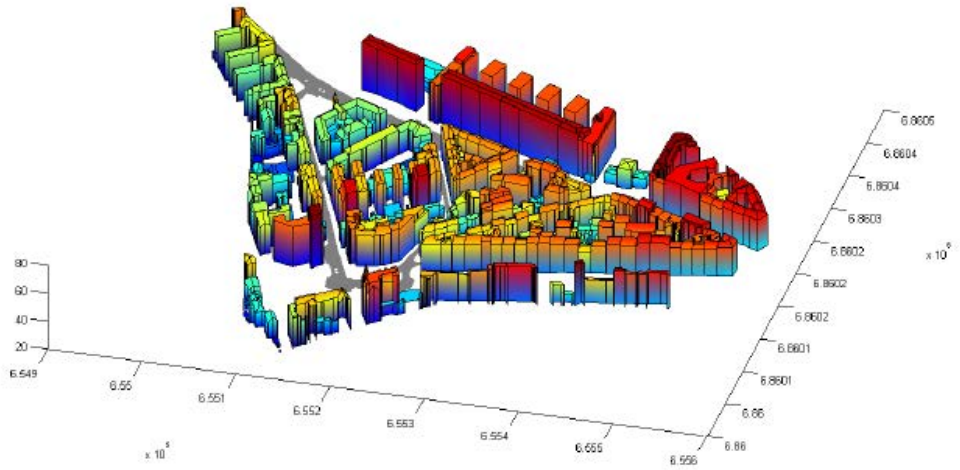
Exploitation of map data for the perception of intelligent vehicles

Marek Kurdej

Par **Marek KURDEJ**

Exploitation of map data for the perception of intelligent vehicles

Thèse présentée
pour l'obtention du grade
de Docteur de l'UTC



Soutenue le 05 février 2015
Spécialité : Technologies de l'Information et des Systèmes

D2174

Exploitation des données cartographiques pour la perception de véhicules intelligents

Marek KURDEJ

Soutenue publiquement le 5 février 2015 devant le jury composé de :

Rapporteurs *:*	
Sylvie Le Hégarat-Mascle	Professeur des Universités Université Paris-Sud Orsay, IEF, Paris
Olivier Aycard	Maître de Conférences (HDR) Université Grenoble 1 (UJF), LIG/AMA, Grenoble
Président *:*	
Éric Lefevre	Professeur des Universités Université d'Artois
Examinateurs *:*	
Philippe Bonnifait	Professeur des Universités UTC, Heudiasyc, Compiègne
Bahman Soheilian	Chargé de Recherche IGN, Matis, Saint-Mandé
Directrice de thèse *:*	
Véronique Cherfaoui	Maître de Conférences (HDR) UTC, Heudiasyc, Compiègne

---

Université de Technologie de Compiègne
Laboratoire Heudiasyc UMR CNRS 7253
5 février 2015

To cite this thesis, please use:

BibTeX entry:

```
@phdthesis{Kurdej2015phd,
    author = {Kurdej, Marek},
   title = {{Exploitation of map data for the perception of intelligent vehicles}},
    year = {2015},
    month = {Feb},
   keywords = {vector maps, perception, obstacle detection, autonomous vehicles,
intelligent cars, evidential occupancy grids, belief functions theory},
    language = {en},
    school = {Universit\'{e} de Technologie de Compi\`{e}gne},
    type = {Ph.D. thesis}
}
```

Classic citation:

Kurdej, Marek: *Exploitation of map data for the perception of intelligent vehicles*, Ph.D. thesis. Heudiasyc, Université de Technologie de Compiègne, France; February 2015.

Dla Taty, mojego anioła stróża.

To Dad, my guardian angel.

[This page intentionally left blank.]

Acknowledgements

First of all, I am very grateful to my advisor, Véronique Cherfaoui. I thank her for the scientific help, support and guidance during this Ph.D. that allowed me to enlarge my horizons. Apart from scientific aspects, she proved to possess loads of patience and human sensibility that helped me to overcome the most difficult problems and to go smoothly through the ups and downs of the thesis.

I would like to thank my colleagues from Heudiasyc laboratory with which I had the chance and pleasure to collaborate, namely Philippe Bonnifait, Reine Talj and Sébastien Destercke. I appreciated your help and remarks as well as the discussions we have had. And of course, I want to express my gratitude to the members of the jury that participated in my Ph.D. defense. Especially, I would like to acknowledge the efforts of the referees of my thesis, Sylvie Le Hégarat and Olivier Aycard. Thank you for the time you devoted and the feedback that you provided.

I cannot forget that this thesis would have been possible without the grant from the French Ministry of Defence DGA (Direction Générale de l'Armement) to whom I owe all the respect. I also would like to thank the participants of the CityVIP project ANR-07_TSFA-013-01 from which comes a part of the dataset that was used for testing studied approaches.

A big thank you goes to all the people that I have met at UTC and Heudiasyc laboratory and especially to the friends I have made during this 3-year-long journey. Felipe, with whom I shared the office, and whose questions nourished my reflection and whose questioning way of thinking (and being) gave me a fresh new look onto the science. Thank you for your wit and good humour that brought the joy into the monotony of work that happened to arrive from time to time. Thanks to Nicole and Farah who were there bringing smile and delicious dishes. I am really grateful to Julien for all the collaboration. My thanks go as well to Clément, Adam, Philippe and many others whose names I would not cite as I would certainly omit somebody.

Thank you to my family, my Mum, my brothers Damian and Jacek who always were and are a support for me in whatever adventure I launched myself throughout my life. I thank as well my nephews, Bianka, Weronika and Franek, for all the joy that they were source of.

Last but not least, my thanks go to my wife, Marion, who constantly gave me words of encouragement and inspired me all along the duration of the thesis, and to my daughter, Ofelia, who (just as her mother) is the light of my life.

20 February 2015

[This page intentionally left blank.]

Abstract

Exploitation of map data for the perception of intelligent vehicles

This thesis is situated in the domains of robotics and data fusion, and concerns geographic information systems. We study the utility of adding digital maps, which model the urban environment in which the vehicle evolves, as a virtual sensor improving the perception results.

Indeed, the maps contain a phenomenal quantity of information about the environment: its geometry, topology and additional contextual information. In this work, we extract road surface geometry and building models in order to deduce the context and the characteristics of each detected object.

Our method is based on an extension of occupancy grids: the evidential perception grids. It permits to model explicitly the uncertainty related to the map and sensor data. By this means, the approach presents also the advantage of representing homogeneously the data originating from various sources: lidar, camera or maps. The maps are handled on equal terms with the physical sensors. This approach allows us to add geographic information without imputing unduly importance to it, which is essential in presence of errors.

In our approach, the information fusion result, stored in a perception grid, is used to predict the state of environment on the next instant. The fact of estimating the characteristics of dynamic elements does not satisfy the hypothesis of static world. Therefore, it is necessary to adjust the level of certainty attributed to these pieces of information. We do so by applying the temporal discounting. Due to the fact that existing methods are not well suited for this application, we propose a family of discount operators that take into account the type of handled information.

The studied algorithms have been validated through tests on real data. We have thus developed the prototypes in Matlab and the C++ software based on Pacpus framework. Thanks to them, we present the results of experiments performed in real conditions.

Keywords: vector maps, perception, obstacle detection, autonomous vehicles, intelligent cars, evidential occupancy grids, belief functions theory.

[This page intentionally left blank.]

Résumé

Exploitation des données cartographiques pour la perception de véhicules intelligents

La plupart des logiciels contrôlant les véhicules intelligents traite de la compréhension de la scène. De nombreuses méthodes existent actuellement pour percevoir les obstacles de façon automatique. La majorité d'entre elles emploie ainsi les capteurs extéroceptifs comme des caméras ou des lidars. Cette thèse porte sur les domaines de la robotique et de la fusion d'information et s'intéresse aux systèmes d'information géographique. Nous étudions ainsi l'utilité d'ajouter des cartes numériques, qui cartographient le milieu urbain dans lequel évolue le véhicule, en tant que capteur virtuel améliorant les résultats de perception.

Les cartes contiennent en effet une quantité phénoménale d'information sur l'environnement : sa géométrie, sa topologie ainsi que d'autres informations contextuelles. Dans nos travaux, nous avons extrait la géométrie des routes et des modèles de bâtiments afin de déduire le contexte et les caractéristiques de chaque objet détecté.

Notre méthode se base sur une extension de grilles d'occupations : les grilles de perception crédibilistes. Elle permet de modéliser explicitement les incertitudes liées aux données de cartes et de capteurs. Elle présente également l'avantage de représenter de façon uniforme les données provenant de différentes sources : lidar, caméra ou cartes. Les cartes sont traitées de la même façon que les capteurs physiques. Cette démarche permet d'ajouter les informations géographiques sans pour autant leur donner trop d'importance, ce qui est essentiel en présence d'erreurs.

Dans notre approche, le résultat de la fusion d'information contenu dans une grille de perception est utilisé pour prédire l'état de l'environnement à l'instant suivant. Le fait d'estimer les caractéristiques des éléments dynamiques ne satisfait donc plus l'hypothèse du monde statique. Par conséquent, il est nécessaire d'ajuster le niveau de certitude attribué à ces informations. Nous y parvenons en appliquant l'affaiblissement temporel. Étant donné que les méthodes existantes n'étaient pas adaptées à cette application, nous proposons une famille d'opérateurs d'affaiblissement prenant en compte le type d'information traitée.

Les algorithmes étudiés ont été validés par des tests sur des données réelles. Nous avons donc développé des prototypes en Matlab et des logiciels en C++ basés sur la plate-forme Pacpus. Grâce à eux nous présentons les résultats des expériences effectués en conditions réelles.

Mots-clés : cartes vectorielles, perception, détection d'obstacles, véhicules autonomes, voitures intelligentes, grilles d'occupation évidentielles, théorie des fonctions de croyance.

[This page intentionally left blank.]

Streszczenie

Wykorzystanie map w systemach percepcji pojazdów inteligentnych

Większość systemów kontrolujących pojazdy inteligentne dotyczy problemów zrozumienia sceny. Istnieje obecnie wiele metod pozwalających na automatyczne rozpoznawanie przeszkód. Większość z nich korzysta z czujników zewnętrznych takich jak kamery czy lidary. Niniejsza praca doktorska jest usytuowana pomiędzy robotyką a fuzją informacji, ale dotyczy również systemów informacji geograficznej. W naszej pracy badamy użyteczność map cyfrowych (modelujących środowisko miejskie w którym porusza się pojazd) zastosowanych jako czujnik wirtualny w celu polepszenia jakości percepji.

Mapy zawierają albowiem niezliczoną ilość informacji na temat środowiska: jego geometrię, topologię czy też inne informacje kontekstowe. W naszych badaniach wykorzystaliśmy geometrię dróg oraz modele budynków, aby odgadnąć kontekst i charakterystykę rozpoznanych obiektów.

Proponowana metoda opiera się na ewidencyjnych siatkach percepcji (ang. *evidential perception grids*) będących rozszerzeniem siatek zajętości (ang. *occupancy grids*). Pozwala ona na odwzorowanie niedokładności danych map oraz czujników. Inną korzyścią jest fakt, iż dane pochodzące z różnorakich źródeł, np. lidaru, kamery czy map, są reprezentowane w sposób jednorodny. Mapy są w dodatku używane w ten sam sposób co czujniki fizyczne. Takie rozwiązanie pozwala na dodanie informacji geograficznej bez nadania jej zbyt dużej ważności, co jest konieczne w razie występowanie błędów.

W naszej metodzie, wynik fuzji informacji przechowywany w siatkach percepcji jest używany do przewidywania stanu środowiska w następnym momencie. Przewidywanie właściwości elementów dynamicznych nie spełnia więc hipotezy świata statycznego. Wynika z tego, że niezbędne jest dopasowanie poziomu pewności przypisanego danej informacji. Wykonaliśmy to dzięki zastosowaniu czasowego obniżania wartości informacji. Ze względu na fakt, iż istniejące metody nie są dostosowane do takiego zastosowania, zaproponowaliśmy rodzinę operatorów, które biorą pod uwagę typ przetwarzanej informacji.

Badane algorytmy zostały potwierdzone przez testy przeprowadzone na danych niesymulowanych. Zaimplementowaliśmy w tym celu prototypy wykorzystując język Matlab oraz oprogramowanie działające w czasie rzeczywistym oparte na platformie Pacpus. Dzięki temu przedstawiamy wyniki tych testów w warunkach naturalnych.

Słowa kluczowe: mapy wektorowe, percepcja, rozpoznawanie przeszkód, pojazdy autonomiczne, samochody inteligentne, ewidencyjne siatki zajętości, teoria ewidencji.

[This page intentionally left blank.]

Contents

[This page intentionally left blank.]

Acronyms

ADAS Advanced Driver Assistance System . . . 143
API application programming interface . . . 19
AUTOSAR AUTomotive Open System ARchitecture . . . 142
BOF Bayesian Occupancy Filter . . . 55
BFT belief functions theory . . . 129
CityGML City Geography Markup Language . . . 141
CRC conjunctive rule of combination . . . 44
CUDA Compute Unified Device Architecture . . . 143
DARPA Defense Advanced Research Projects Agency . . . 4
DATMO detection and tracking of moving objects . . . 55
DEM Digital Elevation Model . . . 22
DR dead reckoning . . . 17
DRC disjunctive rule of combination . . . 44
DST Dempster–Shafer theory . . . 59
DTM Digital Terrain Model . . . 64
EGNOS European Geostationary Navigation Overlay Service . . . 26
EKF Extended Kalman Filter . . . 30
EM Expectation-Maximization . . . 56
fod frame of discernment . . . 90
GG `GISGrid` . . . 125
GIS Geographical Information System . . . 22
GLONASS Глобальная навигационная спутниковая система, Russian for: GNSS . . . 25
GML Geography Markup Language . . . 141
GNSS Global Navigation Satellite System . . . 132
GPGPU General-Purpose computing on Graphics Processing Units . . . 135
GPS Global Positioning System . . . 131
GPU Graphics Processing Unit
GSBBA generalised simple basic belief assignment . . . 43
GUI Graphical User Interface . . . 142
IGN French National Institute of the Geographic and Forest Information . . . 141
IMU Inertial Measurement Unit . . . 131
INS Inertial Navigation System . . . 142
ITS Intelligent Transportation System
LIDAR LIght Detection And Ranging . . . 53
LOD level of detail . . . 48
LRR long-range radar . . . 8
MOT moving object tracking . . . 55

MMS Mobile Mapping System . . . 20
OSM OpenStreetMap . . . 141
OSM3S Overpass API . . . 19
PACPUS Perception et Assistance pour une Conduite Plus Sure . . . 143
PAMU Plateforme Avancée de Mobilité Urbaine . . . 4
PCL Point-Cloud Library . . . 144
PG `PerceptionGrid` . . . 122
POI point of interest . . . 18
RANSAC RANdom SAmple Consensus . . . 11
RF Radio Frequency . . . 25
ROS Robot Operating System . . . 144
RSF Road Structural Feature . . . 26
SG `SensorGrid` . . . 57
SIFT Scale Invariant Feature Transform . . . 26
SLAM Simultaneous Localization and Mapping . . . 57
SLAMMOT Simultaneous Localization, Mapping and Moving Object Tracking . . . 77
SNR signal-to-noise ratio . . . 23
SoG `SourceGrid` . . . 79
SRR short-range radar . . . 8
SRTM Shuttle Radar Topography Mission . . . 23
TBM Transferable Belief Model . . . 43
UKF Unscented Kalman Filter . . . 22
V2I Vehicle-To-Infrastructure . . . 4
V2V Vehicle-To-Vehicle . . . 4
VO visual odometry . . . 26
XAPI eXtended API . . . 19

Notation

In the following chapters, we will stick to the notation described below.

Mass functions The set of possible hypotheses, called frame of discernment (fod), will be designated by capital Greek letter omega Ω, with a subscript if necessary, e.g., Ω_1. A basic belief assignment (bba) defined on a fod Ω obtained from source S will be noted m^{Ω}_{S}. When no ambiguity is possible, the fod Ω will be omitted and so the equivalent notation will be m_S. In order to denote the mass attributed to a given hypothesis A, we will use the notation $\mathrm{m}^{\Omega}_{S}(A)$, which will be usually simplified to $\mathrm{m}(A)$.

Evidential grids The notation $\mathrm{m}\{X,\, Y\}$ will denote the mass function contained in the cell situated at position $\{X,\, Y\}$, i.e. the one covering the box $\{X,\, Y\} = \{[x^-,\, x^+]\,,\, [y^-,\, y^+]\}$. Often, if the same fusion operation is applied to all cells, the cell position will be omitted and the simplified notation will be used, e.g., instead of writing $\mathrm{m}\{X,\, Y\}(A)$, we will simply say $\mathrm{m}(A)$.

Discounting In order to distinguish various discounting types and to avoid any confusion, ${}^{\alpha}\mathrm{m}$ will denote uniform (classical) discounting with decay factor α. We will refer to Mercier's contextual discounting using ${}^{\vec{\alpha}}\mathrm{m}$ notation, where $\vec{\alpha}$ will represent the vector of discount factors. Furthermore, ${}^{\vec{\alpha}}_{c,\,\Theta}\mathrm{m}$ will denote conservative discounting of a bba m using discount rate vector $\vec{\alpha}$ defined for all elements of $\Theta \subseteq 2^{\Omega}$. Similarly, ${}^{\vec{\alpha}}_{p,\,\Theta}\mathrm{m}$ will represent proportional discounting and ${}^{\vec{\alpha}}_{o,\,\Theta}\mathrm{m}$ — optimistic discounting. For contextual discounting operations, α_θ will refer to the discount rate defined for set θ, given that $\theta \in \Theta$.When the set of classes Θ for which discount factors are defined is obvious or unimportant, notation ${}^{\vec{\alpha}}_{c}\mathrm{m}$ will be equivalent to ${}^{\vec{\alpha}}_{c,\,\Theta}\mathrm{m}$. Analogical convention will be used for other types of discounting.

[This page intentionally left blank.]

Author's publications

Kurdej, M. and Bonnifait, P. *Intelligent Transportation Systems (ITS) Society Podcast.*

Kurdej, M. and Cherfaoui, V. (2013). "Conservative, Proportional and Optimistic Contextual Discounting in the Belief Functions Theory". *International Conference on Information Fusion*. C. Istanbul.

Kurdej, M., Moras, J., Cherfaoui, V., and Bonnifait, P. (2012). "Map-aided Fusion Using Evidential Grids for Mobile Perception in Urban Environment". *International Conference on Belief Functions*. Compiègne: Springer, pp. 343–350.

Kurdej, M., Moras, J., Cherfaoui, V., and Bonnifait, P. (2013). "Enhancing Mobile Object Classification Using Geo-referenced Maps and Evidential Grids". *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) Workshop on Planning, Perception and Navigation for Intelligent Vehicles*. Tokyo.

Kurdej, M., Moras, J., Cherfaoui, V., and Bonnifait, P. (2014). "Controlling Remanence in Evidential Grids Using Geodata for Dynamic Scene Perception". *International Journal of Approximate Reasoning (IJAR)* 55.1, pp. 355–375.

Kurdej, M., Moras, J., Cherfaoui, V., and Bonnifait, P. (2015). "Map-aided Evidential Grids for Driving Scene Understanding". *IEEE Intelligent Transportation Systems Magazine (ITS Mag)*, pp. 30–41.

[This page intentionally left blank.]

Part I
Preliminaries

The aim of this part is to present the subject of this dissertation. It serves as a starting point and problem statement. After having read this chapter, the reader should be acquainted with the aims that were set out for this thesis and have some insight into the needs, motivations, conditions and limitations that influenced the current work. The author wanted as well to rouse the reader's interest in the domain of intelligent vehicles and information fusion.

[This page intentionally left blank.]

Chapter 1

General introduction

"There are limits to the power of reason."

Peter Walley, *Statistical Reasoning with Imprecise Probabilities*

Contents

1.1 Context

The automation in the current technophilic world proceeds at a whopping pace in all domains. Yesterday's dreams slowly become today's reality. The car industry seems to be only one step away from delivering an intelligent self-driven vehicle to the mass public.

But before plunging into this vast and fascinating area, we will try to define what is the actual signification of intelligent vehicles. Leaving aside the general meaning of the term "intelligent" concerning the ability to learn, understand and reason, the sense of this word changes in the context of machines. In this respect, the term "intelligent" is defined by Webster's Dictionary as "guided or controlled by a computer; especially using a built-in microprocessor for automatic operation, for processing of data, or for achieving greater versatility"[1].

The above definition seems more adapted to intelligent vehicles. A more general and technically pragmatic definition of an intelligent vehicle can be found in the *Handbook of Intelligent Vehicles* (Eskandarian 2012). According to the authors, an intelligent vehicle performs certain aspects of driving either autonomously or assists the driver to perform his or her driving functions more effectively, all resulting in enhanced safety, efficiency, and environmental impact. In juxtaposition, the adjective "autonomous" implies that a vehicle has the intelligence to carry out a task, like driving or parking, without human guidance. Just as a remark, we have to note that a similar notion, smart vehicles, has recently become popular. This term is usually used to denote cars that communicate with other

[1]Merriam-Webster's Dictionary, web-site: http://www.merriam-webster.com/.

vehicles or with the dedicated infrastructure, often referred to as Vehicle-To-Vehicle (V2V) and Vehicle-To-Infrastructure (V2I) communication, respectively.

Google Car, Mercedes-Benz Bertha drive, the participants of Defense Advanced Research Projects Agency (DARPA) challenges or Chinese annual intelligent vehicle competitions, VisLab team and other entities proved that the autonomous car is almost ready. More and more companies announce working on similar projects, which shows that major advances are to be expected. For example, Baidu, the Chinese company known essentially for their search engine, works on a partially self-driving cars[1]. Also European car producers, as already mentioned German Mercedes-Benz, but also French companies, PSA Peugeot-Citroën and Renault, want to put out an intelligent car on the market by as early as 2018 or 2020[2].

Evidently, the advent of the next generation of intelligent transportation systems will alleviate many problems that haunt large agglomerations nowadays:

- Air contamination.
- Health problems caused by air pollution.
- Noise pollution and provoked discomfort.
- Limited number of parking places.
- Saturated road networks.

Obviously, solutions to these problems will not be found instantly, but intelligent vehicles will help to solve them at least partially. It could be done by:

- Reducing the number of accidents[3].
- Optimising traffic flow and thus increasing road network throughput.
- Decreasing fuel usage and pollution, diminishing impact on environment.
- Facilitating car sharing (e.g., by automated taxi-like services)[4].
- Cooperating driving and exchanging relevant information during the drive.

The aforementioned DARPA Grand Challenges are prize competitions organised by American Department of Defense. Their objective is to accelerate development of the technological foundations for autonomous vehicles. The first DARPA Grand Challenge started at the break of dawn on March 13, 2004. Unfortunately, no participant reached the finish line, even worse, the leader made only 12 km out of 230 km — a clear indication of the state of the technology at the moment. Since then, several other challenges took place, equally in the domain of autonomous driving as well as in other fields, e.g., robotics. All these competitions have had a tremendous impact on the domain. Already the first participants of DARPA 2005 challenge has proved that there are many highly valuable research works stemming from the competition (Broggi, Caraffi, et al. 2006; Buehler, Iagnemma, and Singh 2007). Next

[1] Please see http://thenextweb.com/asia/2014/07/25/chinas-baidu-follows-in-googles-footsteps-as-it-reveals-its-working-on-partial-self-drive-cars/ for details.

[2] More details in http://www.lesechos.fr/industrie-services/automobile/0203539871707-les-voitures-autonomes-de-renault-sur-les-routes-des-2018-1009052.php.

[3] Currently, human error is a major cause of mortality in road accidents, hence it can be eliminated thanks to the automated driving. Some studies mention that human factor accounts for 90% of road accidents, others speak of at least 80%. According to other surveys, the use of safety systems account for 30% to 40% decrease in road accident fatalities (S. Lefèvre, Laugier, and Ibañez-Guzmán 2011). Cf. http://www.alertdriving.com/home/fleet-alert-magazine/international/human-error-accounts-90-road-accidents.

[4] For instance, Plateforme Avancée de Mobilité Urbaine (PAMU) project of Renault, http://blog.renault.com/fr/2013/12/20/pamu-le-service-voiturier-du-futur/.

challenges, such as DARPA Urban Challenge in 2007 helped to foster the development of autonomous cars in urban context (Montemerlo et al. 2008).

There are no more automotive DARPA challenges for the moment, but other competitions and demonstrations still take place. One of the most impressive recent examples has been the Bertha Benz commemorative drive (Ziegler et al. 2014). This achievement was conducted 125 years after the first overland journey in automotive history. The route, posing a large variety of traffic difficulties, has been successfully tamed by a complex localisation–perception–navigation system, a visible sign of the upcoming evolution in the car industry.

