.com/lidar/. 2017-05-28.

VisLab (2013). PROUD-Car Test 2013. Public ROad Urban Driverless-Car Test 2013 -. http://vislab.it/proud/. 2017-03-30.

VOLKSWAGEN (2017). Volkswagen. automatic distance control acc. http://en.volkswagen.com/en/innovation-and-technology/technical-glossary/automatische_distanzregelung_acc.html. 2017-03-30.

WAYMO (2016). WAYMO. We're building a safer driver for everyone. https://waymo.com/. 2017-04-08.

Wedel, A., A. Meissner, C. Rabe, U. Franke, and D. Cremers (2009). Detection and segmentation of independently moving objects from dense scene flow. In *Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, Bonn, Germany, pp. 14–27.

Welch, G. and G. Bishop (1995). An introduction to the kalman filter. Technical report, Department of Computer Science, University of North Carolina at Chapel Hill.

Werber, K., M. Rapp, J. Klappstein, M. Hahn, J. Dickmann, K. Dietmayer, and C. Waldschmidt (2013). Automotive radar gridmap representations. In *International Conference on Microwaves for Intelligent Mobility (ICMIM)*, Karlsruhe, Germany, pp. 3183–3189.

Wojek, C. and B. Schiele (2008). A dynamic conditional random field model for joint labeling of object and scene classes. In *European Conference on Computer Vision (ECCV)*, Marseille, France, pp. 733–747.

Yguel, M., O. Aycard, and C. Laugier (2006). Efficient gpu-based construction of occupancy grids using several laser range-finders. In *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, Beijing, China, pp. 105–110.

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22:1330–1334.

Ziegler, J., P. Bender, M. Schreiber, H. Lategahn, T. Strauss, C. Stiller, T. Dang, U. Franke, N. Appenrodt, C. G. Keller, E. Kaus, R. G. Herrtwich, C. Rabe, D. Pfeiffer, F. Lindner, F. Stein, F. Erbs, M. Enzweiler, C. Knöppel, J. Hipp, M. Haueis, M. Trepte, C. Brenk, A. Tamke, M. Ghanaat, M. Braun, A. Joos, H. Fritz, H. Mock, M. Hein, and E. Zeeb (2014). Making bertha drive - an autonomous journey on a historic route. *IEEE Intelligent Transportation Systems Magazine*, 6:8–20.

Ziegler, J., H. Lategahn, M. Schreiber, C. G. Keller, C. Knöppel, J. Hipp, M. Haueis, and C. Stiller (2014). Video based localization for bertha. In *IEEE Intelligent Vehicles Symposium (IV)*, Dearborn, USA, pp. 1231–1238.