As Grand Cooperative Driving Challenge of 2011 has demonstrated, cooperation between vehicles is another promising course to adopt (Geiger, Lauer, Moosmann, et al. 2012). Cooperative control algorithms, V2V and V2I communication methods prosper, and this for several reasons. Firstly, the knowledge of the crowd is often of interest. Emitting and propagating alerts and informative hints concerning accidents or weather condition would surely have a beneficial effect on the security of road users. Sending official messages about traffic congestion, rescue vehicles in the environs or planned events affecting the traffic would influence the comfort or even help to reduce fuel consumption. Secondly, adding information obtained from nearby vehicles into perception process would serve as cheap and easy replacement for additional sensors. On the other hand, all these methods bear safety issues. Falsified messages and Sybil attacks[1] are common dangers lurking on V2V users.

Even with the ongoing progress, one of the main concerns in the domain of intelligent vehicles is still the complexity of the environment. Rural and uninhabited areas do not often present complexity under the same aspects the urban areas do. The former are generally simpler in terms of the number of existing obstacles, but encompass a huge variety of landscapes. The latter, on the other hand, are complex but relatively uniform. Our work focuses therefore on urban areas and tackles problems such as multiple moving obstacles and rapidly changing state of objects. As a support for our choice, the comparison of various road environments, made by Eskandarian (2012) and presented in Table 1.1 shows clearly that the urban scene is the one presenting the highest level of difficulty.

Cars, motorcyclists and bikers, pedestrians — they all make part of the possibly moving objects that can interact with an autonomous vehicle. Walls, doors and gates, buildings, urban furniture and infrastructure are in turn motionless, but they have to be firstly detected as such to benefit from this knowledge. The semantic information provided by automotive sensors is rarely rich enough to make this distinction instantly. Furthermore, any approach tracking the objects in the surrounding environment is faced with the problem of number of possible associations growing exponentially with respect to the obstacle count. Not only there are problems with the high number of obstacles, but also with the unpredictability of their motion. Crossing and rapidly changing trajectories of road users is just one example of challenges to confront.

The good news is that cheaper, more accurate and semantically richer sensors become available. As for the last category, one can mention the Mobileye® vision-based Advanced Driver Assistance System (ADAS). For example, the messages emitted concern collision warnings, detected pedestrians, recognised traffic signs or lane markings. In comparison with a simple camera, provided information is semantically on much higher level. As for the complexity of the environment, which entails the computational complexity, it is encouraging that embedded systems evolve steadily nonetheless and their capacity to execute demanding algorithms get significant.

[1] A *Sybil attack* is an act of forging identities or creating pseudonymous ones and then sending false messages in order to gain some advantages or simply to wreak havoc amongst other (legitimate) users.

Table 1.1 – Comparison of different environments where autonomous vehicles are going to drive. Adapted from: Eskandarian (2012, Table 2.2).

	Motorway traffic	**Rural traffic**	**Urban traffic**
Homogeneity	Uniform traffic conditions	Non-motorway connections	Heterogeneous class, widely different in terms of size and density
Complexity	Sparse network with few intersections, traffic separated by direction	Moderately dense network, two-way traffic	Dense and complex road networks, intersections and crossings; two-way traffic, traffic signals
Flow composition	Moderate to very high traffic flow levels, homogeneous traffic, generally rather high speeds except in congestion with stop-and-go conditions, standardised and predictable road geometry	Mixed traffic but mainly used by motorised traffic, wide range of driving speeds, wide variety of road geometry	Varying traffic loads, complex traffic composition
Driver attention	Low to moderate levels of driver attention, except in (nearly) congested conditions	Moderate to high driver loads	Heavy driver load, low to moderate speeds

Given these circumstances, autonomous vehicles will not emerge on our markets tomorrow unfortunately. Before that, changes in people's mentality will be necessary for the autonomous vehicles to be accepted. Technical challenges aside, the hardest problem to solve is and will be the human. On the one hand, pedestrians, cyclists and non-autonomous vehicles will always be present side by side with those automatised. Including the human factor, with all its unpredictability, when designing an intelligent car will rest one of the hardest problems (S. Lefèvre, Laugier, and Ibañez-Guzmán 2011). On the other hand, safety issues are evident, an automated car must be robust to a much higher degree than a human driver; otherwise, the idea will be rejected. To make matters even worse, multiple social and legal issues exist, such as unemployment of professional drivers due to lack of demand after the advent of the driver-less car, or the question of responsibility attribution in case of an accident. Necessary amendments in the law are another example of a blocking problem.

All the problems stated above would have to be solved at some time. For us, the most persistent issues are however technological even if other topics are of importance and will have to be resolved before the advent of intelligent vehicles on our roads. Thus, in the present dissertation, we will not treat any social or legal topics, but only the technical ones. We will concentrate on methods for environment perception and scene understanding using semantically poor sensors, slightly cheaper than their precise counterparts richer in provided information, and therefore more likely to be adopted on a larger scale. Following the same idea, we have started our research using highly accurate maps that were created on demand and subsequently switched to the use of publicly available and free, but imprecise and uncertain digital maps.

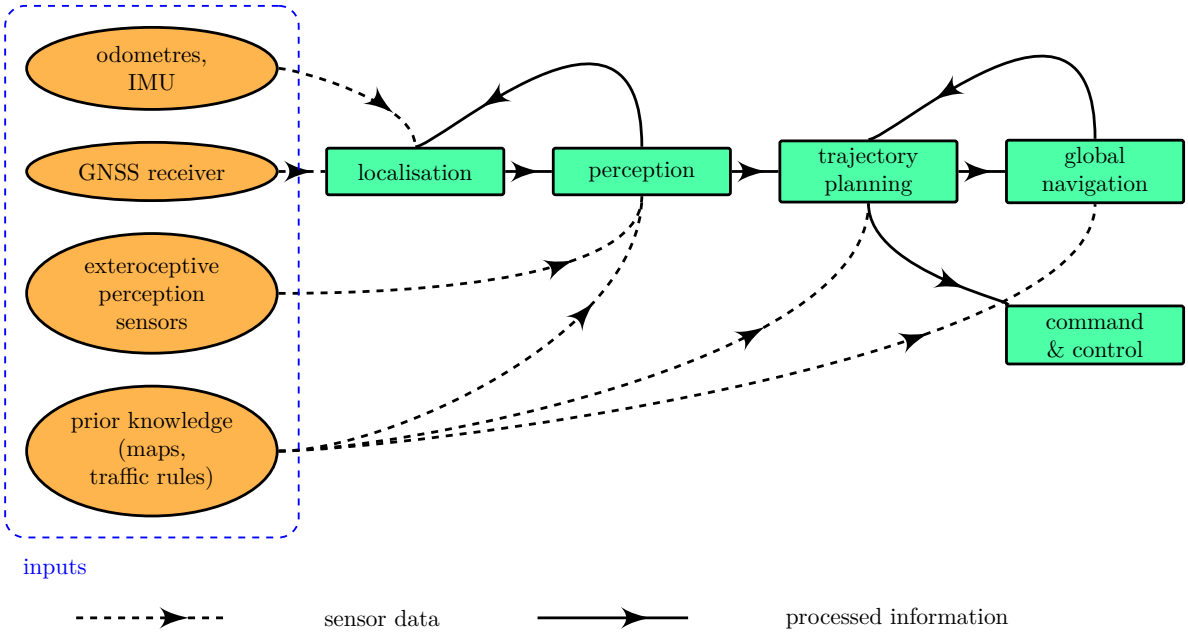


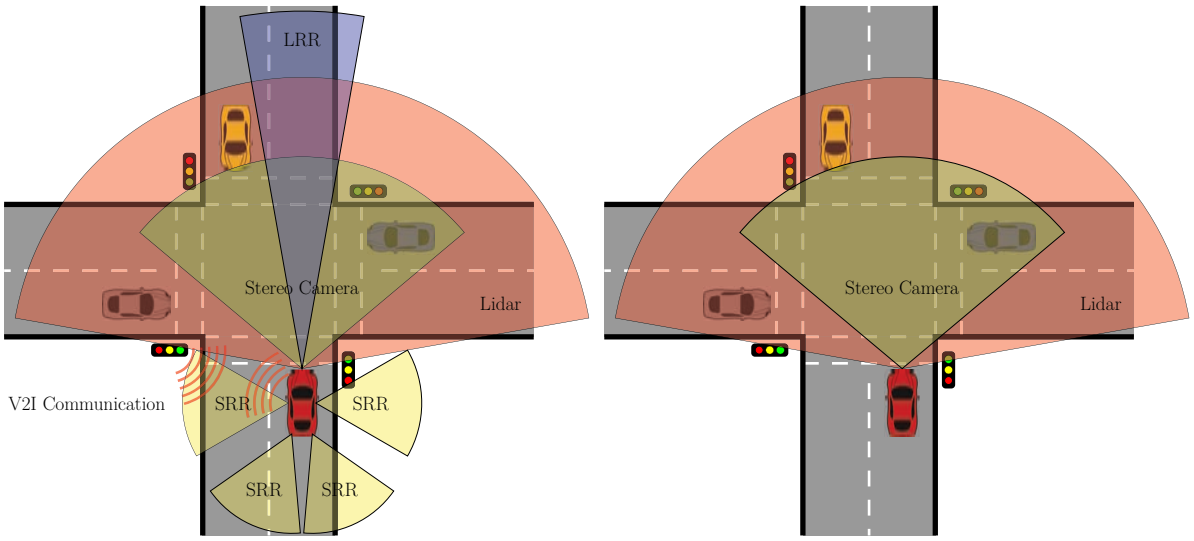
Figure 1.1 – Autonomous driving system overview.

1.2 Base components of an autonomous driving system

An autonomous driving system is generally a complex structure of interdependent modules. Multiple inputs might comprise sensor data, databases and user's desired goal. Subsequent processing modules are typically: localisation, perception, trajectory planning and control. A higher level module for global navigation executing user's destination can coexist as well. An overview of a sample autonomous driving system is shown in Figure 1.1. This particular scheme has a pipeline structure, but various hierarchies can be imagined depending on the specific requirements (Benenson and Parent 2008a). Some authors have even carried out detailed studies on the best adapted architectures for ADAS that use maps as prior knowledge (Durekovic and Smith 2011).

Clearly, each subsystem of a driver-less vehicle can be designed and implemented in various manners. The localisation module presents no exception to this rule. Most often, a global positioning system like Global Positioning System (GPS) is used as the base positioning information source. In systems that incorporate globally referenced data like landmarks or maps, such a solution is a necessity. However, nothing prevents an autonomous system to be constituted only of a local positioning module.

Environment perception systems demonstrate even greater diversification than positioning systems. Firstly, they may perform one or many tasks like obstacle detection, prediction of their motion or moving object tracking (MOT). They might as well include other sorts of scene understanding algorithms like drivable or navigable space detection and object classification. Secondly, one can separate them into two groups: those using one type of sensors and those fusing data coming from multiple sensing devices. Mono-sensor algorithms can be based on vision (single, stereo, or multiple cameras), lidars (single or multi layer, with varying angular aperture), sonars or radars. One can include into that group specialised hybrid sensors, like depth-sensing line-of-motion (e.g., first version of Kinect™) or time-of-flight cameras (e.g., Kinect™ update from Xbox One®) or so called *RACams* combining a camera with a radar. Many perception systems are however supported by multiple sensors of different types. Popular configurations include vision-based systems coupled with radars for both long and short-distance detection and lidars for middle-to-long-range sensing. Typically, a vehicle would have multiple



(a) Richly equipped vehicle.

(b) Poorly equipped vehicle.

Figure 1.2 – Possible autonomous vehicle sensors: cameras, lidars, radars (LRR and SRR), V2I communication.

24 GHz short-range radar (SRR) for detecting near objects and one 77 GHz long-range radar (LRR) for long distance obstacle detection. Possible multi-sensor systems are portrayed on Figure 1.2. Due to their complexity, these systems present often scientific and technological challenges in the domain of data fusion, but on the other hand they provide richer information.

Generally, the next module that builds up on the localisation and the perception subsystems is the one responsible for trajectory planning. That is the place where the autonomous system acts in a manner to advance towards the destination at the local level. The next way point is often provided by a higher level, global, navigation module as described below. This subsystem should also take care of the passengers' comfort. Avoiding excessive acceleration and deceleration as well as keeping the planned path smooth enough, without unreasonable swerving, may be examples of simple comfort criteria. That is also the part of the system responsible for decision making. Tasks carried out include keeping a minimal safety distance from other road users and collision avoidance. The decision process involves not only these actions, but more generally all manoeuvres that can be necessary when executing higher-level behaviours like lane driving, intersection handling or achieving a zone (Urmson et al. 2008). Adhering to the traffic rules is another example where the system has to decide upon a driving strategy.

Once the environment, in which the vehicle is situated, is perceived and understood, the generated trajectory has to be executed. The command and control system is responsible for achieving this task and taking into account the mechanical, electrical and physical constraints of actuators, i.e. motors.

Global navigation subsystems are for the moment the best known components of autonomous vehicles. Their aim is to plan a macro-scale path towards a user-defined destination. Substantially, this type of component does not differ much from the GPS-based navigation systems widely adopted by road users. The main difference is the interface that, in the case of autonomous cars, must communicate with local path planning module and not only with the driver.

Map prior knowledge for intelligent vehicles

Along with the aforementioned advances in car industry, the cartography thrives as well. More and more digital 2-dimensional (2D) and 3-dimensional (3D) maps are available. Starting with proprietary paid services, through commercial but publicly available ones (IGN 2014; Google 2014b) and finishing with open-data projects such as OpenStreetMap (OSM 2013), digital maps become ubiquitous. Notably, during DARPA Urban Challenge in 2007, some teams have used enhanced digital maps (Kammel et al. 2008; CMU 2014). More recently, in commemoration of the famous Bertha's Benz historical route[1], a 100-kilometre-long autonomous drive has been performed (Ziegler et al. 2014). Their system used highly precise road context maps as well as a geo-referenced landmark database for localisation and navigation.

With a completely different purpose in mind, 3D building model database has been recently used to detect aberrant Global Navigation Satellite System (GNSS) measurements and to exclude them (Obst et al. 2012) or to correct them (Wang, Groves, and Ziebart 2012) effectively increasing localisation robustness.

Clearly, there are multiple forms of map priors that can enhance an autonomous driving system. Topological data like route and street graphs are used mainly for global navigation. Semantic information, such as highway code rules or speed limitations may be used for trajectory planning. Geometrical infrastructure models and road surfaces are in turn the ones that interest us most in this writing, where we study how perception can be improved through incorporation of such map-based priors.

1.3 Goal and scope of the thesis

The goal of this thesis is to study the usefulness of prior knowledge for perception and navigation of intelligent vehicles. In this particular context, we examine the methods of information fusion, updating and revision with a highlight on the processing of unreliable or out-of-date data. We study how the use of prior map data improves the perception module of an autonomous driving system. Our other objective is to develop a scene understanding system for intelligent vehicles that delivers semantically rich information about the scene. The output of this system can serve as an input for a trajectory planning algorithm. A further study could therefore draw on these results in order to estimate its contribution on localisation and navigation systems.

Multi-sensor data fusion is one way to improve the performance a perception system and to enlarge its field of view. Unfortunately, adding supplementary sensors is often costly, so the integration of data in time, or temporal fusion, seems to be a necessary workaround. We have therefore chosen to build our perception system on a (relatively) semantically poor exteroceptive sensor, namely a 4-layer lidar. An automotive lidar such as the chosen one can still be considered as cheap, contrarily to e.g. Velodyne sensor. Even if the implementation has been limited to this type of sensing device, the method itself rests general and permits the use of any exteroceptive detector for which a sensor model can be provided.

In order to handle possibly heterogeneous data sources, we were obliged to conceive an adapted fusion scheme that treats all sources homogeneously. Our choice was the grid-based approach where so called occupancy grids are used to represent a part of vehicle environment. Moreover, we have used the prior information source in disguise of a normal exteroceptive sensor, further unifying the fusion

[1]Some more details in the official message at https://www.kit.edu/visit/pi_2013_13901.php.

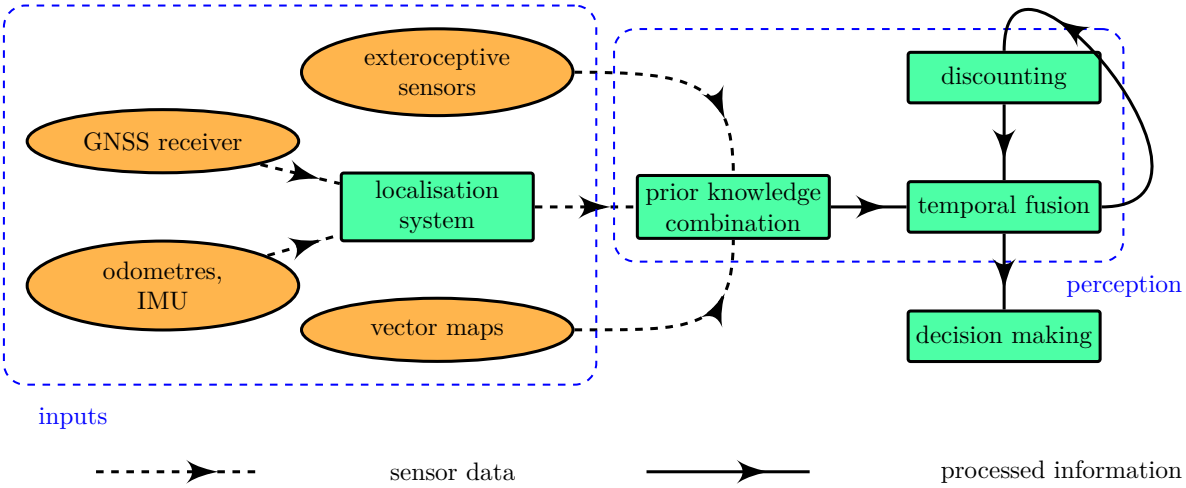


Figure 1.3 – Proposed system overview. Please note that the localisation module is treated as an input.

method. The use of grids as the base data representation has allowed us to design an easily adaptable autonomous system, cf. Figure 1.3. Figure 1.3 presents how our system fits into a general scheme, shown in Figure 1.1. It is worthy to note that as the principal focus was put on the perception module, we can treat other subsystems of intelligent vehicle, such as localisation, as black-box inputs. Also the decision making part is somehow distinct from the general case. Here, we denote under this term the process of understanding the scene by giving a single label to detected elements of the environment.

The prior knowledge mentioned above is in our case based on digital city maps. These maps contain geometric models of buildings as 2D polygons or 3D polyhedrons. Besides, they model the road surface in two or three dimensions as well. The map data that we have chosen to use is clearly geometric, but it contains important contextual meta-information that we exploit as well. We have decided to treat maps as an additional information source and to fuse it with other sensor data. In this way, it is possible to infer more refined information about the vehicle environment. Prior knowledge from geodata is also used in order to control the dynamics of the scene. It is achieved by managing the remanence of scene objects and by using an approach based on evidential grids.

The semantic information gained from the fusion process can be used to perceive and to analyse the dynamics of different objects in the scene. As the objective, we have decided that combined sensor and prior information should be able to distinguish between the following:

- buildings and other urban infrastructure,
- moving and stopped objects,

and characterise free space giving it a label or a measure of *navigability* by an intelligent car.

In our perception system, two different map sources have been used to analyse the behaviour of our method. Firstly, a highly specialised precise 3D maps delivered by National Institute of the Geographic and Forest Information (IGN) and created on demand. Secondly, publicly available 2D map data from OpenStreetMap (OSM) project.

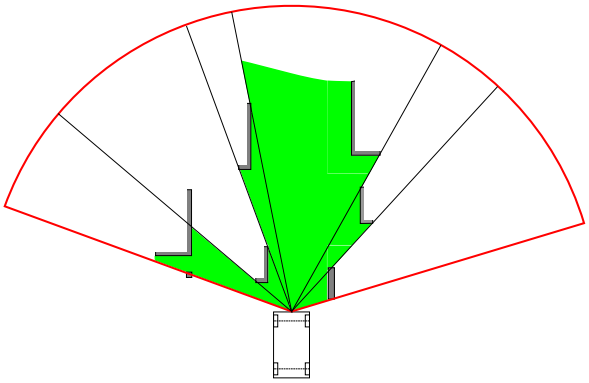
Another objective of this thesis is to find an approach for the fusion of prior information with sensor data and contextual knowledge. We have adopted a mathematical formalism able to perform this step by explicitly managing uncertainty and imprecision inherent to data in an easy and intuitive manner. Temporal fusion exploits the fact that the vehicle moves and so the data from one sensor

can be combined at different moments, therefore at different positions. In this way, the cumulated information makes the limitations of the sensors less stringent. For instance, the effective field of view is enlarged. The combined information is also more reliable, as the incoherent pieces of data, outliers, are eliminated through the fusion process. As seen in Figure 1.4a, raw sensor information is semantically poor and needs more elaborate processing to achieve useful results. Figure 1.4b shows the data that can be retrieved from a city model map. Figure 1.4c visualises various types of objects that a perception system should be able to distinguish using available sensors, prior data and the fusion process. Similarly, the use of multiple data sources and fusion algorithms improves the accuracy and the data integrity, which is another crucial issue in robotic systems. Having accurate data is of course important, but it is essential to maintain data integrity. This means that such data are complete and consistent, and uncertainties are quantified.

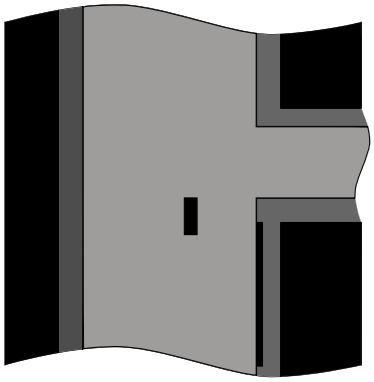
A problem of information update and revision is addressed as well. Indeed, maps are never entirely up-to-date and it is impossible to find a quantitative measure of their correctness or currentness (*Predimap project* 2011). However, we do not address the problem of map creation and updating, but we examine the update and revision operations in the context of information fusion. This can be seen as an alternative to the direct data fusion process.

Urban areas tend to be a demanding environment for an autonomous vehicle. They present problems for both localisation and perception modules. The former suffer from low satellite visibility lowering the quality of received signal and, hence, worsening the accuracy of the positioning. For the latter, the large number of objects is already a first hindrance. Secondly, the movement of dynamic entities is hard to predict and sometimes almost chaotic, e.g. a wandering pedestrian or a straying animal. We have decided to explore this domain for two main reasons. Most importantly, the maps are often not available for rural or unpopulated areas, so we cannot rely on this (non-existent) information for the sensor fusion process. Additionally, we suppose that an intelligent car performing well in a city will also be able to navigate safely through less populated areas (Lategahn, Schreiber, et al. 2013). Under this assumption, other environments can be regarded as secondary. The constraints that we imposed ourselves have been strongly influenced by the requirements of the CityVIP project in which we participated (CityVIP 2012).

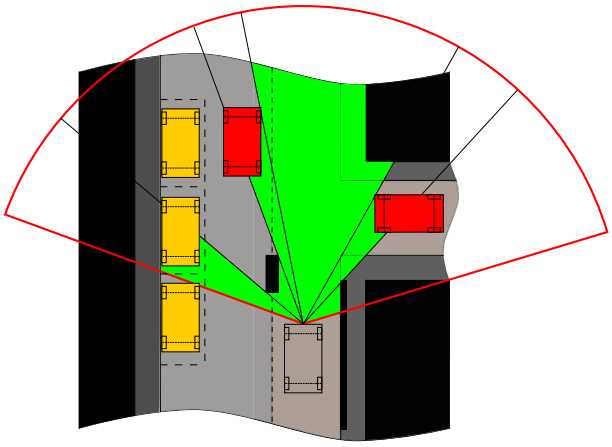
As already mentioned, the notion of scene dynamics presents another important question. Dynamic environments are much more demanding than the static ones, and a robust scene understanding system has to deal with this problem explicitly. Methods used for the perception of static scenes are based on assumptions not necessarily met when dealing with an urban scene. For instance, one cannot suppose that the majority of perceived obstacles are static in order to detect the moving ones. Such a hypothesis may be valid in some particular situations or algorithms, for example RANdom SAmple Consensus (RANSAC), but not in a general perception scheme. The complexity of an urban scene is high in general, so that it is necessary to ameliorate the perception system by adding supplementary information or using higher-level semantic information. The contribution of this research work in this domain is hence to propose a new perception scheme managing the remanence of scene objects through the contextual information obtained from digital cartography. Our approach is based on the fusion of data coming from such maps as well as from embedded sensors and uses evidential grids to represent the state of the vehicle environment. The main idea is to accumulate in time sensor data and to incorporate prior knowledge from maps. The meaning of prior knowledge varies from simply defining the scene context, i.e. road, infrastructure, pavement, park, etc., at a given position to excluding or encouraging the existence of a class of objects. As an example, one can imagine the exclusion of the class “buildings” in the road context or, in reverse, the class “cars” can be fostered in this context. Evidential grids are



(a) Data interpretation of a lidar point cloud.



(b) Prior map information.



(c) Objective of a perception system.

buildings | free space | unknown space | stopped objects | moving objects

Figure 1.4 – Bird's eye view of an example urban scene.

a sort of occupancy grids, but they take advantage of the theory of evidence. A grid covers a part of the environment around the vehicle and describes the position of obstacles and free space relatively to the car. Our system can be juxtaposed against the Simultaneous Localization, Mapping and Moving Object Tracking (SLAMMOT) problem (Wolf and Sukhatme 2005; Thrun, Burgard, and Fox 2005), where the environment is mapped and moving objects are detected. There are however important differences with our approach and the SLAMMOT algorithms, as the latter perform localisation, mapping and object tracking.

The difficulty of dealing with the dynamics of the scene arises from two main reasons. The system carrier (robot, vehicle) is a moving actor in interaction with other objects in the scene. These objects themselves can be mobile: momentarily stopped or on the move. One can remark that moving versus static object detection is not provided by any optical sensor, but is the result of the fusion process. Indeed, our perception system does not include any sensor like a radar, which uses Doppler effect. Temporal fusion and data accumulation serve a double purpose. On the one hand, they allow filtering the sensor noise and, on the other hand, to conserve some pieces of information. Preserved information can, for instance, concern the zones of vehicle environment that are not subject to occlusions. An important assumption is made: the scene dynamics is limited. It means that one can fix a forgetting factor which bounds the process of information conservation. This parameter is closely attached to, and acts on, the data remanence: adapting this parameter changes the persistence of a given stimulus after its disappearance. The term stimulus corresponds to sensor data; the persistence represents the time period during which these data are *expected* to be present in the perception grid.

1.4 Contributions

The contributions exposed in this thesis are varied and are not limited to the domain of intelligent vehicles. What concerns this domain, we have conceived and developed a real-time system capable of perceiving and partially understanding urban scenes. It is able to detect and distinguish between static and mobile obstacles as well as to characterise free space by labelling it as drivable or non-drivable. A part of the here described research constituted the CityVIP project (CityVIP 2012) by the French National Research Agency (ANR) and had as subject the creation of a small autonomous personal vehicle for urban environment.

Another huge part of the research has been done in the subject of information fusion. Especially, we focused our work on the means of updating and revising information. We have elaborated contextual temporal discounting methods that take into account the class, also understood as the context, and the age of the data (Kurdej and Cherfaoui 2013). Our main focus was however put on the fusion of prior knowledge, such as maps, into instantaneous data, obtained from sensors for example (Kurdej et al. 2012; Kurdej et al. 2014).

Furthermore, we discussed the possible elements of digital maps that can be incorporated into a perception system of an intelligent vehicle. We outlined the advantages and the disadvantages of different types of maps and made a survey on the information type useful in such systems (Kurdej et al. 2013; Kurdej et al. 2014; Kurdej et al. 2015).

In order to validate or discredit a tested approach, we wanted to put theory into practice. For this reason, we worked on the implementation of our perception system. It has permitted us to test developed methods on real data sets. This work resulted in a working prototype of a perception system implemented on our test vehicle. It helped as well in the development of our laboratory platform, Perception et

Assistance pour une Conduite Plus Sure (PACPUS), for real-time systems (Heudiasyc 2013). A part of implementation was devoted to the processing of map data, along with handling of various map types and the capability of on-demand downloading and processing. Apart from the open-source PACPUS platform, this thesis resulted in creation of a Matlab toolbox and of an open-source C++ library for data fusion using belief functions theory[1].

1.5 Dissertation organisation

This report is structured in four parts. Part II portrays current state of the research in the domain of intelligent vehicle perception and the use of digital maps in this context. Chapter 2 describes the advances in robotic localisation, perception and navigation. Special attention is drawn to the use of prior knowledge and, in particular, maps. Chapter 3 elaborates on theories of uncertainty management and information fusion.

Part III presents and studies methods for exploiting digital maps for intelligent vehicles. Chapter 4 handles the methods we create the sensor models. A sensor model is responsible for transforming raw sensor data into a homogeneous representation that will be worked upon in further steps. In our case, this representation is based on evidential occupancy grids, which are described in this chapter as well. This chapter gives also some techniques and intuition about the methods for creating mass functions that define a sensor model. In Chapter 5, we describe the data fusion methods that have been developed during this thesis. They concern mainly the way of merging prior map knowledge together with data from sensors like lidar or camera. As the information discounting is an essential part of this fusion process, Chapter 6 goes into this subject with more details and presents a comparison with existing methods. The problems of data updating and revision is treated in this chapter as topics complementary to the data discounting and information ageing.

Part IV is devoted to experimental results. The set-up used to test and validate our approaches is described in Chapter 7, whereas in Chapter 8 we show the performance of the implemented system applied to real data.

The report ends with Part V which gives a brief commentary on this dissertation and outlines a few perspectives for future work.

Appendices contain supplementary information that which are not vital to the subject, but still can be of interest to the reader. And so, Appendix A includes several proofs and development of calculus of presented methods for information discounting, whereas Appendix B describes implementation details of conceived systems. The latter handles the aspects such as functional and non-functional requirements, algorithms as well as lists the software libraries use for implementation.

[1]Available at: `https://github.com/mkurdej/bft`.

Part II

Research on maps and uncertain data fusion for autonomous vehicles

The purpose of this part is to describe the current state of the research in the domains concerned by this dissertation. In addition to the theoretical background, we present briefly the mathematical tools used further in this dissertation.

Chapter 2 introduces the reader to the problems of mapping, localisation, tracking and navigation for autonomous vehicles. Each part was treated in relation to the use of digital maps for this particular purpose.

Chapter 3 gives an overview of mathematical theories used for information fusion. More particularly, it handles the methods of fusing uncertain sensor readings together and with other sources, e.g. contextual information. It presents a detailed introduction to the belief functions theory (BFT) and describes methods for the management of ageing data, i.e. pieces of information that correspond to the reality they describe with accuracy decreasing in time.

[This page intentionally left blank.]

Chapter 2

Using digital maps and city models for intelligent vehicles

"Maps are essential.
Planning a journey without a map
is like building a house without drawings."

Mark Jenkins, *The Hard Way*

Contents

The proliferation of digital maps in the recent years has enabled completely new usage possibilities. The trip planning, widely known to the general public thanks to automotive navigation systems, is no more the only practical usage. Autonomous vehicle systems and Advanced Driver Assistance System (ADAS) are among the possible targets where cartographic databases can be employed. A few years ago, the use of such data in this context was understudied. Only few researchers took interest in the use of maps for the localisation, perception or trajectory planning for intelligent vehicles. Currently, it is generally admitted that the use of prior information is a major subject that has to be taken seriously and that is a promising means of enhancing the performance of ADAS. Still, the manner in which the map content can be employed in automotive systems is not clear enough. Cartographic databases contain tons of information out of which not everything is exploited yet. Both the map elements to be used and the approach for using them are not obvious. It is also interesting to know what is the minimal necessary information needed for a specific application or a particular vehicle subsystem.

In this chapter, we give an overview of various methods that try to enhance an automotive system by using some prior knowledge from maps. We tried to present as diverse approaches as possible. The different types of map data used in the presented works are hence discussed as well. In the first place, we describe the maps themselves. The methods for map creation and update are then discussed as well as a few related problems. We treat the domains of localisation and perception, from time to time mentioning navigation, in the following sections.

2.1 Digital cartographic maps

Maps are ubiquitous, used by humans for at least a few thousand years, they are almost employed at every day basis without much need for learning to use them. Available on various supports, traditionally paper, recently more often used on electronic devices, they are indispensable when one needs to find a place, optimise one's transport time or simply explore an area. Earlier sources referred to maps using phrases like *a picture or chart that shows the rivers, mountains, streets, etc., in a particular area*[1]. Nowadays however, these are just possible representations of maps. A more general description of a map may say that it is a symbolic depiction without defining explicitly what are the symbols in use.

Digital maps are a perfect example of a symbolic representation without necessarily a picture or a chart. They can be represented by charts, but more often they are a dataset consisting of vectorial approximations of modelled entities, e.g., a multi-segment line can approximate a road. Eskandarian (2012) in *Handbook of Intelligent Vehicles* defines a standard digital map in automotive applications as a one that mainly contains geometric information and other relevant attributes about the road. The core geometry consists of links and nodes connected together forming the road centrelines of the road network or, less often, forming a multi-polygon modelling the road surface. The links between nodes are important for applications like routing. The shape of a link, if it is not a straight line, may be represented by one or more shape points which are intermediate points between the start and end nodes of the link. As it is implied above, the shape points that describe a road segment are not placed at equidistant intervals. All the map annotations are referenced to links, nodes, and shape points. These attributes can be point of interests (POIs), traffic signs, speed limits, etc., which are sufficient for routing and navigation applications. Moreover, the map can be enhanced with further attributes such as the type of road, number of lanes, lane width, and type of lane markings which are needed for more sophisticated applications. In the following, we will deal with maps like this, highlighting geometrical or topological relationships between elements of space; the elements would more often than not be streets and roads, buildings and infrastructure, signs, markings etc.

There are many providers of digital maps nowadays, both commercial companies, open-source communities and government institutions. Everyone presents benefits that others cannot claim to possess, but suffers from its own problems. Freely available maps usually face financial problems and cannot be as exhaustive and as up to date as others. They have often a huge advantage of large active community of volunteers that develop, correct and update the maps. On the other side of the barrier, there are public institutions, such as National Institute of the Geographic and Forest Information (IGN) (IGN 2014) and privately funded companies that provide free or paid[2] services. Companies specialised in map creation, such as the former NavTeq, currently the part of Nokia HERE platform,[3] or Tele Atlas[4] have gathered an important know-how in the domain of the construction and update of digital maps. Similarly, Internet

[1] Source: Merriam-Webster dictionary, http://www.merriam-webster.com/.
[2] Both directly or indirectly, e.g., through advertisement income.
[3] http://www.navteq.com/
[4] http://www.teleatlas.com/, http://www.tomtom.com/

giants like Microsoft (2014) and Google (2014b) as well as many others offer their cartography services. The major advantages these enterprises offer is the quality of service that is maintained and can be relied on. Still, offered maps are not flawless and errors or imperfections may creep into the data. As the environment being mapped is in constant change, these shortcomings are inherent to this domain and should be dealt with whenever possible.

2.1.1 Free map databases

One of the free map providers is the OpenStreetMap (OSM) open data project (OSM 2013). It presents a very good example of a publicly available map provider, so we will focus our discussion on this map, hopefully without any loss in generalisation. OSM has become popular in recent years and has been promoted to the first choice for researchers working with digital maps. To name a few of its advantages:

- modifiable by everyone (and hence rich and responsive),
- easy application programming interface (API),
- progressing exhaustiveness,
- sufficient accuracy for most usages[1],
- possibility to download map data and use it off-line,
- good documentation,
- open-source components (renderers, editors, plug-ins),
- free of charge,
- modifiable both globally (on-line) and locally (off-line), hence adaptable for specific purposes.

It goes without saying that such an ambitious project cannot be flawless. The majority of defects are actually the other side of the coin of the above stated advantages. The most important disadvantages to mention are:

- modifiable by everyone (and hence easily corrupted),
- limited query size due to modest infrastructure and computational power,
- complex queries impossible without complementary API (e.g. XAPI[2] or OSM3S[3]).

OSM project gained its popularity partially due to the easily comprehensive data organisation used to manage all the geodata. Nodes, ways and relations with tags are the main elements of this easily extensible scheme. This project enabled many research projects that would miscarry without its existence. Several undermentioned works have successfully employed OSM as the principal map source. Relatively recently, OSM project has been enhanced by introduction of 3-dimensional (3D) building models and so it has decreased a major gap that was separating it from commercial map-makers[4].

[1] The accuracy of digital maps is though difficult to be measured (Eskandarian 2012).
[2] For more information about eXtended API (XAPI), please see http://wiki.openstreetmap.org/wiki/XAPI.
[3] More details on Overpass API (OSM3S) can be found at http://wiki.openstreetmap.org/wiki/OSM3S.
[4] See http://wiki.openstreetmap.org/wiki/3D (OSM 2013).

2.1.2 Proprietary maps

Commercial maps are at the other side of the cartographic horizon. They are essentially superior to their free counterparts, when one excludes the question of pricing. Guaranteed quality of service for on-line services is probably the most important plus-value missing in free services.

The exact way in which major map providers such as Navteq or Google create their maps is unknown, but there has been some research that elucidates a bit the subject. Published research from public organisations such as MATIS laboratory gives an invaluable insight into domain (Soheilian, Paparoditis, and Vallet 2013; Paparoditis et al. 2012). These authors present both the platform used for recording of data and the algorithms used for detection and 3D reconstruction of environment elements, in this case, traffic sign. Hammoudi (2011) wrote a Ph.D. thesis about 3D city modelling and his contributions to this subject. A whole spectrum of methods based on various sensors have been approached. Processing of aerial images, 3D point clouds and terrestrial camera images have been described. This dissertation gives an insight into how modern maps could be created. At aerial level, a method of polyhedral building reconstruction is proposed. As an input, a set of calibrated aerial images is necessary. The presented approach is direct and does not use image features. At terrestrial level, data used for map construction were obtained thanks to a specialised Mobile Mapping System (MMS): a vehicle equipped with highly precise Riegl laser range sensors and a multi-camera system (see Figure 2.1). Several approaches aiming at 3D building façade modelling were proposed. The lidar data in form of a point cloud was processed using segmentation and classification algorithms. Camera images were used in order to extract façade texture information and, in conjunction with point clouds, to model building geometries.

A lot of research were done on the processing of such data. Another Ph.D. thesis conducted at IGN describes more in detail the way how the building façade images were segmented. Described methods, based on alignment and repetitivity properties of façade structures (Burochin 2012). Other works that describe the way in which the maps are created are numerous, but we can at least list Soheilian, Tournaire, et al. (2013) that highlights some methods for producing a 3D city model as well as Tournaire (2007) which concentrates on the extraction of horizontal road marking.

Highly accurate and precise maps obtained through methods similar to the ones described above are in use, e.g. on the so called GéoPortail – geographical web platform with multiple types of map data (IGN 2014). Such maps have been used also within various research projects in intelligent vehicles such as CityVIP (2012). Please refer to Chapter 7 for details. Once again, Soheilian, Tournaire, et al. (2013) is a valuable source of information about map-making methods for applications in autonomous vehicles.

2.1.3 Research on maps for autonomous vehicles

The area of map creation itself is a well-developed topic studied for centuries. However, with the emerging field of intelligent transportation systems, this field had to be reviewed and updated. Lots of recent research works have consecrated their energy into the question of creating maps for automated vehicles. Pretiv and Predimap projects conducted by the Heudiasyc laboratory (*Pretiv project* 2012; *Predimap project* 2011), for instance, addressed this problem explicitly. Both of them concern extracting map data that is relevant to actions performed by intelligent vehicles. For example, does a perception system need to know the existence of all traffic signs, bus stops, trees? How should be changing information handled, is there necessity to store seasonal cartographic data about the environment? Does temporal road deviations, information about works and accidents may enhance the performance

Figure 2.1 – IGN urban Mobile Mapping System equipped with a multi-camera system, turning Velodyne lidar, several high-precision lidars and a precise hybrid localisation system based on a GPS receiver coupled with an INS. Stereopolis vehicle is also equipped with a panoramic head and two pairs of stereo cameras. Source: Paparoditis et al. (2012).

of an autonomous vehicle? Predimap project worked as well on the ways of map management and revision. Necessary level of detail for various activities has been studied as well. Lots of above questions remain open, but there are research papers that shed some light on the actual question of map creation (Tournaire 2007; Soheilian, Tournaire, et al. 2013; Yoneda et al. 2014).

2.1.3.1 Landmark databases

As a landmark database, we denote a database containing the description of physical objects (landmarks) along with their position, usually geographical or topographical. One of important works considering the creation of such maps has been done by Lategahn, Schreiber, et al. (2013). The authors of this work addressed this important problem from quite a novel point of view. Since geo-referenced landmark databases are their most common use case, they presented the mapping pipeline required for the creation of landmark maps. In the proposed approach, a backward facing stereo camera is used, as well as a high precision GPS receiver. First step involves bounding consecutive pose estimates obtained from GPS sensor by visual odometry constraints. Secondly, image points get matched across entire data sequence and 3D landmarks are reconstructed and saved with associated image descriptors. Setting up constraints between camera poses in the first step is crucial for the algorithm. Optimisation process fixes these poses and calculates landmark positions using a global approach.

Images in the aforementioned method are described by holistic image feature vectors as presented in (Lategahn, Beck, et al. 2013). The authors argue that it remains unclear how a descriptor should be constructed, notably under varying illumination conditions. The article states that the problem of choosing the right descriptor becomes even more pronounced in the context of life long mapping, the demand on which increases with the advent of ADAS technologies. They present therefore a set of building blocks for automatic image descriptor construction.

The subject of landmark-based geographic databases and the number of applications taking advantage of such data is very vast. An interested reader would certainly have a useful insight into this domain

by reading one of the following sources Royer (2006), Larnaout et al. (2013), and Lothe et al. (2010).

2.1.3.2 Geodata for intelligent vehicles

In the research works that we present, different types of geodata are used. Dawood et al. (2011) and Cappelle et al. (2012) exploit for this purpose a 3D Geographical Information System (GIS) enhanced with geo-localised images. Drevelle and Bonnifait (2011) propose using a drivable space map for precise localisation. Many authors manipulate Digital Elevation Model (DEM) information to enhance localisation (Mandel and Laue 2010; Drevelle and Bonnifait 2011; Obst et al. 2012). GISs containing road signs and markers are often employed as well, especially in landmark detection systems (Lategahn, Schreiber, et al. 2013; Ziegler et al. 2014).

Not only geometrical and visual information is used in GISs. Topological data is of great importance as well. Road graphs have been employed for localisation and navigation purposes for years now, but they are still being developed and enhanced (Fouque, Bonnifait, and Bétaille 2008; Velaga, Quddus, and Bristow 2009). One can summarise the different map data that can serve to enhance an intelligent vehicle with the following elements:

- geometry,
- classification (type),
- orientation (heading),
- segment proximity,
- segment connectivity.
- turn restrictions.

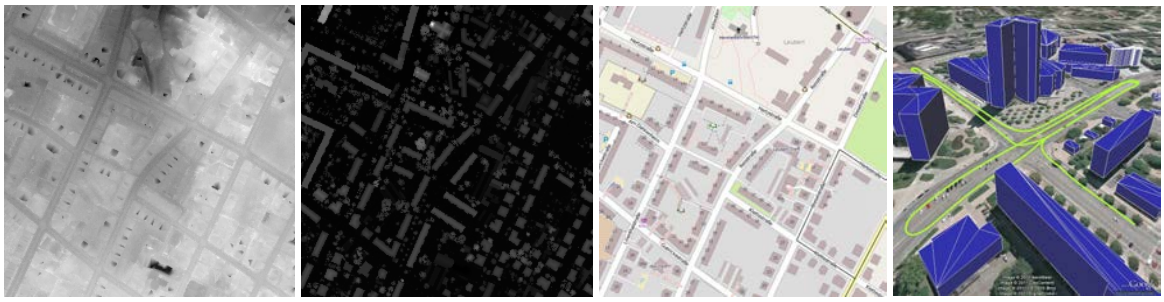
Further elements can obviously enhance maps, but the aforementioned represent the vast majority. As it will become clear further, we focus on the geometrical aspects of maps.

2.1.3.3 Road map database management, updating and error detection

Efficient map data management is one of practical issues that any map-based ADAS faces all the time. Supervising such databases in real-time has been the subject of some publications. For instant, Boucher and Noyer (2011) presented and approach based on an Unscented Kalman Filter (UKF) for missing road detection. The authors model a map database as an additional sensor, which permits to take into account the uncertainties and the errors of the database. Moreover, such an approach allows them to merge GPS and road map data together in an unified centralised fusion scheme.

Due to the spreading usage of maps, the detection of map errors and the updating of their content become necessary. The maps for ADAS are no exception. Missing data, misclassified landmarks or simply not up-to-date information can bring an ADAS that relies on maps to hazardous situations or other misbehaviours. Zinoune, Bonnifait, and Ibañez-Guzmán (2012a) recognised this problem and proposed an approach based on comparison of estimated vehicle trajectory with the geometric map data. Monitoring of residuals allows them to estimate if the digital map can be considered reliable.

Another problem, the detection of missing roundabouts has been the subject of a recent publication (Zinoune, Bonnifait, and Ibañez-Guzmán 2012b). The proposed method tries to identify during the drive misclassified roundabouts through graphical pattern recognition. A Bayesian classifier is trained



(a) DTM (b) DEM (c) OSM (d) Final 3D model

Figure 2.2 – Generation of 3D building models from 2D map (OpenStreetMap) and DEM data. Source: Obst et al. (2012, Figure 3).

in order to recognise in the vehicle trajectory, or precisely in its buffer track, common patterns of driving through a roundabout. In turn, detected potential roundabout centres are submitted to the algorithm called *instantaneous centre of rotation* in order to find the best match. The authors of this research are aware that maps can contain geometrical, topological and attribute errors, but the scope of the approach is limited to geometrical ones.

Quite a different point of view was taken by Xiao et al. (2012). Their objective was to detect the changes in trees in urban areas in order to update existing maps. Multi-temporal point clouds from airborne lidar were the main data employed in this study. Several stages of processing were necessary. Firstly, a classification of tree and non-tree classes is executed. Next, a point cloud, which consists of many non-connected points, has to be segmented in order to connect lidar impacts corresponding to the same tree. The connected components algorithm has been used for this purpose. Single trees were in turn separated from multiple tree components. In the following, two different methods, one point-based and one model-based, were applied in order to derive tree parameters. Lastly, such created tree models were matched against their counterparts from different time moment in a tree-to-tree comparison.

2.1.3.4 Creation of 3D map for applications in intelligent vehicles

3D maps are not always available, but some algorithms may need them. Even if this is a case, a solution has been proposed by Obst et al. (2012). Only a 2-dimensional (2D) map and DEM data are needed to produce an approximate 3D map. The authors generate 3D building models from OpenStreetMap 2D map and STS-99 Shuttle Radar Topography Mission (SRTM) DEM data. Figure 2.2 shows the input data involved and some possible 3D output building model. The approach is relatively simple and consists in extruding 2D building footprints into 3D models. DEM data is queried to find the height of the building. In this manner, only approximative model is obtained, as exact form of the building cannot be determined. The accuracy depends heavily on the resolution of the DEM data. Typically, first freely available Digital Terrain Model (DTM) data called GTOPO30 were obtained at the resolution of 30 arcseconds, i.e. approximately 1 km. The SRTM project improved this result bringing it down to about 90 m on certain areas. Even if the DTM data have rather poor accuracy, together with building models, they create an accurate elevation model that should be useful in most of the situations.

Kim et al. (2008) proposed an innovative way of creating (and possibly updating) 3D building maps. 3D buildings are reconstructed by measuring and exploiting the diminution in signal-to-noise ratio (SNR) of GNSS receiver. Such an effect arises when the buildings obstruct the line-of-sight between the receiver and the satellites (see Figure 2.3 for a typical situation). As mentioned in the article, such a method, if applied on a mass scale in mobile devices, would provide an inexpensive and quite accurate maps.

More importantly, these maps could be updated continuously and so be a reliable source of information about urban environments. The method has two main stages. Firstly, knowing the positions of satellites, density maps are created to detect multiple buildings. Secondly, the region and the size of a specific building are estimated and the corresponding 3D model is generated. Before the reconstruction can take place, the position of the receiver has to be estimated. A density map shows the probability of the GPS signal at given location being obstructed by a building. Using mean-shift clustering (Cheng 1995), the centre of a building is determined and its region is estimated by applying a threshold on the dominant cluster. A 3D building model can be then reconstructed using a grid-based voxelisation algorithm.

The interior space modelling is situated at the opposite end of the domain of map construction. Building 3D visual maps of interior space has been approached in a novel way by Kwon, Ahmad Yousef, and Kak (2013). Preoccupied by the propagation of errors from low-level fusion mechanisms to the higher levels, the authors conceived a hierarchical Simultaneous Localization and Mapping (SLAM) method. The map building process is divided into 3: local, intermediate and global. The extraction of primitive line features from range sensor data is performed in the local step. The intermediate phase consists in stitching locally created maps together and adjusting robot rotation. Final phase, the global one, integrates intermediate results constructed at different positions of the robot into a single global map. This approach is based on a few assumptions about the world in which the robot evolves. Indoor environment is completely different than an outdoor urban scene and it is also easier to understand by a computer system. Regular shapes and easily detectable features are the most salient differences in comparison to complex urban environments. Anyway, presented results are very satisfactory and demonstrate that such a hierarchical approach tends to give better results, possibly applied to outdoor scenes as well.

2.1.3.5 Map update and data quality

The quality of data is a metrics than can be hardly obtained for maps (*Predimap project* 2011). One could of course undertake a survey on the correspondence between the reality and the mapped model. The difficulty in this approach is that the constantly changing environment would out-date both the reference model and the map. For this reason, the map data cannot be evaluated in the same way as it is often done for sensor data. A viable solution for this problem might be decreasing the level of confidence attributed to the map as the time goes, supposing that an out-dated piece of information is unsure. This approach can be unfortunately executed only partially, the main reason being the lack of necessary information in maps. Normally, a digital map contains only one timestamp being the date of the last update. It is impossible however to obtain the date and the time of the last modification or verification of a single map element. There are notable exceptions to this rule, for instance, the OSM data contains a timestamp for each changeset modifying the map. The reliability of this timestamp is however questionable.

2.2 Maps for localisation

Having a reliable, precise and accurate piece of information about the ego-position of vehicles is an important requirement for ADAS and autonomous cars. As the positioning is often one of the first steps executed by a perception and navigation system, we present hereafter a selection of recent algorithms

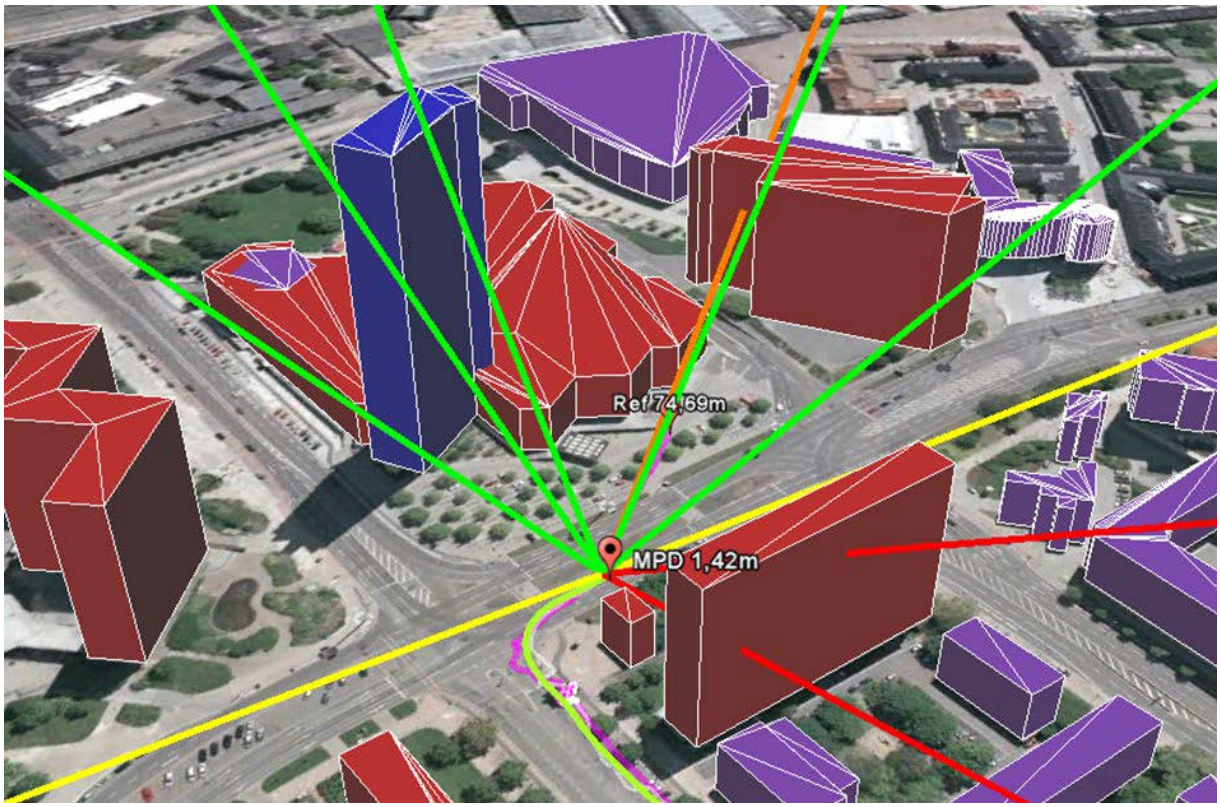


Figure 2.3 – Typical multi-path situation in urban environment. Blocked signals are indicated in red and those directly observable — in green. Source: Obst et al. (2012, Figure 1).

that serve for vehicle localisation. A growing number of research papers focus on the use of digital maps in order to enhance the performance of existing localisation methods.

2.2.1 Maps, GNSS and dead reckoning (DR)

An important group of localisation methods using map data consists of the approaches that try to detect and correct wrong measures thanks to the knowledge of building shapes. One of such methods has been developed by Obst et al. (2012) in order to reliably localise a vehicle in urban environments. Thanks to building geometric models, satellite signal multi-paths can be detected. They are subsequently excluded from the GNSS localisation algorithm. In this work, it was applied to a hybrid GPS and GNSS (GLONASS) system. The principle of the presented approach is to detect if the direct line-of-sight of the vehicle GNSS receiver towards some satellites is hindered by buildings. This is presented in Figure 2.3. Pseudo-ranges which are not directly observable will not take part in the position computation. This algorithm needs obviously an initial uncorrected position.

Other authors were interested in detecting multi-path and echo effects in GNSS signals as well. Ben-Moshe et al. developed a similar algorithm that serves to remove satellites that are not in the direct line-of-sight from the position computation algorithm (Ben-Moshe, Carmi, and Friedman 2014). The approach is based on a 3D building model and visibility graphs. What is more, authors created a GNSS simulator for testing positioning algorithms. In order to improve the performance of tested algorithms, they introduce several heuristics and take advantage of the information about Radio Frequency (RF)-signals disturbed by urban infrastructure. Other research works, instead of using buildings model, use DEM data. We can include in this group the works of Mandel and Laue (2010) and Drevelle and Bonnifait (2011) amongst others.

To the same group of GNSS-based localisation methods, we can add approaches that combine as well the dead reckoning data. Toledo-Moreo et al. (2009) proposed a method that applies map-matching techniques. Authors integrate in their method GNSS, dead-reckoning (odometry and gyro) and map data. A particle filter is used in order to perform lane-level localisation. Proposed method uses also European Geostationary Navigation Overlay Service (EGNOS) data to enhance the accuracy and provide the integrity of the positioning. Velaga, Quddus, and Bristow (2009) carried out a comprehensive comparison of map-matching algorithms for localisation. The authors developed an enhanced weight-based topological algorithm for Intelligent Transportation Systems. For this map-matching method, they use proprioceptive vehicle data like its speed, positioning data obtained from a GPS or a GPS+DR system as well as a spatial road network.

2.2.2 Maps and vision-based localisation algorithms

Somewhere between the methods based on dead-reckoning and camera-based approaches, there is situated an interesting group of methods, conjunctively known under the name of visual odometry (VO). These algorithms do not typically use any map information, but they should be signalled as a viable replacement for hardware odometers. There are many approaches, using either mono or stereo-vision. Recently, the stereo-camera approaches comparing current set of images with previous recordings, called sometimes quadrifocal VO, have proved very good performance and can be used in real-time (Royer 2006; Comport, Malis, and Rives 2010).

A lot of research is done on camera-based localisation with the aid of digital maps. Visual landmarks inside a GIS database are among the most popular choices for a localisation system. Highly precise ego-localisation methods using a mono camera and an Inertial Measurement Unit (IMU), like those proposed in (Lategahn and Stiller 2012a; Lategahn and Stiller 2012b; Lategahn, Schreiber, et al. 2013) under the name of *city GPS*, demonstrate very good performances with an accuracy of 10 cm over a sequence of 10 km. The conceptual simplicity of their system along with its performance seem extraordinary. The idea is to get a rough position estimate through processing a camera image and searching it in a landmark map. By fusing this first guess with IMU measurements, a refined localisation update is obtained. This idea has been presented as well (Lategahn and Stiller 2012a).

Cappelle et al. (2007) employ a 3D GIS database enhanced with geo-localised images. The idea is conceptually simple: two images are to be matched, the real image captured by a camera and the geo-localised virtual image from the database, in order to improve the position estimate obtained from a GPS receiver. As validation, the authors chose to compare the vision-based results to the laser-range sensor readouts (Cappelle et al. 2008). The presented approach is composed of two stages: it starts with image feature computation and ends by an information fusion step. As opposed to the work of Y. Yu et al. (2013), the features used in this method are low-level characteristic points of the image, rather than models of the road scene (lane markings etc.). This method is further developed by Dawood et al. (2011) using Scale Invariant Feature Transform (SIFT) and Harris corner detection (Lowe 2004; Harris and Stephens 1988). This study has been developed in (Cappelle et al. 2012) and extended to obstacle detection and navigation.

An interesting approach for lane-level localisation has been proposed by Y. Yu et al. (2013). The major hypothesis is that the vehicle environment, particularly in urban situations, is well structured. It implies that the dominant features are line-like, e.g. lane markings, curbs, poles, building edges, etc. Furthermore, they coincide with main axis of the road: longitude, latitude and vertical. The authors propose therefore a method based on so called Road Structural Features (RSFs) extracted from a set of

line segments. The approach works by associating map-predicted RSFs with the ones obtained through measurements.

SLAM methods are another class of algorithms used not only for perception, but for localisation as well. As an example of this algorithm type is a localisation system based on data from OSM project proposed by Floros, Zander, and Leibe (2013). The main information source is a stereo camera producing images processed by a visual odometry (VO) module. This method is a meta-algorithm, since, at bottom, it uses a Monte Carlo localisation framework, which is then enhanced by maps. However, a whole class of positioning algorithms could be improved in this way. In this particular case, the map data acts as an additional cue incorporated into the observation model.

2.2.3 Lidar-based localisation enhanced by map data

The last group of localisation methods that we want to present are based on lidars. These laser-range sensors are generally used for object detection. The sparsity of the delivered information was the main reason for such a state of things. The introduction of dense laser scanners like Velodyne allowed their use also for road and infrastructure detection. This opened a whole new spectrum of methods, some of them described below.

One of approaches coupling a lidar with map data is the one presented by Hentschel, Wulf, and Wagner (2008) which couples a GPS receiver, a laser-based sensor and a 2D reference map containing static line features. This method is dedicated for localisation in both, urban and non-urban, outdoor environments. The principle of functioning is the following. GPS measurements are filtered by a Kalman filter using inertial data and wheel odometry. Next, line features from reference map and 3D laser range data are integrated with the result of Kalman filtering into a particle filter (Thrun, Burgard, and Fox 2005). The main advantage of the approach is that when the GNSS satellite signal quality gets poor (like in close distance to buildings), the sensor fusion permits the system to localise the robot precisely. An interesting idea of so called virtual scans is applied in the method. It consists in generating the expected lidar measurements given a GPS pose and map data. The output point cloud is the one that would be generated by a lidar if the surroundings were exactly like the infrastructure modelled by the linear 2D model.

Another approach for localisation in mapped environments based on lidar scan features was proposed by Yoneda et al. (2014). The authors use a highly precise 3D map and a 32-layer Velodyne lidar. They propose a feature quantity that helps to choose information-rich layers of point clouds depending on the type of the surrounding environment. This quantity is calculated from inclination angles of scan points and the size of point clusters. The authors show that the computed quantity is related to the environment type and that selecting an appropriate scan area improves the positioning accuracy. Likewise, it helps to effectively extract points of interest from the whole lidar scan. Presented experiments demonstrate respectable accuracy of the localisation system below 3 m.

Localisation systems can be based on occupancy grids[1], or grid maps, as well. One of such approaches using digital maps and multi-modal sensor fusion was presented by Konrad, Nuss, and Dietmayer (2012). The authors defined a general grid map definition and presented three grids to demonstrate their results: a laser range scanner occupancy grid, a video grid based on an image processing method called Inverse Perspective Mapping, and a feature grid containing prior knowledge in form of lane marking features. As a digital map, the authors used a GPS-like road map containing waypoints that

[1] See Chapter 4 for details on different environment representations, including occupancy grids.

describe the road network. The method estimates the road course and road width by matching the digital road map and a sensor-based grid map. One of the most important step of the algorithm is the detection of road borders. This step is however largely sensor-dependent and is of less interest in the discussion here. In the next step, the road border hypotheses are compared with the digital map features. These features are estimated from two elements of the map: road width and road centre line. The matching is then approached as an optimisation problem trying to minimise given error criterion. This measure takes into account the 2D position of the vehicle, its heading and the road width.

2.3 Automatic perception enhanced by map data

The essential part of an autonomous systems is perception. Having merely localised itself is not of great value if vehicle's surroundings are not perceived and understood. To detect obstacles is the first goal of a perception module. Foresee their movements and estimating their intentions as well as responding to them is important too. The importance of this topic is reflected in the number of researchers working on it and on the means granted to finance such research.

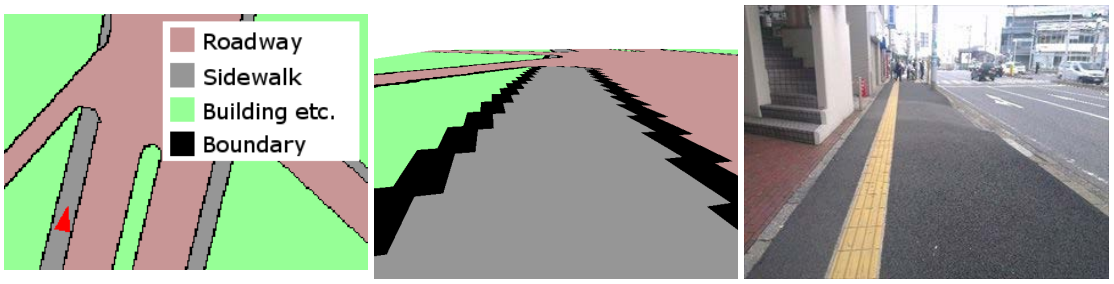
There are plethora of methods for scene perception, but only a small selection use maps and we will focus on them. They are diverse, so we tried to regroup them by the main technology they use to perceive their environment. Below presented methods have one aspect shared across all of them — maps. We regrouped here various usages of map data in perception systems for intelligent vehicles.

Maps have always been an interest point for lots of researchers working in the field of robotics. General-purpose cartographic maps, i.e. not created on purpose by a mapping technique, present a different story and became popular quite recently, but seem to stay there for a long time. One of the most important reasons is that the use of hybrid systems has been recognised as a method of improving the performance (Broggi, Bombini, et al. 2010). Under this term one understands often coupling the sensors like camera and lidar, but generally we can speak of a hybrid system if the information used to perceive the environment originates from more than one source. This information source can be a map database.

The basic reason for the popularity of such hybrid systems is that sensing devices reached a level of progress where they barely advance. This limitation being hit, only enhancing the algorithms can be a solution. The way is to use more sensors, so that the disadvantages of one device can be alleviated by the other one. Ultimately, additional hardware is an onerous solution. On the other hand, maps are available even for free and can be used as virtual sensors.

2.3.1 Map as a virtual sensor

In order to palliate the problem of lacking information, lots of research works use virtual sensors. A perfect example of a virtual sensor is, as we have just mentioned, the use of map knowledge. Various applications that exploit map prior information have been envisioned. For instance, S. Lefèvre, Laugier, and Ibañez-Guzmán (2011) use it in order to estimate the intentions of car drivers at road intersections. Strictly speaking, this is not a perception algorithm, but it can be seen as one if we consider that the map, that is the main data source, is a virtual sensor. Then, the authors exploit the geometrical and topological properties of road intersections that the map contains. This data together is used to infer drivers' intended manoeuvres. The advantage of estimating the vehicle behaviour is to be able to react in advance in case of a possibly dangerous situation.



(a) Map with robot position. (b) Map projected on image plane. (c) Actual camera capture.

Figure 2.4 – Camera-based approach enhanced with map data. Source: Irie and Tomono (2013).

2.3.2 Vision-based perception enhanced with maps

The majority of research in automatic perception is based on mimicry of human drivers. While driving, our main source of information are our eyes, so vision-based algorithms are methods that come naturally when devising an autonomous car. Unfortunately, the imitation of driver's behaviour ends there, on using cameras. Several authors have gone a step further however and tried to use map data as we, humans, use our memory.

An interesting approach for improving the performance of camera-based algorithms concerns the visibility of interest points. Alcantarilla et al. (2011) presented a machine learning method to predict the visibility of known 3D points with respect to a query camera. The approach has been applied to large-scale urban environments and, at bottom, it goes back to exploiting geometric relationships between the 3D map and camera poses. Additionally, the algorithm takes advantage of appearance information from multiple neighbouring cameras. Predicting visible points shows two at least two immediate benefits. Firstly, knowing the visible zones permits to focus on them and limit necessary computation, which in turn speeds up the whole process. Secondly, limiting the amount of processed data proves beneficial, both in terms of robustness and accuracy, for the data association between known points and features detected by the camera.

Another novel approach for traffic perception limits itself to use a single-camera system. The method proposed by Irie and Tomono (2013) is intended for mobile navigation of outdoor robots. The approach exploits digital street maps along with the robot position and prior knowledge of the environment, as illustrated by Figure 2.4. The image processing part uses the technique of *superpixels*, i.e. an input image is over-segmented and then these superpixels are grouped into various semantic classes, e.g. carriageway, pavement, wall etc. The algorithm is divided in two complementary parts: classification and localisation. The first part is formulated as an energy minimisation problem in which the authors have employed graph cuts to estimate the optimal class for each superpixel of the image. The observation are combined with the prior information coming from the map using the maximum a posterior (MAP) estimation. Due to the fact that erroneous information from map can lead to false recognition, the localisation information is incorporated into the classification result. The authors have used the map from OpenStreetMap (OSM) project. For the needs of their method, they extracted information about roadway surface, buildings and sidewalks.

Cappelle et al. (2012) use a 3D GIS database with geo-localised images for both localisation and perception. In order to perceive dynamic obstacles in the vehicle environment, the approach exploits the differences between real (acquired) image and virtual (from database) ones. Images acquired by the on-board camera may contain obstacles which are absent in the 3D model; when the inverse situation happens, the map is probably faulty.

2.3.3 Perception using laser-range sensors

An important application in road perception is the detection of road lanes. First works in the automotive domain used 2D lidar data. By exploiting provided sensor information, Ogawa and Takagi (2006) proposed a method detecting lane marks and other objects. The novelty in their approach was to use both range and reflectivity data. The method applies an Extended Kalman Filter (EKF) based on the movement of the vehicle and the detected lane position. The problem of detecting pedestrians which are much less predictable in their movement has been the subject of many works. Among others, the same authors, Ogawa, Sakai, et al. (2011), presented an approach for pedestrian recognition only using an on-vehicle lidar.

A complementary method for vehicle position detection was described in (Takagi et al. 2006). A 2D lidar is used as a forward object detection sensor. In addition, the authors describe a method of coordinating lidar-based detections with a map in order to create a highly accurate navigation system.

The first research work that used OpenStreetMap (OSM) data for all parts of an autonomous vehicle has been done by Hentschel and Wagner (2010). The map knowledge was integrated into robotic tasks, ranging from localisation and trajectory planning to autonomous vehicle control. The authors went even further by proposing to apply standardised geodata from the OSM project as the environmental representation for intelligent vehicles. The idea of the approach is based on detecting surrounding buildings with a lidar sensor. Then, this information is combined with the extracted map data in order to obtain fine-grained localisation estimate. The authors opted for a solution using a GPS position fix filtered using Kalman filtering together with wheel odometry and IMU data. The pose obtained in such a manner is then integrated into a particle filter. In order to combine the data from a lidar (in form of 3D point clouds) with a map, the method of virtual scans has been employed. This approach permitted the authors to extract two-dimensional landmark information about, e.g. vertical planes, from a 3D scan.

Velodyne lidar is a powerful sensor capable of providing over 1 million cloud points per second. Many perception algorithms take advantage of the high level of detail, long range and high precision of clouds obtained by this device. The team working on MuCAR-3 autonomous ground vehicle presented an efficient lidar-based 3D object perception method (Himmelsbach, Müller, et al. 2008). In further work (Himmelsbach, Luettel, et al. 2011), this approach has been developed to enable the automotive system to navigate autonomously.

Providing an exact description of the environment and understanding the scene in which the vehicle evolves has been the subject of (Stiller and Ziegler 2012). Situation recognition algorithm is based on Markov logic networks and employs as well topological and geometrical reasoning. The choice of trajectory is done based on a quality measure. This measure takes into consideration factors like driver safety, passenger comfort and, obviously, the efficiency of following the reference path. This approach was designed for a priori unknown environments and implemented on the AnnieWAY vehicle that won the Grand Cooperative Driving Challenge. Given map priors, the authors proposed another algorithm that considers map knowledge in order to improve the driving performance (Ziegler et al. 2014). The map is highly usage-specific and contains hints about the road priority or speed limitations. The main geographical knowledge is a database of geo-referenced visual landmarks, such as road signs and markings.

On the other hand, one can mention some works that state explicitly that the use map data, lidars or GNSS sensors is not the best way to move on. Geiger, Lauer, Wojek, et al. (2014) suggests as a solution

a bio-mimetic approach, based solely on visual clues.

[This page intentionally left blank.]

Chapter 3

Data fusion using belief functions theory

"Amicus certus in re incerta cernitur."
"A certain friend is distinguished in an uncertain affair."

Marcus Tullius Cicero, *Amicitia (64,8)*

Contents

3.1 Introduction

As a note to the reader, we suggest that a person experienced with the theory of belief functions skip a part of this chapter and go directly to Section 3.6 explaining the advantages of this theory in our application.

Any system that observes its environment has to deal with data obtained from one or many sources of information. Each source can be generally subject to noise introducing imperfections or aberrant values. For this reason, the combination of various data sources together or fusion of information over time can significantly improve the quality of the resulting information. Data fusion has been employed in a variety of applications, ranging from meteorology (Denœux 1989), medicine (Yi et al. 2014), finance, military operations (Mahler 2007) through industry (Destercke 2008) or even ecology and paleontology (Peng et al. 2011) and, last but not least, robotics (Siciliano and Khatib 2008). In this chapter, we will present a theoretical tool, called belief functions theory (BFT), used routinely for the fusion of uncertain information. A focus will be put, firstly, on the methods for combining information from multiple data sources and, secondly, on the decision making. We will also study how the uncertainty and ignorance can be modelled and taken into account in the fusion process. Finally, a special attention will be drawn to the procedures allowing for information ageing, such as discounting, updating and revision.

The introduced formalism of belief functions will be accompanied by references and comparisons, where applicable, to the following popular frameworks for data fusion:

- probability theory (Bernoulli 1713),
- fuzzy set and possibility theory (Zadeh 1978).

As the theory of belief functions have been chosen in this thesis as the main tool for information fusion, we will provide a detailed description of this formalism and its tools. An overview of several fusion rules, each presenting some advantages in different contexts, will be given as well.

For starters, we will focus on defining important notions and describing common vocabulary for characterising information. We will subsequently introduce the basics of mathematical formalisms used for information fusion. These will serve us as a basis for the rest of the chapter where we will handle the details of the Dempster–Shafer theory.

3.2 Addressing data fusion problem

Reading the following sections, the reader should bear in mind that this chapter handles information fusion, in comparison to sensor fusion being a subset of the former. We diverge from the application in the domain of intelligent vehicles in order to make a better abstraction of underlying physical behaviour and sensor structure. In the rest of this dissertation, the term *information fusion* will be used interchangeably with *data fusion*, even if one can argue that the latter is a restriction of the former.

Data fusion, being easily understood, has unfortunately no single definition. The proposed definitions of this term are legion, all more or less conveying the same message tainted with some domain-specific details. For instance, in Khaleghi et al. (2013), one of the mentioned definitions says that data fusion is a "multi-level, multifaceted process handling the automatic detection, association, correlation, estimation, and combination of data and information from several sources". Wald (1999) opted for an even more general definition abstracting from the application and details of this procedure:
Definition 1 (Data fusion)**.** Data fusion *is a formal framework in which are expressed means and tools for the alliance of data originating from different sources. It aims at obtaining information of greater quality; the exact definition of "greater quality" will depend upon the application.*

Generally, one can separate the fusion process into distinct parts:

- modelling,

- combination,
- decision.

The meaning of the term modelling is twofold. Firstly, it describes the choice of the theoretical formalism used for data fusion. In our case, we chose the Dempster–Shafer theory for the reasons described further in Section 3.6. The second interpretation is related to the level of detail with which the data is modelled in the chosen framework. Moreover, creating the data model comprises another crucial step of characterising the values attributed to elements of the model (e.g. mass or probability) through the process of automatic learning. In our case, this stage would be encapsulated by the sensor models, described in Chapter 4.

Under the term combination hides what can be sometimes referred to as fusion itself. This stage is defined by the combination rule, which is chosen depending on the data so that the final information be of *greater quality*. The choice of the fusion operator is a crucial one and may take into account the information about the data sources themselves, which we call the meta-information, i.e. information about information. Section 3.5.2 describes a few fusion rules and explains, where possible, which operator should be used in what circumstances and why.

Final part of the information fusion, the decision making translates the result of the combination into a final decision. Some popular tools for decision making are, for example, the maximum likelihood (ML) method or the maximum a posteriori (MAP) in Bayesian statistics or the pignistic transform in Dempster–Shafer theory (Smets 1994). It is often desirable that a data fusion framework should be able to decide that no decision could have been taken. This is for example the case of incoherent transactions in the theory of imprecise probabilities (Miranda 2008; Cooman, Quaeghebeur, and Miranda 2009).

3.2.1 Role of data fusion

Information fusion is a process permitting the combination of pieces of information. This data can originate from one source and be combined over time or from multiple sources and be blended together synchronously. In the first case, we talk about temporal fusion, whilst in the second one — multi-source fusion. An approach that fuses data from multiple sources over a period of time, will be denoted under the term of multi-source temporal fusion. In the present dissertation, this hybrid approach will be the main topic of interest, since we deal with a vehicle equipped with multiple sensors that are to be combined over time.

The aim of information fusion is to obtain a better estimation of the state of measured entity than given by initial data. For intelligent vehicles, it means, for instance, localising the vehicle with more accuracy or detecting obstacles in the surrounding environment with higher levels of confidence. Another purpose of the data fusion is to make a decision. By combining multiple sources of information, one can enhance the results of a classification or a diagnosis, thus improving the quality of the decision. In this thesis, we are concerned by both of these aims.

3.2.2 Important terms

In the following sections and chapters, we will characterise the data being fused as well as the combination rules acting on different types of data. Generally, the data fusion deals with the problem of treating imperfect data in order to improve its quality. Information is perfect when it is precise and certain. The choice of the best fusion system will depend on the knowledge we have about these imperfections.

Imperfections can be due to imprecision, incompleteness, uncertainty and inconsistency, terms that will be defined subsequently. According to their type and severity, one will be inclined towards one fusion method or another. In order to perform this choice effectively, one has to describe them in a clear manner. The possible defects of a piece of information are therefore described in the next paragraphs using generally accepted definitions.

Definition 2 (Uncertainty)**.** Uncertainty *attributed to a piece of information describes the degree of conformance between the information and the reality. This property results from a lack of information about the world necessary for deciding if the statement is true or false. In certain manner, uncertainty evaluates the relation between the information and our knowledge about the world (Smets 1997).*

As an example of data uncertainty, one can mention the case of a military system detecting planes and missiles, the uncertainty of the sensing system manifests itself when targets cannot be distinguished or are not detected systematically (Mercier, Quost, and Denœux 2006).

Definition 3 (Ignorance)**.** Ignorance *describes the lack of information. In the case where no information at all is available, one talks about* total *or* complete ignorance.

In some cases, the ignorance can be assimilated to uncertainty. This is however usually undesired and is often due to the low expressiveness of the formalism used for the fusion.

As opposed to uncertainty and ignorance which describe our knowledge about the world, imprecision and inconsistency are related to the content of the conveyed statement and are properties of the information itself.

Definition 4 (Imprecision)**.** *Information* imprecision *measures quantitatively the imperfection of this datum. It can concern the lack of exactness in describing size, quantity, duration or other measure.*

In an object perception system, the imprecision is, for instance, an error in the estimated position of the object.

To illustrate the difference between these properties, let us consider the following statements:

1. This family has at least two children and I am sure about it.
2. This family has three children but I am not sure about it.
3. This family has between two and four children but I am not sure about it.
4. I do not know this family.

In the first case, the number of children is imprecise but certain. In case 2, this number is precise but uncertain. Case 3 is an example where both imprecision and uncertainty coexist. Finally, the fourth case demonstrates the total ignorance about the number of children.

Apart from describing the datum itself, the information sources should be characterised as well.

Definition 5 (Conflict)**.** *The* conflict *is a characteristics of two or more sources whose information implies incompatible or contradictory interpretations. The* degree of conflict *is sometimes used in order to quantify this particularity.*

A related term, internal conflict, is also used from time to time in order to describe the contradiction conveyed by a single source in the information it transmits (Schubert 2011).

As opposed to the conflicting sources, the information delivered by different sources can be also redundant. This term can be found in two contexts, on the one hand, when the combined information is of *same quality* (cf. Definition 1) as the information from a single source. On the other hand, it describes complementary sources in which data fusion algorithms should be able to exploit this redundancy to

reduce their efforts and, thence, improve the overall quality of the resulting information (Khaleghi et al. 2013).

Among other imperfections of the fused data, one should mention at least the incompleteness, such as limited field of view of a sensor, and the ambiguity. In the precedent example of target identification, a very vague piece information, giving place for a misinterpretation of a plane as a missile, would witness of the data imprecision, but also of the ambiguity it provokes.

3.3 Probability theory

Hereunder, we present the theory of probability, *probably* the oldest and the best known of all information fusion techniques as well as the oldest theory for management of uncertainty. It will serve us the purpose of introducing common vocabulary used as well in belief functions theory. This theory dates back to the beginning of the 18th century and the art of conjecturing (*ars conjectandi*) by Bernoulli (1713). The main concern of this theory are random events, random variables and their evolution over time.

Here, we are interested in the discrete probability theory handling events with an outcome out of a finite sample space Ω.

Definition 6 (Probability mass)**.** *A probability mass function* p *on the finite space* Ω *is a non-negative mapping* $\mathrm{p}\colon \Omega \to [0,\, 1]$*, such that:*

$$\sum_{\omega\in\Omega} \mathrm{p}(\omega) = 1 \tag{3.1}$$

Definition 7 (Event)**.** *A subset* $A \subseteq \Omega$ *is called an event.*

Definition 8 (Probability measure)**.** *Given a probability mass* p*, the probability measure* P *of the event* A *is:*

$$\mathrm{P}(A) = \sum_{\omega\in A} \mathrm{p}(\omega) \tag{3.2}$$

$\mathrm{P}(A)$ *is an evaluation of the likelihood that the event* A *will occur.*

The two following axioms should be verified by any probability measure P:

Axiom 1 (Additivity)**.**

$$\forall A, B \subseteq \Omega\colon\ \mathrm{P}(A \cup B) = \mathrm{P}(A) + \mathrm{P}(B) - \mathrm{P}(A \cap B) \tag{3.3}$$

Axiom 2 (Duality)**.**

$$\forall A \subseteq \Omega\colon\ \mathrm{P}(A) = 1 - \mathrm{P}(A^c) \tag{3.4}$$

where A^c *denotes the complement of the set* A*.*

Combining information with the Bayes' rule In probability theory, in order to combine the knowledge about an event, the Bayes' rule is typically applied. Let $\mathrm{P}(A)$ be the probability of the event A. This value quantifies the degree of belief or the objective probability, depending on the interpretation

given to the probability measure, that a particular arbitrary element ω of Ω[1] belongs to a particular set A (Smets 1999). When a new piece of information arrives stating that ω belongs to $B \in A$ and that $\mathrm{P}(B) > 0$, the probability measure P must be updated into P_B. P_B takes into consideration the fact that $\omega \in B$. However, it still quantifies the same event as previously. In this case, one can obtain the value of P_B applying the Bayes' rule of conditioning:

$$\mathrm{P}_B(A) = \mathrm{P}(A|B) = \frac{\mathrm{P}(A \cap B)}{\mathrm{P}(B)} \tag{3.5}$$

From the above equation, the Bayes' theorem has been derived in order to relate current to prior evidence (Bayes and Price 1763):

$$\mathrm{P}(A|B) = \frac{\mathrm{P}(B|A) \cdot \mathrm{P}(A)}{\mathrm{P}(B)} \tag{3.6}$$

Advantages Since the probability theory is an established tool, well-known in various communities, a lot of theoretical and practical tools are available. Its benefits and drawbacks have been well studied. More importantly, the probability theory is a *de facto* standard technique for information fusion. Among the practical reasons for which the probability theory prevails, one can name the simplicity, the effectiveness and the fact of being light-weight in computational terms (cf. further sections). Furthermore, the probability theory is well adapted in case where there are huge amounts of data and statistical reasoning is fully justified.

Disadvantages The major drawback of the probability theory is its manner of modelling the ignorance. Let consider a set of possible outcomes $\Omega = \{A, B\}$ along with a probability mass $\mathrm{p}(A) = 0.5$, $\mathrm{p}(B) = 0.5$. In fact, for these two equiprobable events, one cannot distinguish between the two situations given no additional information:

- The likelihoods of events A and B occurring are equal.
- There is no knowledge about the events A and B.

3.4 Possibility theory

The possibility theory is a response to the above mentioned inability of the probability theory to deal with uncertainty. It was first introduced by Zadeh (1978) as an extension of the theory of fuzzy sets, further developed by Dubois and Prade (1985).

This theory is defined in terms of possibility distributions.
Definition 9 (Possibility distribution)**.** *Given a random variable R taking values in space Ω, a possibility distribution is a mapping $\pi : \Omega \to [0, 1]$ from the space Ω to the unit interval. π quantifies the uncertainty about the variable R.*

Having defined the possibility distribution π, several set-functions can be defined:
Definition 10 (Possibility measure)**.** *The possibility measure $\Pi(A)$ of an event A expresses the extent to*

[1] ω is a priori not located in any of the sets of A.

which this event is consistent with the available evidence, i.e. plausible.

$$\Pi(A) = \sup_{\omega \in A} \pi(\omega) \tag{3.7}$$

Definition 11 (Necessity measure)**.** *The degree of necessity $N(A)$ of an event A is an evaluation of the certainty that this event will occur.*

$$N(A) = 1 - \Pi(A^c) \tag{3.8}$$

Definition 12 (Sufficiency measure)**.** *Sufficiency, also called guaranteed possibility, evaluates the extent to which all states of universe Ω where A occurs are plausible.*

$$\Delta(A) = \inf_{\omega \in A} \pi(\omega) \tag{3.9}$$

The possibility measure Π must satisfy the following axiom:
Axiom 3 (Composability)**.**

$$\Pi(A \cup B) = \max\left(\Pi(A),\, \Pi(B)\right) \qquad \forall A, B \subseteq \Omega,\ A \cap B = \emptyset \tag{3.10}$$

This axiom is the analogue of the additivity axiom of the probability theory (cf. Axiom 1).

Under the *closed-world hypothesis*, the mapping Π should satisfy as well:
Axiom 4 (Closed-world assumption)**.**

$$\Pi(\emptyset) = 0 \tag{3.11}$$

The closed-world assumption means that Ω is an exhaustive description of possible states of the world (possible outcomes) and that no belief weight is attributed to elements outside of Ω. Analogously, the open-world assumption would refer to the case where Ω is not exhaustive and there are some possible outcomes that are unknown.

Furthermore, one can require that a possibility measure is conflict-free, i.e. no contradiction arises between the hypotheses described by Π. In this case, another axiom should be satisfied:
Axiom 5 (Conflict-free)**.**

$$\Pi(\Omega) = 1 \tag{3.12}$$

3.4.1 Combination rules in possibility theory

Minimum rule If π_1, π_2 denote two possibility distributions obtained from two reliable sources, the standard conjunctive combination rule between these two distributions is the pointwise minimum, defined as follows (Dubois and Prade 1988):

$$\pi_\wedge(A) = (\pi_1 \wedge \pi_1)(A) = \min(\pi_1(A), \pi_2(A)) \qquad \forall A \in \Omega \tag{3.13}$$

Maximum rule The maximum rule is the disjunctive counterpart of the minimum rule if the combined information comes from sources of which only one is reliable.

$$\pi_{\vee}(A) = (\pi_1 \vee \pi_1)(A) = \max(\pi_1(A), \pi_2(A)) \qquad \forall A \in \Omega \tag{3.14}$$

Adaptative rule The adaptative rule was proposed in order to solve the problem of choosing between the two antagonistic rules described above. This rule needs to define the degree of consensus $h(\pi_1, \pi_2)$ between two sources.

$$(\pi_1 \underset{AD}{\circledast} \pi_1)(A) = \max\left(\frac{\pi_{\wedge}(A)}{h(\pi_1, \pi_2)}, \min\left(1 - h(\pi_1, \pi_2), \pi_{\vee}(A)\right)\right) \qquad \forall A \in \Omega \tag{3.15}$$

Indeed, the minimum rule corresponds to the situation where h is close to 0, whereas the maximum rule — when h is close to 1. The advantage of the adaptative rule is that it permits to move from one method to another in a continuous manner instead of switching abruptly. Moreover, in such a case, one has to define a threshold for h where this transition would take place.

3.4.2 Equivalence with probability theory

The possibility theory is the first mathematical formalism that successfully generalised the probability theory. Given a probability mass p, we can create an equivalent possibility distribution π, simply by applying the following expression:

$$\pi(\omega) = [\mathrm{p}(\omega),\ \mathrm{p}(\omega)] \qquad \forall \omega \in \Omega \tag{3.16}$$

Moreover, the degree of ignorance can be expressed. The above stated possibility distribution communicates that the piece of information concerning ω is totally certain. On the other hand, complete ignorance about hypothesis ω is conveyed through $\pi(\omega) = [0,\ 1]$.

3.4.3 Advantages

The possibility theory is an elegant generalisation of the probability theory. By introducing an interval-based measure instead of a single value of probability, this theory enables much wider range of expressiveness, compared to probabilities, when handling uncertain data. Also, when regarded from the computational perspective, this theory rests attractive, as it does not introduce any expensive operations (e.g., no exponential explosion).

3.4.4 Disadvantages

As with all likelihood measures which are not directly translatable into probabilities, it is not obvious how to make decisions when using this formalism. That is, it is application-dependent whether the right measure to use is the possibility or the necessity, or some mix of both.

3.5 Belief functions theory

The theory of belief functions, also known as Dempster–Shafer theory (DST), was proposed by Dempster (1968) and developed, among others, by Shafer (1976) and Smets (1994; 2005). This formalism

gained its popularity thanks to various interesting properties. DST not only generalises the probability theory, but the possibility theory as well.

3.5.1 Fundamentals

In the belief functions theory, one can attribute a mass from the unit interval to any *subset* of the possible hypotheses Ω and not only to a single element.

Definition 13 (Frame of discernment)**.** *The frame of discernment (fod) is a finite set of possible outcomes* $\Omega = \{\omega_1,\ \omega_2,\ \ldots,\ \omega_K\}$ *from which a random variable* R *takes values.*

The information obtained from source S concerning the actual value taken by variable R is quantitatively described by a basic belief assignment (bba) m_S^Ω, also called a mass function.

Definition 14 (Basic belief assignment, mass function)**.** *The basic belief assignment (bba)* m_S^Ω *is defined as a mapping* $\mathrm{m}\colon 2^\Omega \to [0,\ 1]$ *from the power set of* Ω *to the unit interval satisfying the condition:*

$$\sum_{A \subseteq \Omega} \mathrm{m}_S^\Omega(A) = 1 \tag{3.17}$$

The notation $\mathrm{m}_S^\Omega(A)$ will be further simplified to $\mathrm{m}^\Omega(A)$ or $\mathrm{m}(A)$ when no ambiguity is possible.

In the DST, it is possible to express the ignorance, partial or total, about the considered random variable. The partial ignorance is expressed, for example, by assigning a non-zero value to the fod, i.e. $\mathrm{m}(A) = \mu \neq 0$. In case where the ignorance is complete, one talks about the vacuous mass function.

Definition 15 (Vacuous bba)**.** *A* vacuous bba m *is a mass function corresponding to the state of total ignorance about the variable* R*. The bba* m *satisfies then the condition* $\mathrm{m}(\Omega) = 1$*.*

In the sequel, we will use also the following vocabulary (Smets and Kennes 1994):

Definition 16 (Normal bba, regular bba)**.** *A mass function* m *for which* $\mathrm{m}(\emptyset) = 0$ *is called* normal *or* regular*.*

Definition 17 (Subnormal bba)**.** *A mass function* m *for which* $\mathrm{m}(\emptyset) \neq 0$ *is called* subnormal*.*

Definition 18 (Normalisation, degree of conflict)**.** *An operation that maps a subnormal bba* m *into a normal bba* m'*, by applying the following transformation:*

$$\mathrm{m}'(\emptyset) = 0 \tag{3.18}$$

$$\mathrm{m}'(A) = \frac{\mathrm{m}(A)}{1 - K} \qquad \forall A \subseteq \Omega, A \neq \emptyset \tag{3.19}$$

$$K = \mathrm{m}(\emptyset) \tag{3.20}$$

will be called normalisation *or* Demspter's normalisation*. The mass* K *attributed to the empty set* $\emptyset$ *before normalisation will be called the* degree of conflict*.*

Definition 19 (Focal set)**.** *Every subset* $A \subseteq \Omega$ *of the fod* Ω *for which the mass* m *takes a non-zero value, i.e.* $\mathrm{m}(A) \neq 0$ *is called a* focal set *or a* focal element*.*

Definition 20 (Categorical bba)**.** *A* categorical bba m *is a mass function corresponding to the state of complete certainty about the state of the variable* R*.* m *satisfies the condition* $\mathrm{m}(A) = 1$*. In other words, a categorical bba has only one focal set.*

Definition 21 (Simple bba)**.** *We will call* m *a* simple mass function *if it has no more than two focal elements,* Ω *being include. Such a bba will be denoted by* A^w*, where the set* A *is a focal element different*

from Ω. *The mass function* A^w *is equivalent a mass* m*:*

$$\mathrm{m}(A) = w \tag{3.21}$$
$$\mathrm{m}(\Omega) = 1 - w \tag{3.22}$$
$$\mathrm{m}(B) = 0 \qquad B \neq A,\ B \neq \Omega \tag{3.23}$$

Definition 22 (Bayesian bba)**.** *A mass function* m *will be called* Bayesian *if all its focal sets are singletons, i.e. their cardinality equals one.*

$$\mathrm{m}(A) \neq 0 \implies |A| = 1 \qquad \forall A \subseteq \Omega \tag{3.24}$$

Definition 23 (Consonant bba)**.** *A mass function* m *is called* consonant *if and only if all its focal sets are nested, i.e. a non-zero mass value attributed to set* A *implies that all its supersets have a non-zero mass as well.*

$$\mathrm{m}(A) \neq 0 \implies \mathrm{m}(B) \neq 0 \qquad \forall A \subset \Omega,\ \forall B \supset A \tag{3.25}$$

For the majority of combination operators, a precondition is that two masses to be fused use the same common frame of discernment (fod). As another application, it is often necessary or easier to work with the description of the possessed knowledge at certain level of details. For instance, in a very detailed representation consisting of many classes, one can require to lessen the cognitive burden by introducing generalised categories. Or inversely, supplementary information may be more detailed than the current model. For these purposes, the DST provides tools needed for transforming one fod into another.

In order to convert mass function m^{Ω_1} defined on fod Ω_1 to another mass function m^{Ω_2} on Ω_2, one should use refining functions.

Definition 24 (Refining)**.** *A refining* r *is a one-to-many mapping from* Ω_1 *to* Ω_2 *(Shafer 1976) defined as the following:*

$$\begin{aligned} r &: 2^{\Omega_1} \to 2^{\Omega_2} \setminus \emptyset \\ r(\omega) &\neq \emptyset \qquad \forall \omega \in \Omega_1 \\ \bigcup_{\omega \in \Omega_1} r(\omega) &= \Omega_2 \\ r(A) &= \bigcup_{\omega \in A} r(\omega) \end{aligned} \tag{3.26}$$

The refined mass function m^{Ω_2} can then be expressed as:

$$\mathrm{m}^{\Omega_2}(r(A)) = \mathrm{m}^{\Omega_1}(A) \qquad \forall A \subseteq \Omega_1 \tag{3.27}$$

Definition 25 (Refinement and coarsening)**.** *Having defined a refining function, the fod* Ω_2 *is then termed the* refinement *of* Ω_1*, and* Ω_1 *is the* coarsening *of* Ω_2 *(Shafer 1976).*

Equivalent representations

Belief and plausibility A bba can be expressed not only by mass function m, but there are equivalent functions representing the same information. One of them is the *belief function* being a mapping

$\text{bel}\colon 2^\Omega \to [0,\,1]$ which in the Transferable Belief Model (TBM) (Smets and Kennes 1994) takes the form of:

$$\text{bel}(A) = \sum_{\emptyset \neq B \subseteq A} \text{m}(B) \qquad \forall A \subseteq \Omega \tag{3.28}$$

Since the function bel is bijective, it is possible to recover the mass function from the belief function:

$$\text{m}(\emptyset) = 1 - \text{bel}(\Omega) \tag{3.29}$$

$$\text{m}(A) = \sum_{B \subseteq A} (-1)^{|A \setminus B|} \text{bel}(B) \qquad \forall A \subseteq \Omega, A \neq \emptyset \tag{3.30}$$

A mass function m can also be expressed as a plausibility function pl (Dempster 1968):

$$\text{pl}(A) = \sum_{A \cap B \neq \emptyset} \text{m}(B) = 1 - \text{bel}(B^c) - \text{m}(\emptyset) \qquad \forall A \subseteq \Omega \tag{3.31}$$

Commonality and implicability Other equivalent representations are used less often, but their promise various advantages. For instance, the commonality and implicability function are often used when implementing respectively the conjunctive and disjunctive rule of combination (please refer to Section 3.5.2) because of their lower computational complexity of this operation.

$$\text{q}(A) = \sum_{B \supseteq A} \text{m}(B) \qquad \forall A \subseteq \Omega \tag{3.32}$$

Its analogue, the implicability function is defined as:

$$\text{b}(A) = \sum_{B \subseteq A} \text{m}(B) = \text{bel}(A) + \text{m}(\emptyset) = 1 - \text{pl}(A^c) \qquad \forall A \subseteq \Omega \tag{3.33}$$

Similarly to belief functions, bba m can be recovered from any of these functions. For instance,

$$\text{m}(A) = \sum_{B \supseteq A} (-1)^{|B| - |A|} \text{q}(B) \qquad \forall A \subseteq \Omega \tag{3.34}$$

$$\text{m}(A) = \sum_{B \subseteq A} (-1)^{|B| - |A|} \text{b}(B)\,\text{b}(A) \qquad \forall A \subseteq \Omega \tag{3.35}$$

Canonical decomposition The notion of decomposition is inherently linked to the separability of mass functions. Shafer (1976, Chapter 4) defined a separable bba as the result of conjunctive combination of simple bbas. A later work of Smets (1995) extended the idea to any non-dogmatic bba that can be uniquely represented as the conjunctive combination of generalised simple basic belief assignments (GSBBAs). Denœux (2008) renamed this operation to canonical conjunctive decomposition and proposed its disjunctive equivalent.

Definition 26. *Canonical conjunctive decomposition*

$$\text{m} = \bigcap_{A \subset \Omega} A^{w(A)} \tag{3.36}$$

The weights $w(A) \in (0, +\infty)$ *define the* canonical conjunctive decomposition *and can be computed using the following formula:*

$$w(A) = \prod_{B \supseteq A} \mathrm{q}(B)^{(-1)^{|B|-|A|+1}} \tag{3.37}$$

In (Denœux 2008), the canonical *disjunctive* decomposition was proposed as a counterpart of the above described operation.

Definition 27. *Canonical disjunctive decomposition*

$$\mathrm{m} = \bigcirc\!\!\!\!\!\bigcup_{A \subset \Omega, A \neq \emptyset} A_{v(A)} \tag{3.38}$$

The weights $v(A) \in (0, +\infty)$ *define the* canonical disjunctive decomposition *and can be computed using the following formula:*

$$v(A) = \bar{w}\big(\bar{A}\big) \tag{3.39}$$

where the weight function $\bar{w}$ *is associated to the negation (or complement)* $\bar{m}$ *of* m*, the former being defined as a function verifying* $\bar{m}(A) = m\big(\bar{A}\big)$.

Pignistic probability Decision making in DST imposes from time to time that a mass function be transformed into a probability function (Smets 2005). Smets (1994) proposed the so-called *pignistic transformation*. Pignistic probability betP was defined as for $B \subseteq \Omega$:

$$\mathrm{betP}(B) = \sum_{A \subseteq \Omega} \mathrm{m}(A) \cdot \frac{|B \cap A|}{|A|} \tag{3.40}$$

where $|A|$ denotes the cardinality of set A.

3.5.2 Combination rules

Conjunctive rule The conjunctive rule of combination (CRC) is one of the widely used combination operators in the DST. It is used to combine two mass functions m_1, m_2 from two distinct and reliable sources.

$$(\mathrm{m}_1 \textcircled{\cap} \mathrm{m}_2)(A) = \sum_{B \cap C = A} \mathrm{m}_1(B)\,\mathrm{m}_2(C) \qquad \forall A \subseteq \Omega \tag{3.41}$$

It can be equivalently defined in terms of commonality functions with a simple notation and the complexity linear in the number of elements of 2^Ω (Denœux 2008):

$$(\mathrm{q}_1 \textcircled{\cap} \mathrm{q}_2)(A) = \mathrm{q}_1(A) \cdot \mathrm{q}_2(A) \qquad \forall A \subseteq \Omega \tag{3.42}$$

Disjunctive rule The disjunctive rule of combination (DRC) may be used to combine two distinct pieces of evidence m_1, m_2 under the assumption that only one of the two information sources is reliable

(Smets 2008). DRC is defined by:

$$(\mathrm{m}_1 \textcircled{\cup} \mathrm{m}_2)(A) = \sum_{B \cup C = A} \mathrm{m}_1(B)\,\mathrm{m}_2(C) \qquad \forall A \subseteq \Omega \tag{3.43}$$

or, equivalently (Denœux 2008):

$$(\mathrm{b}_1 \textcircled{\cup} \mathrm{b}_2)(A) = \mathrm{b}_1(A) \cdot \mathrm{b}_2(A) \qquad \forall A \subseteq \Omega \tag{3.44}$$

Yager's rule The Yager's rule of combination was proposed as a response to the critics of the conjunctive rule in case of non-reliable sources of information. Since this operator considers that the furnished information can be unreliable, the conflict mass $\mathrm{m}(\emptyset)$ is therefore transferred to the ignorance $\mathrm{m}(\Omega)$. As a major difference with respect to conjunctive and disjunctive rules, this operation is not associative. The author explicitly highlighted the importance of a rule that is dependent on the order of combination (Yager 1987).

$$\begin{aligned}
&(\mathrm{m}_1 \underset{Y}{\circledast} \mathrm{m}_2)(A) = (\mathrm{m}_1 \textcircled{\cap} \mathrm{m}_2)(A) && \forall A \subset \Omega, A \neq \emptyset \\
&(\mathrm{m}_1 \underset{Y}{\circledast} \mathrm{m}_2)(\emptyset) = 0 \\
&(\mathrm{m}_1 \underset{Y}{\circledast} \mathrm{m}_2)(\Omega) = (\mathrm{m}_1 \textcircled{\cap} \mathrm{m}_2)(\Omega) + (\mathrm{m}_1 \textcircled{\cap} \mathrm{m}_2)(\emptyset)
\end{aligned} \tag{3.45}$$

Cautious conjunctive rule and bold disjunctive rules Cautious conjunctive and bold disjunctive rules of combination were proposed by Denœux (2006). They both possess an important property of *idempotence*, i.e. a mass function combined with itself results in itself. This is an essential characteristics when dealing with non-distinct sources of evidence. The cautious rule, similarly to the conjunctive rule, should be used when combining reliable sources, whereas the bold rule is used when at least one of the sources is reliable. The cautious $\textcircled{\wedge}$ and bold $\textcircled{\vee}$ rules are expressed in terms of, respectively, conjunctive and disjunctive canonical decompositions:

$$\mathrm{m}_1 \textcircled{\wedge} \mathrm{m}_2 = \bigcirc\!\!\!\!\bigcap_{A \subset \Omega} A^{w_1(A) \wedge w_2(A)} \tag{3.46}$$

$$\mathrm{m}_1 \textcircled{\vee} \mathrm{m}_2 = \bigcirc\!\!\!\!\bigcup_{A \subseteq \Omega, A \neq \emptyset} A_{v_1(A) \vee v_2(A)} \tag{3.47}$$

3.5.3 Mass discounting

When dealing with beliefs or mass functions, one needs to take into account the level of certainty that can be attributed to the information source and hence to the processed piece of information. To achieve this, the information discounting is used. Classically, the discount of information in the Dempster–Shafer theory is performed using uniform discounting. This operation is well adapted for discounting all information types from a source in the same manner. It is however unsuited when dealing with classes that need varying level of discount.

Uniform discounting The most commonly used form of discounting operation given discount factor α has been proposed by Shafer (1976, pp. 251–255) and will be subsequently called *classical* or *uniform discounting*:

$$^{\alpha}\mathrm{bel}(A) = (1-\alpha)\,\mathrm{bel}(A) \qquad \forall A \subsetneq \Omega \tag{3.48}$$

This operation can be expressed equivalently using mass functions as:

$$^{\alpha}\mathrm{m}(A) = (1-\alpha)\,\mathrm{m}(A) \qquad \forall A \subsetneq \Omega \tag{3.49}$$

$$^{\alpha}\mathrm{m}(\Omega) = (1-\alpha)\,\mathrm{m}(\Omega) + \alpha \tag{3.50}$$

The advantage of the uniform discounting is its simplicity. Only one discount factor has to be defined; it can be easily learnt from data or derived empirically. However, the simplicity of this approach rests as well its biggest drawback. Excluding one or more classes from discounting is impossible. Similarly, one cannot exploit more refined information about the discount rates of some classes. As the response, several works tried to alleviate these problems.

Among other works, Pichon, Dubois, and Denœux (2012) devoted some research to the subject of information correction schemes by proposing a strategy taking into account the source's relevance and truthfulness. In (Klein and Colot 2010), the discounting operation is automatised by introducing a measure of conflict, called dissent criterion, and using it to compute the discounting rate. The aforementioned contextual discounting has been polished and refined in (Mercier, Quost, and Denœux 2006; Mercier, Denœux, and Masson 2006) where a method for automatic learning of discount rates out of a data set was presented.

3.5.4 Information updating and revision

Other mechanisms worth to be mentioned comprise among others the data revision. It have been studied as well in the context of the evidence theory. A review of existing revision rules can be found in (Ma et al. 2011), along with an extension of one of them able to cope with inconsistency between prior and input information.

In order to illustrate the problem of revision, let us consider the following example. A die has been tossed. You assess the probability that the outcome is "six". Then a reliable witness says that the outcome is an even number. How do you update the probability that the outcome is "six" taking in due consideration the new piece of information. This scenario corresponds to a revision as the probability is modified to take into account a new piece of information (Smets 1999).

One of the classical rules for information revision, is the Jeffrey's rule of conditioning derived from the Bayes' rule (cf. Section 3.3). Using similar notation as in Section 3.3, let P_1 correspond to the initial knowledge and P_2 — to the new piece of information. The Jeffrey's rule of conditioning results in the revised probability P_1 that can be calculated through the following equation (Jaffray 2008):

$$\mathrm{P}_3(A) = \mathrm{P}_1(A|B) \cdot \mathrm{P}_2(B) = \frac{\mathrm{P}_1(A \cap B)}{\mathrm{P}_1(B)} \cdot \mathrm{P}_2(B) \tag{3.51}$$

where $\mathrm{P}_1(A|B) = 0$ if $\mathrm{P}_1(B) = 0$.

The Jeffray's rule has been adapted to the theory of belief functions by Smets (1999). If the initial

knowledge is represented by a bba $\mathrm{m}_1^{\Omega_1}$ and the new information by $\mathrm{m}_2^{\Omega_2}$, then the revised mass function m_3 can been obtained in the following manner. Firstly, the constraint $\forall \omega \in \Omega_2 \colon \mathrm{bel}_3(\omega) = \mathrm{bel}_2(\omega)$ should be satisfied. Then, let define for every $A \in \Omega_1$, the mapping $B(A) \in \Omega_2$ be the smallest element of Ω_2 such that $A \subseteq B(A)$ and there is no other $B' \in \Omega_2$ such that $A \subseteq B' \subseteq B(A)$. Similarly, let $b(A)$ be the set of $A \in \Omega_1$ that shares the same $B(A)$.

$$\mathrm{m}_3(A) = \frac{c\left(A, B(A)\right)}{\sum\limits_{\omega \in b(A)} c\left(\omega, B(A)\right)} \cdot \mathrm{m}_2(B(A)) \qquad \forall A \in \Omega_1 \tag{3.52}$$

where $c\left(A, B(A)\right) \geq 0$ is chosen arbitrarily except that it should be positive so that m_3 is non-negative.

Jeffrey geometric and Jeffrey–Dempster rules of conditioning This somehow general rule comprises two rules specified by Smets (1999). Firstly, the so called *Jeffrey geometric rule of conditioning* defined as:

$$\mathrm{m}_3(A) = \frac{\mathrm{m}_1(A)}{\sum\limits_{\omega \in b(A)} \mathrm{m}_1(\omega)} \cdot \mathrm{m}_2(B(A)) \qquad \forall A \in \Omega_1 \tag{3.53}$$

And the *Jeffrey–Dempster rule of conditioning*:

$$\mathrm{m}_3(A) = \frac{\mathrm{m}_1(A|B(A))}{\sum\limits_{\omega \in b(A)} \mathrm{m}_1(\omega|B(A))} \cdot \mathrm{m}_2(B(A)) \qquad \forall A \in \Omega_1 \tag{3.54}$$

In this work, we decided however not to use the revision methods such as the presented rule of Jeffrey–Dempster. The reason is that the revision supposes that a new piece of information gives additional knowledge about the observed entity. Moreover, this information refines the already possessed knowledge. These assumptions are completely valid in a static framework. In our case however, the environment is dynamic. A new information may concern a different entity, e.g. a moving object, that was described by the revised mass function.

3.6 Rationale for using belief functions

As stated in the beginning of this chapter, we have chosen to use the theory of belief functions as the basic tool for information fusion. The reasons that persuaded us to employ this formalism are multiple and we will try to justify our choice hereafter.

First of all, we are persuaded that a method for explicitly modelling the ignorance is necessary. In formalisms where the lack of information is sometimes, wrongly, represented in the same way as some type of uncertain information, the methods for data fusion must compensate for this effect explicitly if possible or accept the fact that the both situations rest undistinguished. As an example, one can cite the case of equiprobable events[1] in the theory of probability. Supposing that no prior information is available, one cannot distinguish if no data is present or if the given evidence proves that the both events are equally viable.

Another advantage is that the theory of belief functions generalises both the theories of probability and possibility thus having at least the same power of expressiveness. For the former, it is trivially

[1] I.e. events to which we attribute the same probability measure.

provable that a mass function with masses attributed only to singletons (sets of cardinality 1) is equal to a probability mass function. Furthermore for the latter theory, it has been demonstrated that a necessity measure N can be expressed by belief functions (Shafer 1976). Indeed, a random set with nested focal elements, i.e. a consonant mass function, induces the N measure.

Thanks to the possibility of attributing masses to sets, one can effectively work with class hierarchies. Moreover, DST provides mechanisms for switching between these representations, namely, the refinement and the coarsening. For instance, a class object may be separated into dynamic and static objects. The corresponding frame of discernment will contain only classes dynamic and static, but the class "object" can be implied. In this way, the fusion can be applied on the level of details needed for a given task, without the need for an overly detailed or a too coarse fod.

Last of all, one can mention a somewhat subjective advantage, namely the fact that defining a mass function is intuitive. Furthermore, so it is the fusion, which permits to translate in a natural way the meta-knowledge about the information source when choosing an appropriate combination rules.

3.6.1 Disadvantages

Unfortunately, as almost each theoretical tool, also the theory of belief functions should be avoided in some cases. The computations involved in the majority of fusion operations using the DST are expensive. This comes from the fact that for a frame of discernment (fod) of size n, there are 2^n mass values. Any algorithm taking into account all these values is therefore exponential with respect to the number of hypotheses, resulting in #P-complete computational complexity (Orponen 1990). For this reason, only relatively few hypotheses can be included into the fod for the sake of practical feasibility. Some theoretical advantages are hence eclipsed by this exponential explosion.

As this complexity has been an important issue, the researchers have proposed some practical methods for computation of belief functions that permit to reduce the time complexity from exponential to linear (Barnett 2008). Among other approaches, an interesting one is a random algorithm for computation of fusion rules that is based on the hypothesis that minor deviations from exact mass values should not (Wilson 1991). All in all, the major disadvantage of computational greediness is strongly alleviated or disappears if implemented correctly.

3.6.2 Application in intelligent transportation systems

The most important reasons that convinced the authors to use the DST are as follows. The vehicle environment contains many occlusions and barely observed or non-observed zones. The representation of the unknown, inherent to the DST and missing in the theory of probability permits to handle this notion in a way. Fusion operators defined in the DST are able to manage uncertainties and incoherence conveyed by data sources. Recent work of Klein and Colot (2011) has shown by introducing a particular conflict criterion that a proper conflict analysis may be helpful to identify singular, or outlying, information sources. There has been also substantial research work on data association problems, such as multi-target tracking, which exploits conflict management (Ayoun and Smets 2001).

In a perception system, it is desirable to have a tool to manage different levels of detail (LODs), since the obtained information cannot be always interpreted clearly and precisely. In the DST, this tool is at the core of the theory. A frame of discernment can be as refined as the most detailed data obtained from the sensors, but still, it remains possible and easy to combine information which is more general

by affecting masses on non-singletons. There are also well-established methods to deal with multiple frames of discernment (Kruse and Klawonn 1994). Such management of LODs would be impossible or at least difficult with an accumulation schema. There are already some works which take advantage of the theory of evidence in the context of mobile perception (Moras, Cherfaoui, and Bonnifait 2010; Moras, Cherfaoui, and Bonnifait 2011a; Moras, Cherfaoui, and Bonnifait 2011b). In other domains, the Dempster–Shafer theory has been used as well, e.g., for visual tracking. Klein, Lecomte, and Miché (2010) presented a hierarchical combination scheme that makes use of existing fusion rules and source classification with respect to their reliability and precision. Of course, there are more research works that apply the belief functions theory to the domain of intelligent vehicles.

[This page intentionally left blank.]

Part III
Remanence in occupancy grids

This part constitutes the core of this dissertation presenting the author's contributions in the domain of intelligent transportation systems. It handles as well underlying theoretical problems and their solutions in the domain of information fusion as well as the processing of uncertain and imprecise data.

Chapter 4 presents the methods used to define sensor models adapted for an autonomous perception system and handling the uncertainty inherent to the sensor data.

Chapter 5 describes a fusion scheme that enables the incorporation of prior knowledge, coming from e.g. digital maps, into information from sensors. Besides, it describes how such a system may be applied to the problem of scene understanding in difficult urban environments.

Chapter 6 focuses on one element of the fusion process – the temporal discounting of information. A family of contextual discounting operations is proposed.

[This page intentionally left blank.]

Chapter 4

Environment modelling with evidential grids

"L'espace commence ainsi, seulement avec des mots,
avec des signes tracés sur une page blanche... "
"This is how space begins, with words only, with signs traced on a blank page... "

Georges Perec, *Espèces d'espaces (Species of spaces)*

Contents

Processing raw sensor data is an inevitable part of each perception system. However, it is obviously a difficult and more often than not inefficient approach to work with raw data throughout all the steps of

information fusion. For instance, handling directly data like point clouds, e.g. as the one in Figure 4.1a, is usually impractical and burdensome. There are however several works that successfully handle such data, and what proves the popularity of libraries like Point-Cloud Library (PCL) (Rusu and Cousins n.d.). The large amount of information and unsuitable data structures results in high computational complexity. Hence, reducing long processing times is amongst the main motivations for representing sensor data in an alternative manner.

Object-based approaches, which consist in detecting, recognising and tracking objects in the scene, although useful and semantically rich, are difficult to manage except in the case of well-known environments. In opposition, grid-based approaches can handle any kind of environment. In this dissertation, we have decided to apply an extension of occupancy grids, 2-dimensional (2D) evidential grids, to represent vehicle's environment. The methods based on occupancy grids gained a lot of support in the domain of robotics and are an efficient form of environment representation. Some authors claim that an inconvenience of the cell-based approaches is that no additional semantic value can be attributed to grid cells representing the environment. Grids can however serve as a basis for further processing steps, which can be object-based. In the following, we will present original probabilistic occupancy grids as well as their multi-dimensional enhancements.

Further in this chapter, an evidential version of occupancy grids called "perception grids", adapted for the theory of belief functions, will be introduced. These evidential occupancy grids will serve as in the sequel in order to define sensor models used to transform raw data from sensing device into a homogeneous representation of the vehicle surroundings. Such data structures will serve us in further chapters during the process of fusion of information contained in various grids.

Despite multiple benefits, occupancy and evidential grids present some challenges. Various computational problems, such as large needs for computing power are to be mentioned. They can be however diminished on highly efficient modern architectures, exploiting, for instance, many-core systems by parallel computation. Other issues can arise due to necessary coordinate transformations and the transit between local (robot-centred) and global (world-centred) reference frames.

4.1 Occupancy grids

Originally proposed by Elfes (1989) for spatial sensing and modelling for robot perception, occupancy grids have been first adopted for indoor environments. Their success in this area boosted their acceptance for the representation of outdoor scenes as well. Counterparts of occupancy grids, called certainty grids (Moravec 1988) did not meet with the same welcome and their name went into oblivion, even if they are conceptually equivalent and more elaborate on certain aspects like multiple levels of detail.

Occupancy grids are defined in (Elfes 1989) as a probabilistic tessellated representation of spatial information, more precisely a multi-dimensional random field. They store stochastic estimates of the occupancy state (free F, occupied O) of the cells in a spatial lattice. Classically, one's level of confidence of finding an obstacle in a cell is attributed to $\mathrm{P}(O)$. The probability of this cell being free is then $\mathrm{P}(F) = 1 - \mathrm{P}(O)$ In order to construct grids, the author proposed to interpret the incoming range readings from a sensor using probabilistic models. For incremental updating of grids, a procedure of Bayesian estimation has been proposed.

A large number of cells in a grid and the need of processing them have been an issue for some time due to limited computational power of computers. Recent advances in this domain have permitted

a renewal of interest of this form of environment modelling. Notably sensors like the lidar have been among the first use cases of grid-based approaches.

4.1.1 Cell dynamics

Occupancy grids are applied for various purposes in indoor, outdoor and space robotics. They are still used for localisation (Konrad, Nuss, and Dietmayer 2012) and perception, for instance, for the detection of free space (Schreier and Willert 2012). Their efficiency has been proven in other areas of vehicle perception like moving object tracking (MOT) and detection and tracking of moving objects (DATMO). An interesting and efficient approach was proposed by Coué et al. (2006). This method, called Bayesian Occupancy Filter (BOF), uses techniques of Bayesian filtering in order to robustly perceive and analyse highly dynamic environments. BOF method takes into account the uncertainty inherent to the process of estimating the environment state. Oppositely to multi-target tracking algorithms, this method addresses the problem of multiple appearances and occlusions by working with four-dimensional (including time) occupancy grid representation of the obstacle state space. In (Perrollaz 2008), the author applied probabilistic occupancy grids for obstacle detection using stereo-vision. Baig (2012) presented another dissertation on the use of probability grids for perception in a dynamic autonomous vehicle. The author used grids as a basic tool for the data fusion in a multi-sensor system. The easiness of fusion of data coming from homogeneous sensor was demonstrated in (Baig et al. 2011).

4.1.2 Multi-dimensional grids

Occupancy grids could be enhanced in various ways. There are different types of grids but we try to separate them into two major groups. The first one contains approaches that modify the content of each cell of the grid, for instance by adding supplementary information or by using non-probabilistic information (as it is the case with evidential grids that we use). The second group is built up from the methods that extend grids into more dimensions. We also put into this category the approaches modifying the data structure.

A hybrid category may be designated as well for the so called 2.5-dimensional (2.5D) grids, where supplementary dimensional information is added to the grid (Cao, Gu, and Huang 2008). Among many proposed strategies, one should mention the 2.5D grids proposed by Himmelsbach, Müller, et al. (2008). The authors use such grids to create a map out of a Velodyne point cloud. From this representation, they extract bounding boxes around cells that regroup obstacles that exceed a particular threshold of minimal height. 2.5D grids were successfully used for navigation purposes (Hundelshausen et al. 2008; Himmelsbach, Luettel, et al. 2011). As a natural extension of 2D grids, also 3-dimensional (3D) occupancy grids are sometimes used. However, they present important limitations due to their greediness in terms of memory usage.

One of the problems present when handling occupancy grids is their large memory footprint increasing rapidly with higher grid resolutions. In order to minimise this effect, optimised 2D grids, called quadtree grids have been proposed. Quadtrees are multi-resolution grids implemented as a tree data structure. Each cell in a 2D grid is separated into 4 subcells, those being divided as well recursively if needed. The idea of tree-based grids has been generalised into more dimensions and, for instance, octree is the three-dimensional equivalent of the aforementioned quadtree representations. In Octrees, each cell can be divided in 8 (2^3) subcells. The above principle is illustrated in Figure 4.1c. In this manner, uniform zones without need for detailed description are represented by larger cells, whereas the heterogeneous

(a) Point cloud. (b) OctoMap. (c) Illustration of tree-based grids.

Figure 4.1 – Examples of 3D modelling. Source: Wurm et al. (2010).

zones can be modelled at a finer level of detail by split cells. The advantage of this Quadtrees have been successfully used in (Xie et al. 2010) for cascade matching with increasing resolution. An approach based on this representation, called OctoMap, has been used for modelling 3D environments for the development of mapping systems (Wurm et al. 2010). A perfect example of an object that need such modelling are trees, cf. Figure 4.1b.

Besides certain advantages of the tree-based grids, like smaller memory usage, this approach has a few inconveniences as well. Most importantly, combining such structures is more complex than for other approaches. As a result, this solution may turn out to be very demanding in computation time. Tree-based structures, quadtrees and octrees, are well adapted for mapping purposes when the scene is static. Dynamic objects were not taken into consideration and are hard to be modelled in this way. To alleviate this problem, Hähnel et al. (2003) proposed a method for producing dynamic textured 2D and 3D models using probabilistic occupancy grids. Authors of this method use an Expectation-Maximization (EM) algorithm in an incremental manner to estimate which sensor measurements correspond to static obstacles. This approach seems to be inclined towards the detection of static objects and the filtering out of dynamic ones, even if the authors claim that it can be used to isolate dynamic objects.

4.2 Evidential occupancy grids and perception grids

Evidential occupancy grids are not as coveted as their probabilistic ancestors. The principal cause is their increased computational complexity. Secondly, the mathematical formalism of belief functions is less known and a little bit more complicated than the probability theory.

4.2.1 Theoretical background

Evidential occupancy grids differ from their probabilistic counterparts only through the content of cells. In an evidential grid, each cell contains a mass function[1] defined on frame of discernment (fod) $\Omega = \{F,\ O\}$. Here, the mass attributed to F corresponds to the belief that the space is free, whereas the mass of class O refers to this space being occupied. Finally, the information uncertainty is quantified by the mass $\mathrm{m}(\Omega)$.

[1]Generally, one can use any equivalent representation of a mass function, such as belief, plausibility, communality etc., but we will stick with the mass functions.

Perception grids In the following, we will not only use evidential occupancy grids, but we will treat as well perception grids. Under the term perception grid, we will consider evidential grids with any frame of discernment. We consider that frames of discernment in use are exhaustive, accordingly to the closed-world assumption.

An occupancy grid models the world using a tessellated representation of spatial information[1]. In general, it is a multidimensional spatial lattice with cells storing some stochastic information. In a two-dimensional grid, each cell represents a box (a part of environment) $X \times Y$ where $X = [x_-, x_+]$, $Y = [y_-, y_+]$. In case of a perception grid, a cell stores a mass function $\mathrm{m}_G^{\Omega}\{X, Y\}$ with an arbitrary frame of discernment Ω. In this notation, $\mathrm{m}_G^{\Omega,(t)}\{X, Y\}(A)$ is the mass on Ω of element A for the grid G at time t and at position X, Y. Some parts of this notation will be omitted in the following when no risk of confusion exists. The notation describing grid and cell contents used in the following will adhere to the rules as described on page xxi.

4.2.2 Applications

There is some research that takes advantage of evidential grids. Pagac, Nebot, and Durrant-Whyte (1998) presented methods for construction of this type of grids using ultrasonic sensors. Non-negligible uncertainty of data obtained from such sensors found the response in the evidential grids, since the belief functions theory (BFT) has necessary tools to model information uncertainty. In (Aitken 2005), a detailed analysis of evidential occupancy grids obtained by simulations has been presented.

Evidential grids have been also employed for outdoor scene perception. Moras, Cherfaoui, and Bonnifait (2010), Moras, Cherfaoui, and Bonnifait (2011b), and Moras, Cherfaoui, and Bonnifait (2011a) used evidential occupancy grids created from lidar data for detection of mobile obstacles in urban environments. Vision-based grids are more recent, but promising approaches have been demonstrated. Xu et al. (2014) presented for instance an image processing system with multiple classifiers whose results are combined using evidential grids.

4.2.3 Grid reference systems

World-centred grids A world-centred grid is attached to a fixed point of the environment and does not move. The mobile robot evolves in the environment, and thus in the grid, as do other objects in its surroundings. Such a configuration is used often when mapping is to be done, e.g. in Simultaneous Localization and Mapping (SLAM) approaches. In order to illustrate this approach, let us consider two grids `SensorGrid` (SG) and `PerceptionGrid` (PG). The perception grid PG which is world-centred and stores the fusion result, whereas the sensor grid SG represents the instantaneous sensor reading. When the SG is constructed, it is transformed into the coordinates of PG. The PG from the preceding epoch is used in order to predict the current state of the vehicle's environment. Next, both grids are combined together in order to integrate the information contained in the new SG. Finally, a portion of PG around a point of interest, e.g. around the vehicle, can be extracted and re-transformed into vehicle's coordinates for further processing, like navigation. The advantage of this method is that it needs few operations performed on the grids: a single transformation and interpolation are necessary for each SG. Moreover, one can obtain the global map of the environment after processing a single sensor scan. On the other hand, this method possesses several inconveniences. First of all, PG has fixed dimensions

[1] The term "occupancy grid" is often an abuse of language and an informal term, because the grid specifies not only the occupied space but also free space and other information as well.

that limits the zone in which the vehicle can evolve. Obviously, one can alleviate this problem by dynamically recreating the grid around the current position of the mobile actor, but such an approach makes the system more complex. Finally, the majority of known navigation systems use grid in local coordinates, so an additional transformation after each scan is necessary to pass from global reference system to vehicle's coordinates. Such an approach effectively makes the above mentioned advantages disappear.

Ego-centred grids In the case of ego-centred grids, the resulting grid (called PG) is mobile and attached to the mobile actor and generally centred on it. In this case, we will call such a grid a local dynamic map (LDM)[1]. This grid moves with the robot, which requires to reprocess it on each movement of the vehicle. Such a grid has the advantage of always covering the same zone around the vehicle and, more importantly, it does not limit the area where the robot may evolve. In opposition to the world-centred grids, in the ego-centred case, it is the local dynamic map that gets repositioned to the vehicle's coordinate system. Cells that were outside preceding grid limits are considered unknown and attributed corresponding masses. As in the world-centred approach, the grid is used for prediction. An important part of this step is the discounting of cell contents. Next, one can combine two grids, instantaneous SG and preceding PG using vehicle's coordinates. The disadvantage of this method rests in the fact that the update of a grid needs an interpolation of the local dynamic map at each moment. Such processing degrades the information contained in the grids. An imaginable, but unrealistic solution would be to store the history of preceding instantaneous grids SG in order to perform complete fusion of the dynamic grid PG at each moment. This approach is however unusable in practice due to its important computational and memory needs.

4.3 Perception grids for dynamic perception

Evidential grids use the theory of evidence and benefit from its properties like natural representation of the unknown and well-developed theoretical tools. The use of evidential grids allows the fusion of multiple sensors in a straightforward manner. A grid can be constructed for each data source and all grids can be combined together into one `SensorGrid` before further processing as described in the next chapter.

4.3.1 `PerceptionGrid`

One of the evidential grids used in the system is the `PerceptionGrid` (PG). This grid is unquestionably the most important of all evidential grids. It has been introduced to store the results of information fusion. PG is as well the output of the perception system and could be used in further steps of processing in an intelligent vehicle, e.g. for trajectory planning.

The choice of such a fod is determined by the objectives we want to achieve. In our approach, respective classes represent:

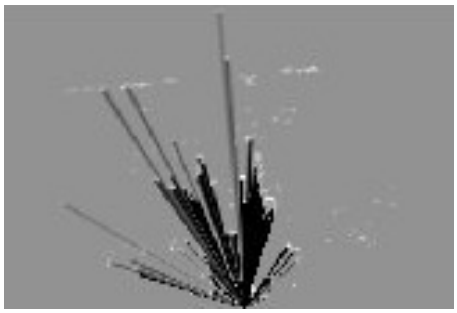
D drivable free space,

N non-drivable free space,

M mobile moving objects,

[1] This name, is general, used for grids containing information about dynamic objects.

(a) Camera view of the scene.



(b) `SourceGrid` (SoG) in Cartesian coordinates.

Figure 4.2 – An example of an occupancy grid obtained using a multi-echo lidar sensor. Grid colour code: white – occupied, black – free, grey – unknown.

S temporarily stopped objects,

I mapped infrastructure like buildings, walls, etc.,

U unmapped infrastructure.

Mass functions of each cell of PG use therefore $\Omega_{PG} = \{D,\ N,\ I,\ M,\ S,\ U\}$ as the frame of discernment (fod). From time to time, we will denote by *movable* the union $\{M,\ S\}$. An important note is to be done about the definition of drivable space D. In the following, we consider drivable the free space, such as road surface, where a vehicle can drive on, i.e. any of its parts (notably wheels) can be situated there. One must be reminded that some research works use the notion of *navigable* free space. Such a free space can be seen as the drivable space reduced in order to take into account the geometric model of the vehicle. In other words, the centre of the vehicle is situated on the navigable space if and only if, after applying any possible rotation, its bounding box (geometric model) is wholly situated on the drivable space. Since the drivable space, in contrast to the navigable space, is independent of the size and the shape of the vehicle, we opted for the use of the former.

Ω_{PG}, besides describing the final result of the fusion process, is also a common, most refined, frame of discernment used in our information fusion system. Indeed, in the Dempster–Shafer theory (DST), two pieces of evidence need to be defined on the same frame of discernment in order to be combined[1]. When the frames of discernment in question differ during data processing, one has to transform them to a common frame. The transition between one fod and another is done by applying a refining function, cf. Definition 24 for details.

As the `PerceptionGrid` (PG) retains the result of information fusion, the need to store previous data disappears. Otherwise, it would be necessary to store all the input grids for a given horizon of time. Such an approach is obviously inefficient and can be envisioned only if the horizon is very limited.

4.3.2 `SourceGrids`

For each exteroceptive sensor, such as a lidar or a camera, an evidential grid called `SourceGrid` (SoG) should be created. The system architecture permits equally the use of a single or multiple sensors. When two or more SoGs are used, they have to be combined into one grid before further processing or alternatively they can be fused at the time of their arrival. The manner in which such grids are combined is described in Chapter 5.

[1]However, there may exist fusion operators that allow combining pieces of evidence defined on distinct frames.

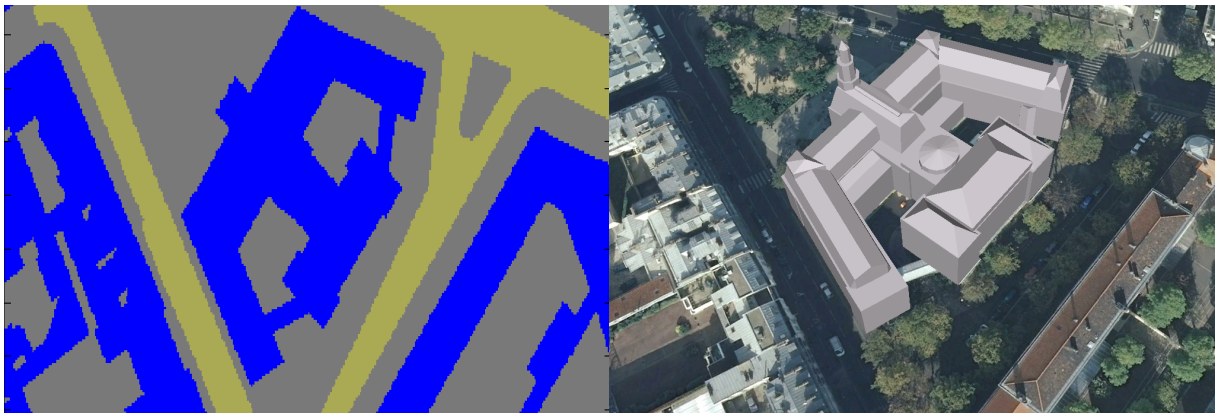


Figure 4.3 – Paris 12th district city-hall. Left: a portion of the constructed `GISGrid` obtained using data from IGN maps (Soheilian, Tournaire, et al. 2013). Right: 3D view of the area (Google 2014a). Colour code: blue — buildings, dark yellow — roads, grey — intermediate space.

A new `SourceGrid` is created for each incoming data acquisition. Each cell of the `SourceGrid` stores a mass function m_{S_i} defined on the frame of discernment Ω_{S_i}. The frame can vary depending on the sensor in use. The higher the expressiveness of the sensing device, the more classes the corresponding grid will represent. Typically for a lidar, $\Omega_{S_i} = \{F,\ O\}$, where F refers to the free space and O to the occupied space. The basic belief assignment depends on the model of the actual sensor. Details about the sensor model used in this dissertation are given further in this chapter in Section 4.4 onwards. Another sensor, for example camera, could be much more informative and comprehend detailed classes such as road surface, pavement, building, grass etc. An example of a simple occupancy grid is illustrated in Figure 4.2. Other evidential grids based on a vision system are described in (Xu et al. 2014).

The frame of discernment Ω_{S_i} is distinct from Ω_{PG} and a common frame for all sources has to be found. Hence, a refining r_{S_i} is defined as stated in Equation 4.1.

$$
\begin{aligned}
r_{S_i} : 2^{\Omega_{S_i}} &\rightarrow 2^{\Omega_{PG}} \\
\{F\} &\mapsto \{D,\ N\} \\
\{O\} &\mapsto \{I,\ U,\ S,\ M\} \\
A &\mapsto \bigcup_{\theta \in A} r_{S_i}(\{\theta\}) \qquad\qquad \forall A \subseteq \Omega_{S_i} \text{ and } A \notin \{\{F\},\ \{O\}\}
\end{aligned}
\tag{4.1}
$$

Refining r_{S_i} makes it possible to perform the fusion of $\texttt{SourceGrid}_i$ containing instantaneous grid obtained from sensor i with other grids. Equation 4.2 expresses the refined mass function.

$$
\mathrm{m}_{S_i}^{\Omega_{PG}}(r_{S_i}(A)) = \mathrm{m}_{S_i}^{\Omega_{S_i}}(A) \qquad\qquad \forall A \subseteq \Omega_{S_i} \tag{4.2}
$$

4.3.3 `GISGrid`

The purpose of the `GISGrid` (`GG`) is to contain all the data exploited from maps. In our approach, we limited the use of this data to geometrical information about the surface of the road and buildings. This grid allows us to perform contextual information fusion incorporating the meta-knowledge about the environment. Again, the meta-knowledge is related to the geometrical information furnished by maps. We separated three different contexts for which the meta-information differs. Figure 4.3 juxtaposes a sample of `GISGrid` and a three-dimensional view on a building model.

Urban scene contexts

These contexts correspond to the classes of the frame of discernment used by the `GISGrid`. Namely, `GG` uses the fod $\Omega_{GG} = \{B,\ R,\ T\}$. Class B corresponds to the area occupied by buildings. Analogously, R defines the road surface. Finally, the class T models intermediate space that is not contained in either of the above. For example, the intermediate space contains pavements.

Each context has its proper characteristics. In the building context, the only classes we are supposed to detect are infrastructure I and non-drivable free space N. This last case is possible only if the map is faulty and depicts a non-existing building.

The road context is much more complicated and may contain any class except for mapped infrastructure I and non-drivable space N. Indeed, one usually finds moving obstacles like cars or motorbikes on the road, but one cannot exclude the presence of pedestrians, especially on zebra crossings. Moreover, stopped vehicles are often present on (the side of) the road. What concerns the infrastructure, one should allow the existence of small urban furniture (class U) such as lamps or barriers. Finally, an important assumption is made about the drivability of the road surface, supposing that the road is by definition drivable and thus excluding the non-drivable class D.

The last context, the intermediate space T should be understood as non-building and non-road environment. Such a vague definition corresponds exactly to the knowledge possessed about this part of vehicle's environment. In this context, mobile M and stationary S objects as well as small urban infrastructure U can be present. Obviously, one should disallow the vehicle to drive on the intermediate space unless in the case of emergency[1].

The `GISGrid` (`GG`) is created, for instance, by projecting map data onto a two-dimensional world-referenced grid. This is the step where the meta-information from maps is included. As stated above, this meta-knowledge can ban the existence of mobile objects where buildings are present and, conversely, it indicates the possibility to find these objects on roads. The exact construction method of the `GISGrid` depends however on available geodata.

The fod Ω_{GG} is different from the common frame Ω_{PG}. Some rules in the theory of evidence, such as Dempster's rule, do not allow the direct combination of BBAs expressed on different frames of discernment, as this in the case with the `SourceGrid`. It is then necessary to express every belief assignment on a common frame of discernment before the combination. In our work, the mapping r_{GG} is used when needed:

$$
\begin{aligned}
r_{GG}\colon 2^{\Omega_{GG}} &\to 2^{\Omega_{PG}} \\
\{B\} &\mapsto \{I\} \\
\{R\} &\mapsto \{D,\ S,\ M\} \\
\{T\} &\mapsto \{N,\ U,\ S,\ M\} \\
A &\mapsto \bigcup_{\theta \in A} r_{GG}(\{\theta\}) \qquad \forall A \subseteq \Omega_{GG} \text{ and } A \notin \{\{B\},\ \{R\},\ \{T\}\}
\end{aligned}
\tag{4.3}
$$

Using this refining, one can compute the mass transfer as follows:

$$
\mathrm{m}_{GG}^{\Omega_{PG}}(r_{GG}(A)) = \mathrm{m}_{GG}^{\Omega_{GG}}(A) \qquad \forall A \subseteq \Omega_{GG} \tag{4.4}
$$

[1] Please see perspectives for details.

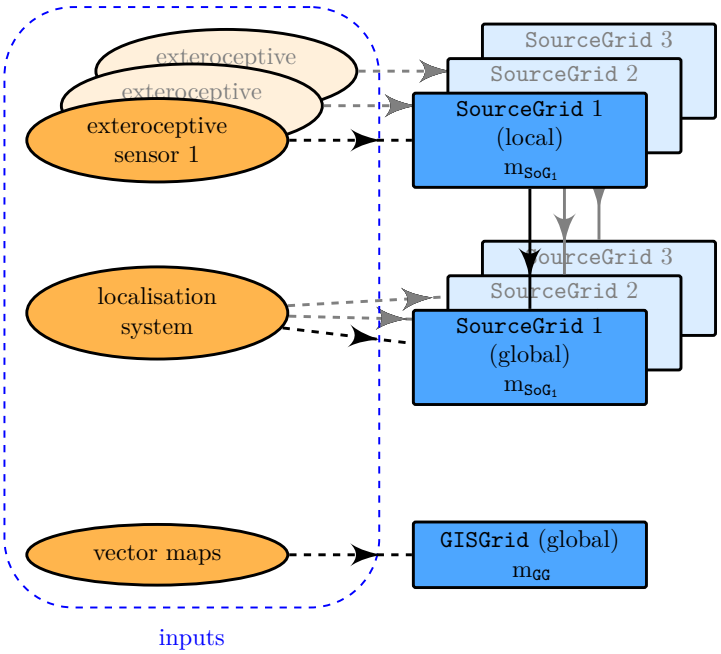


Figure 4.4 – Part of our perception system where the sensor models are applied.

The mapping r_{GG} indicates that, for instance, building information B fosters mass transfer to class I. On the road surface R, the existence of drivable free space D as well as stopped S and moving M objects is possible. Lastly, on the intermediate area T, the existence of mapped infrastructure I can be excluded. Similarly, the free space is non-drivable therefore class D is disallowed as well. The presence of all other classes is however allowed.

4.4 Sensor models

Exteroceptive sensors deliver us information about the surrounding environment. The received information is already processed, as the sensing devices "translate" physical properties and quantities into exploitable data. In this dissertation, we are interested in describing the vehicle scene: positions of surrounding objects, their class or generally the occupancy of the space. In order to be able to describe this, it is necessary to transform raw sensor data into valuable information that can be used in further processing. In our case, the information should be represented as an evidential occupancy grid. For this purpose, we will define functions called *sensor models*. A sensor model can be determined both in an empirical manner using statistical characteristics of the sensing device or thanks to a physical model. Figure 4.4 shows the part of our perception system where the sensor models are applied to transform sensor data into evidential perception grids.

One can distinguish two types of sensor models: direct and inverse. A direct model predicts the measure given the state of the observed system. For instance, given a laser sensing device, this model answers the question: what is the likelihood of having a lidar echo at distance $d' \in [0, +\infty]$ if there is an obstacle at distance d from the sensor.

An inverse model, *inversely*, predicts the state of the system given the sensor measure. Such a model tells us, for example, the confidence in the fact that there is an obstacle at distance $d \in [0, +\infty]$ if the sensor (a lidar) returned an echo at distance d'. When occupancy grids are used to represent the environment, a sensor model defines the relation between the state of grid cells and the sensor data.

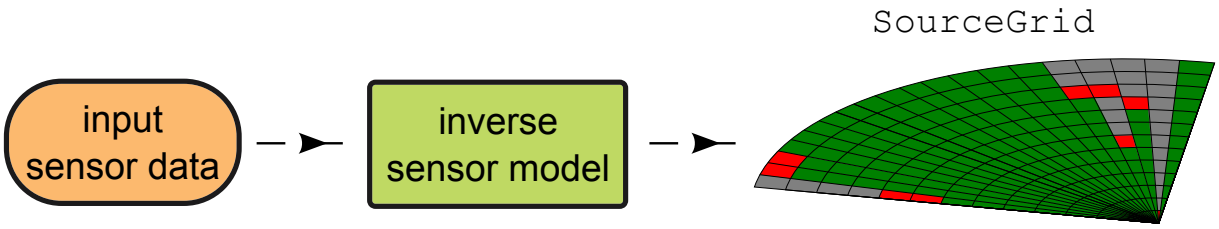


Figure 4.5 – An example of construction of a polar `SourceGrid` (SoG) from sensor data using an inverse sensor model.

In the following, we will define inverse sensor models in order to make connection between the data delivered by a sensor and occupation of the grid cells, as it is illustrated in Figure 4.5 to create an input occupancy grid called `SourceGrid` (SoG) described later. Please take in mind that in sequel, the term sensor model will denote an inverse sensor model unless explicitly stated otherwise.

4.5 Sensor model for a lidar

In this section, we present an inverse sensor model for a lidar that we developed. We propose here various sensor models for lidar. The first model will correspond to the actual implementation on our test-bed platform. In comparison to the other models, this one simplifies the processing of impacts that occur in the same cell. It is done by considering in the same way the case where there is only one laser impact in a given grid cell and the case where there are two or more impacts. In the latter situation, the next model would enhance our certainty about an obstacle being present in the corresponding cell. Subsequently, we will therefore demonstrate possible improvements and describe a discrete model that manages the imprecision of the processed information implicitly. Namely, the size of a grid cell is chosen so that it is greater than the mean imprecision of the sensing device. Finally, the third and the last model is a continuous imprecise model that takes into account both the sensor uncertainty and data imprecision. However theoretically appealing, the complexity of implementation of this model and its computational inefficacy made us consider its simplified versions.

The input of a sensor model is the lidar scan that can be denoted by a point cloud (set of points) $\mathcal{P}$ defined by Equation 4.5. Each point p_i is described using 3D Cartesian coordinates.

$$\mathcal{P} = \left\{ p_i = \begin{bmatrix} r_i \\ \theta_i \\ \phi_i \end{bmatrix}, i \in [0, n_P] \right\} \tag{4.5}$$

The x, y, z coordinates correspond respectively to the left-right (positive-negative), front-back and up-down axes. The assumption we make about the point cloud is that it does not contain points on the ground or below it, i.e. $\forall p_i \in \mathcal{P} : z_i > \epsilon$[1]. Usually, a pre-processing step is needed to satisfy this condition.

4.5.1 LIght Detection And Ranging (LIDAR)

Our principle sensor is a lidar, a ranging device analogous to radars and sonars, but employing laser-beams for detection. All these sensors measure the distances between the device and obstacles using

[1] ϵ denotes a small value that makes for any bumps on the road.

the physical properties of radio wave (radars), sound wave (sonars) or light (lidars) propagation.

Lidars became popular quite recently and are currently used in various domains, ranging from topography and cartography, as well as other mapping systems to studies of atmospheric and weather changes. Laser ranging devices are also widely employed in order to create Digital Terrain Model (DTM) discussed in Section 2.1. More recently, lidars gained popularity in the field of robotics for the purpose of environment perception.

The operating principle of a lidar is the following, please refer to Figure 4.6. The device emits a laser beam towards a particular direction and measures the time up to the arrival of the echo. Depending on the application, one can extract from this echo quite different information about the detected obstacle. In our case, the position of the obstacle is of interest. The so called time-of-flight principle is used to estimate the distance to the object[1]. Given that the laser beam flies at the speed of light c, the time of beam emission is t_e and the time of arrival is t_a, let $\Delta t = t_a - t_e$. One can compute the distance d to the obstacle using the formula $d = \frac{c \cdot \Delta t}{2}$. The laser beam propagates along the line that corresponds to its optical axe, this line being parametrised generally by its orientations, angles θ and ψ, which are supposed to be known precisely. Then, taking account the distance computed above and the direction of the laser beam, one obtains 3D spherical coordinates d, θ, ψ of the impact point. In the reality, this physical phenomenon is more complex, since the generated laser beam is imperfect and diverges. As a result, the light beam on detected objects are of non-negligible surface and possibly hit obstacles further away giving origin to multiple received echos for a single emitted beam. Moreover, the properties of the incident surface may distort the measure significantly, as e.g. the reflectivity changes the energy of the received beam. All in all, these effects may introduce uncertainty about the existence and the position of the obstacle.

Hypotheses Before describing the sensors models used to translate the data from lidar into an occupancy grid, let us list the hypotheses that we adopted in order to build these models. First of all, a lidar sensor, being possibly installed on the front of the vehicle, as in our configuration, is prone to provide impact points that correspond to the ground. As further processing supposes that no impact points hit the ground, there is a preprocessing phase that filters such points out. For the purpose of this preprocessing, we suppose that the ground is locally flat around the vehicle and it can be modelled as a plane. Another difficulty with laser-based sensors is that transparent obstacles cannot be reliably detected, because the light beam traverses such objects with almost no reflection. Therefore, the assumption we have made stipulates that all obstacles reflect the laser beam.

Without trying to estimate dynamic parameters of the surrounding environment, a correct prediction of the scene cannot be realised. It is therefore imperative to make an assumption of static world. However, if we consider that the scan frequency of the sensor is high with respect to the scene dynamics, then the prediction error is negligible, because the environment barely changes during the short time lapse between consecutive scans. If a part of environment is not perceived, this hypothesis does not hold any more. We take this phenomenon into account by applying a discount (decay) factor not only to the unobserved parts (grid cells) but also to the observed ones. The observed cells need indeed to be discounted since the environment is dynamic and the previous grid is used on the arrival of the current scan to predict the state of the environment at this moment.

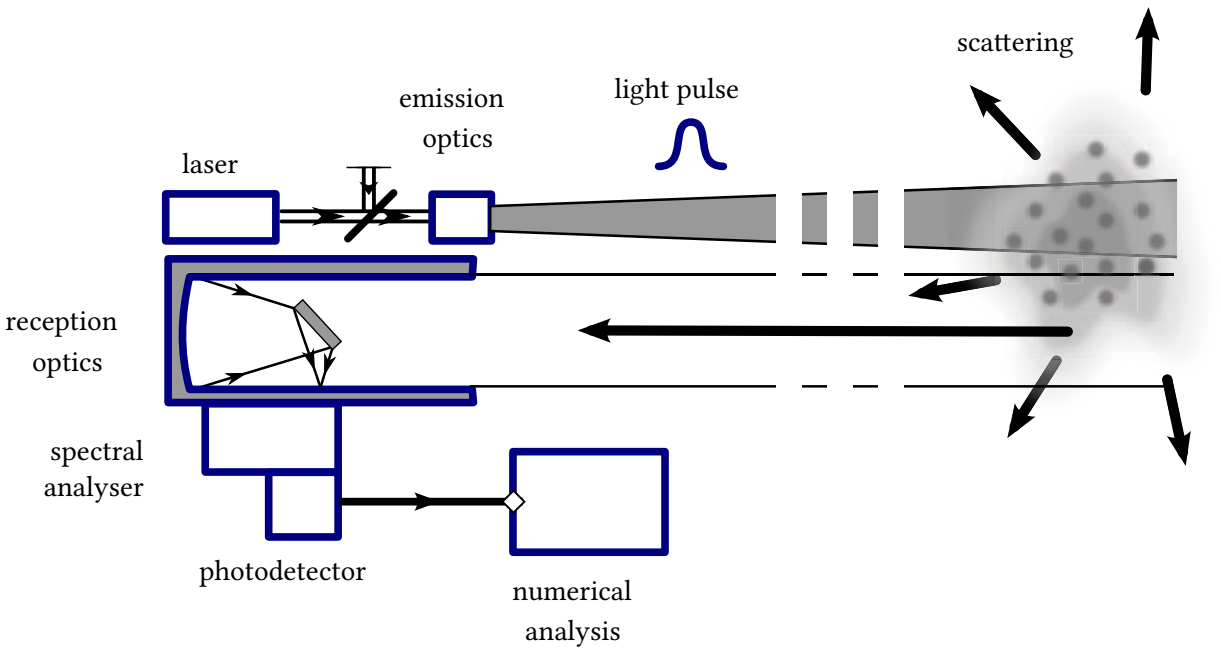


Figure 4.6 – Operating principle of a lidar.

4.5.2 Building a lidar sensor model

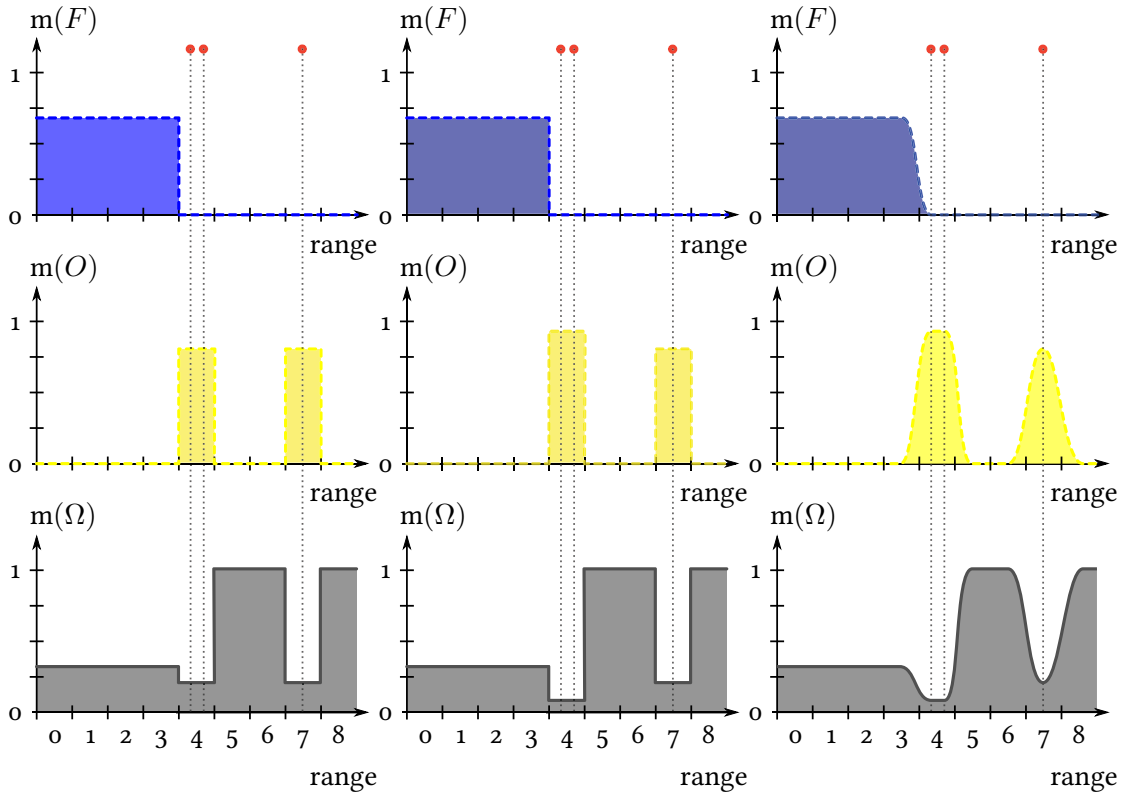
In order to create a lidar model, our approach is to treat the data along a single laser beam. We consider each angular sector independently and define a model for a 1-dimensional (1D) lidar, where the only dimension of interest is the distance from the sensor. Such one-dimensional sectors are then regrouped to produce a polar evidential occupancy grid, as presented in Figure 4.5A polar grid closely reflects the underlying sensor yet it is rather impractical to use. A necessary transform is used to pass from polar reference frame into Cartesian one. As such a processing allows that a single cell in a polar grid maps to possibly multiple cells in a Cartesian grid (or vice versa), a bilinear method for interpolation is applied where necessary. We omit the details of such a transformation here and refer the reader to (Moras 2013).

When defining a sensor model, we proceed in the following way basing our reasoning on the operation principle of a lidar as shows the schema from Figure 4.6. A single laser beam can result in multiple impacts, giving several detections in a single angular sector. It is assumed that the space before the closest obstacle detected by lidar is free. The space around the impact point is occupied. Space between impact points cannot be attributed as free, as there might be obstacles occulted by objects situated closer to the lidar.

For a particular sensor, we need to know statistical values that can be determined empirically. The first of them, the true positives rate μ_O equals to $1 -$ false alarm rate. The false alarm rate describes how often the sensor indicates a detection of a non-existent obstacle. The second quantity, detection rate μ_F, tells us what is the frequency of miss-detection, i.e. how often the sensor detects no obstacle where there is one. A recurring example of such elements are windows and other objects made of glass. There are however other types of objects that rest undetected. The above defined parameters serve us to model the uncertainty of the information provided by the sensor.

The importance of the construction of sensor model is underlined by many authors. Recent works more and more often work in three dimensional spaces in order to take into account the height of

[1]However, some models use a technique based on the phase difference of emitted and received signals.



(a) Multi-echo. (b) Reinforced multi-echo. (c) Imprecise multi-echo.

Figure 4.7 – Lidar inverse sensor models. Example of one angular segment of a lidar acquisition. Red dots represent laser impacts, diagrams show mass attribution.

detected obstacles and the incurred occlusions. An original approach proposed in (C. Yu, Cherfaoui, and Bonnifait 2014) which makes use of the provided sensor data in a two-fold manner. On the one hand, point cloud impacts are used to ascertain the existence of obstacles as in classic methods. The novelty lies in the fact that the proposed algorithm, using the height above the ground of each impact, tries to deduce the certainty of space being free. This can be done for the cells which are crossed by the laser beam. That being told, the use of this method is limited to high-quality 3D Velodyne lidars. We present in the following lidar models that are two-dimensional and cannot take advantage of such approaches.

4.5.2.1 Multi-echo model

The models mentioned further in this chapter reinforce the certainty of a detected obstacle if the lidar provides more impact points in a given cell. A simplification of this approach leads us to a model without reinforcement of the certainty. This assumption can be made safely, because of the small grid size and relatively low spatial resolution of a lidar scan, i.e. a sparse point cloud. This model can be used when the situation where two or more impacts are situated in a single cell of the grid lattice occurs rarely. With lidars having multiple layers, the use of such model can present some impediments, since additional information given by supplementary impacts from other layers will effectively not be taken into account. The simplified algorithm is presented in the listing of Algorithm 1. Furthermore, an illustration of its application is shown in Figure 4.7a.

Algorithm 1 Discrete multi-echo lidar sensor model

Require: point cloud $P = \{p\colon \text{lidar point}\}$
Ensure: Evidential grid SG models the point cloud data.
{Initialisation.}
for all angle $\theta \in [\theta_{min},\, \theta_{max}]$ **do**
MinimalRadius$[p.\theta] \leftarrow \infty$
end for
{Find minimal radius for each angle.}
for all $p \in P$ **do**
$r \leftarrow p.r$
$\theta \leftarrow p.\theta$
if $r <$ MinimalRadius$[\theta]$ **then**
MinimalRadius$[\theta] \leftarrow r$
end if
$\mathtt{SG}[r, \theta](\Omega) \leftarrow \mathtt{SG}[r, \theta](\Omega) \cdot (1 - \mu_O)$
$\mathtt{SG}[r, \theta](O) \leftarrow 1 - \mathtt{SG}[r, \theta](\Omega)$ {This step can be performed later for optimisation. The advantage: it is executed only once.}
{Other masses do not change.}
end for
{Affect masses.}
for all angle $\theta \in [\theta_{min},\, \theta_{max}]$ **do**
for all $r \in [0,$ MinimalRadius$[\theta])$ **do**
$\mathtt{SG}[r, \theta](\Omega) \leftarrow (1 - \mu_F)$
$\mathtt{SG}[r, \theta](F) \leftarrow 1 - \mathtt{SG}[r, \theta](\Omega)$
{The mass attributed to class O can be affected here for better performance (cf. previous comments).}
end for
end for
return SG

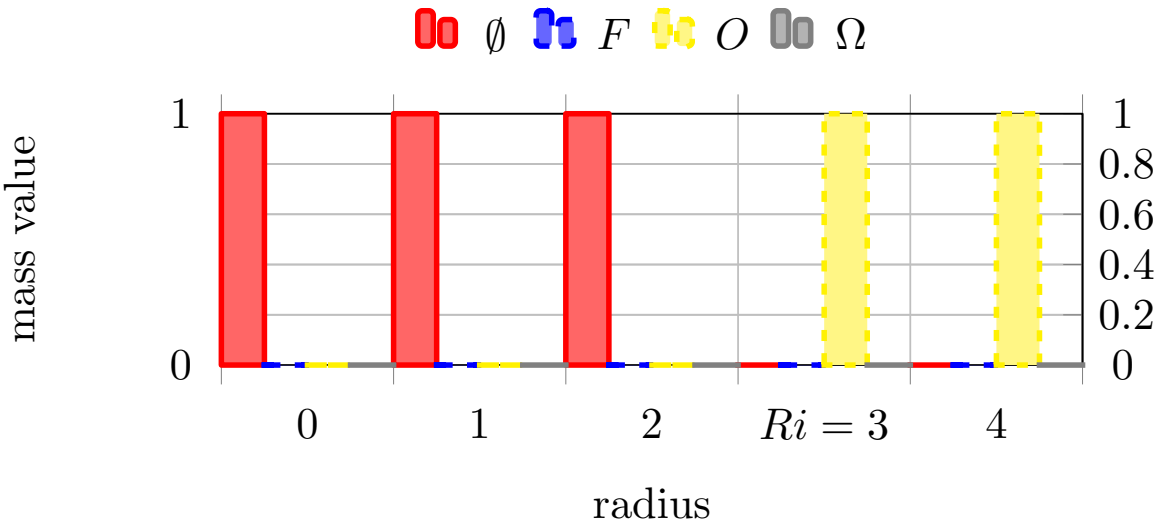


Figure 4.8 – Lidar sensor model — mask function for a single scan point.

4.5.2.2 Reinforced multi-echo model

In order to take into account multiple impacts in one cell, we propose a reinforced multi-echo model. It has another advantage of closely adhering to the reality while still having low algorithmic complexity. This fact allows a real-time implementation.

In this reinforced model, for each point in the sensor data, two basic belief assignments (bbas) are defined: impact and mask mass functions that both store information about the position of the scan point. The reinforcement comes from the fact that each additional impact in a cell will provide more certainty about an obstacle being present in the corresponding part of the vehicle's environment. This effect can be easily noticed on Figure 4.7b. Comparing to Figure 4.7a, one remarks that the cell in which two impacts are present gains in certainty. This fact is expressed by higher mass attributed to the occupied state O.

Mask function The mask function for an impact point at distance R_i is defined using the following equation:

$$\mathrm{m}_{R_i}^{\text{mask}}\{r,\,\theta\}(\emptyset) = \begin{cases} 1 & r < R_i \\ 0 & r \geq R_i \end{cases} \tag{4.6}$$

$$\mathrm{m}_{R_i}^{\text{mask}}\{r,\,\theta\}(F) = 0 \tag{4.7}$$

$$\mathrm{m}_{R_i}^{\text{mask}}\{r,\,\theta\}(O) = \begin{cases} 0 & r < R_i \\ 1 & r \geq R_i \end{cases} \tag{4.8}$$

$$\mathrm{m}_{R_i}^{\text{mask}}\{r,\,\theta\}(\Omega) = 0 \tag{4.9}$$

The interpretation of the mask is the following. It models the confidence we possess and that we deduce from a single laser impact point. In this manner, there is a possibility to find any class *ahead* of the impact (which is expressed by the attribution of mass to $\emptyset$.). Conversely, the mass allows only occupied or unknown space *behind* the impact point. These assumptions are coherent with the fact that an impact gives confidence about the free space before the impact, since the laser beam traversed the

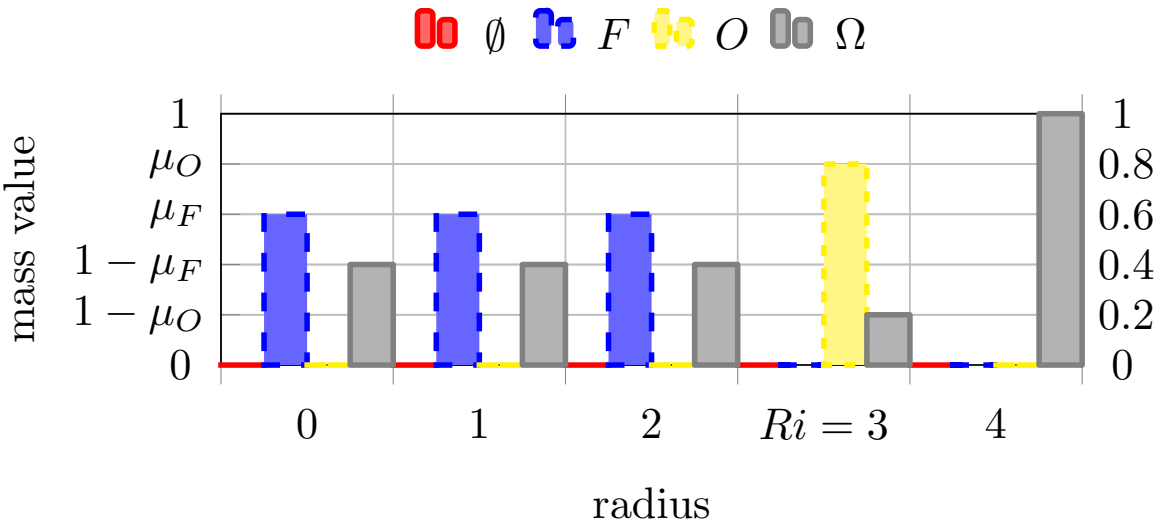


Figure 4.9 – Sensor model — impact function for a single scan point.

space between the sensor and this point. On the other hand, the existence of an impact point does not altogether give reliable information about the free space on the impact place or *behind* it. An example of a mask mass function is depicted in Figure 4.8.

Impact function In turn, the *impact function* stores the information about the position of the impact and the space occupied by the corresponding obstacle. An example of an impact mass function is presented in Figure 4.9. The impact mass function for an impact point at distance R_i and at angle θ is defined as follows:

$$\mathrm{m}_{R_i}\{r,\,\theta\}(\emptyset) = 0 \tag{4.10}$$

$$\mathrm{m}_{R_i}\{r,\,\theta\}(F) = \begin{cases} \mu_F & r < R_i \\ 0 & r \geq R_i \end{cases} \tag{4.11}$$

$$\mathrm{m}_{R_i}\{r,\,\theta\}(O) = \begin{cases} 0 & r < R_i \\ \mu_O & r = R_i \\ 0 & r > R_i \end{cases} \tag{4.12}$$

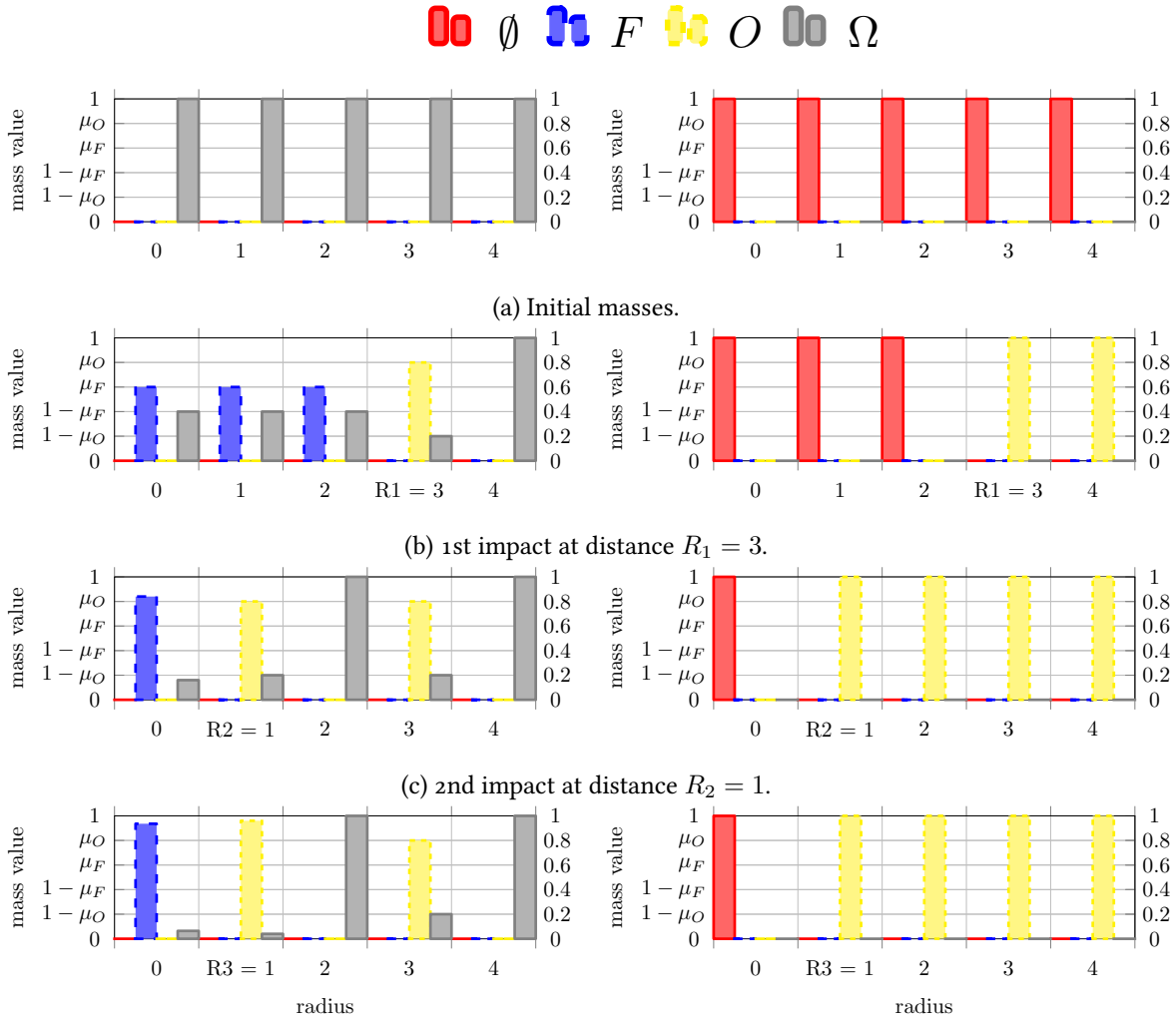
$$\mathrm{m}_{R_i}\{r,\,\theta\}(\Omega) = \begin{cases} 1-\mu_F & r < R_i \\ 1-\mu_O & r = R_i \\ 1 & r > R_i \end{cases} \tag{4.13}$$

The mass functions are initialised as follows. The mask is initially a degenerate mass function with total conflict, whereas the cumulative impact mass models complete ignorance, thus it is a vacuous mass function:

$$\mathrm{m}_0^{\text{mask}} = \mathrm{m}^{\emptyset} \qquad \text{degenerate mass function} \tag{4.14}$$

$$\mathrm{m}_0^{\text{cumulative}} = \mathrm{m}^{\Omega} \qquad \text{vacuous mass function} \tag{4.15}$$

With each impact $R_i \in \mathcal{R}, 0 \leq i < N$ mass functions m_{R_i} and $\mathrm{m}_{R_i}^{\text{mask}}$ are created (see Equations 4.9



(a) Initial masses.

(b) 1st impact at distance $R_1 = 3$.

(c) 2nd impact at distance $R_2 = 1$.

(d) 3rd impact at distance $R_3 = 1 = R_2$.

Figure 4.10 – Lidar sensor model — mass evolution. Left: cumulated impact mass. Right: mask mass.

and 4.13). They are then combined using the following equations using an on-line[1] algorithm:

$$\mathrm{m}_i^{\text{mask}} = \mathrm{m}_{i-1}^{\text{mask}} \textcircled{\cup} \mathrm{m}_{R_i}^{\text{mask}} \tag{4.16}$$

$$\mathrm{m}_i^{\text{cumulative}} = \mathrm{m}_{i-1}^{\text{cumulative}} \textcircled{\cap} \left(\mathrm{m}_i^{\text{mask}} \textcircled{\cup} \mathrm{m}_{R_i} \right) \tag{4.17}$$

Figure 4.10 demonstrates how the subsequent points are combined with the current result.

When all the points of a lidar scan are given at once, the preceding algorithm can be simplified and optimised to take this fact into account. The resulting off-line[2] algorithm is described by Equation 4.19. Mass functions included in this computation are defined in the same way as previously outlined by Equations 4.9 and 4.13.

$$\mathrm{m}_N^{\text{mask}} = \bigcircledcup_{0 \leq i < N} \mathrm{m}_{R_i}^{\text{mask}} \tag{4.18}$$

$$\mathrm{m}_N^{\text{cumulative}} = \left(\bigcircledcap_{0 \leq i < N} \left(\mathrm{m}_{R_i} \textcircled{\cup} \mathrm{m}_N^{\text{mask}} \right) \right) \tag{4.19}$$

Due to specific form of the mask mass function and the impact mass function, the conjunctive combination would induce *no conflict*, i.e. $\mathrm{m}(\emptyset) = 0$ for all cells.

4.5.2.3 Reinforced imprecise multi-echo model

Ideally, a lidar sensor model should take into account both the uncertainty of the device and the imprecision of provided measures. In the proposed model, the measurement imprecision is indicated by the variance of sensor measures $\sigma_{\text{measurement}}$ and is taken into account. The assumption of Gaussian noise model in lidar data is made.

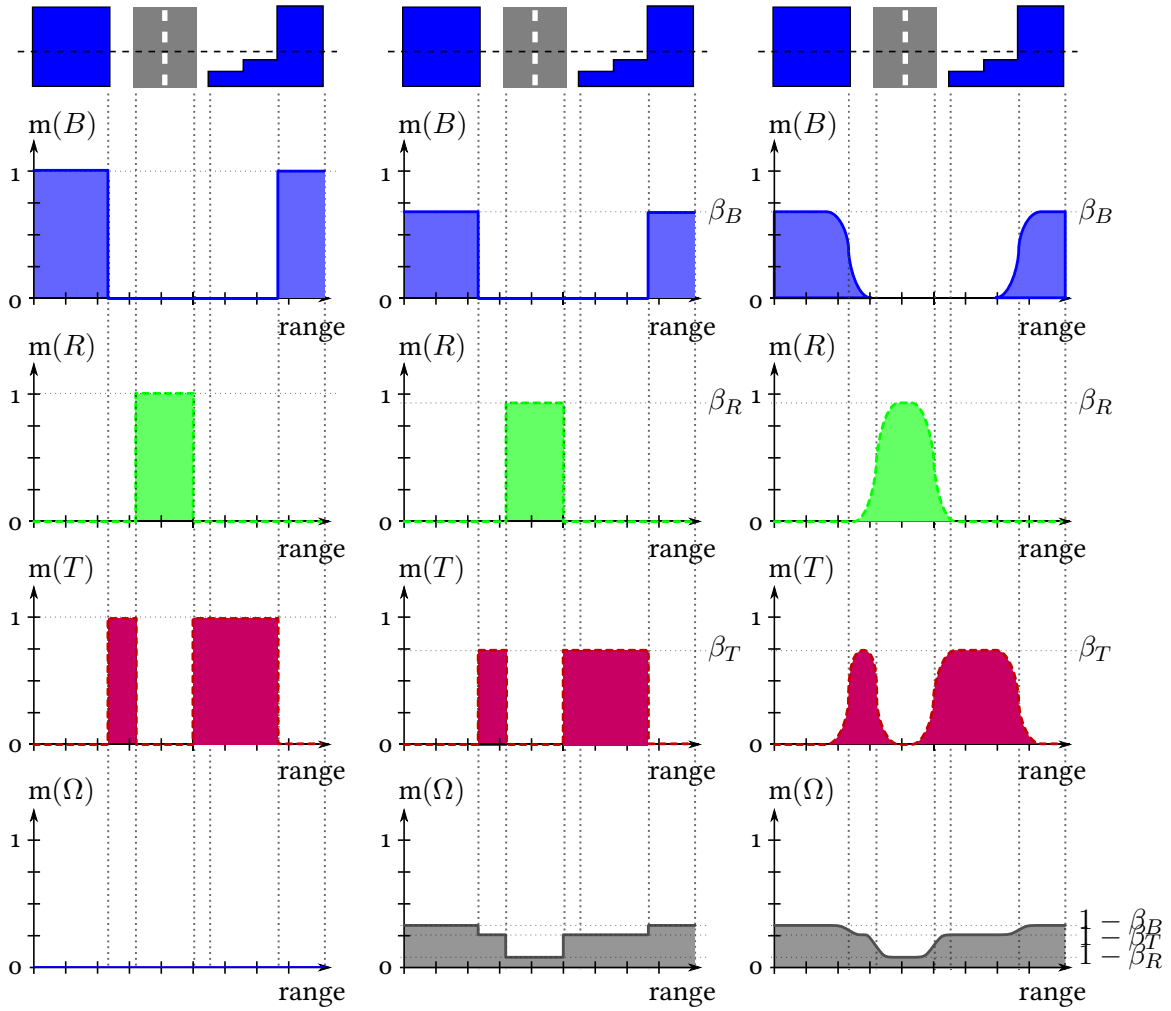
The output grid is created as a fusion of individual mass functions defined for each impact point. Such masses are then combined incrementally to obtain the final grid. Figure 4.7c presents masses attributed to occupied space O, free space F and to the ignorance when using an idealised continuous model.

In further stages of data fusion, the first multi-echo model (cf. Section 4.5.2.1) will be used. This simplified model implicitly manages the imprecision inherent to the sensor data. Moreover, the fact that the reinforcement of certainty about the presence of objects is omitted does not hinder this approach. One can even argue that such behaviour is desired as the sensor model cannot overestimate the chances of finding an obstacle and thus is a safer solution.

4.6 Virtual sensor model for maps

In our approach, we decided to use map data as if it were an additional source of information on par with a lidar or a camera. In other words, a map is a virtual sensor of the same importance as other sensing devices installed on the board of an intelligent vehicle. In this way, we can create different sensor models that translate the geodata from the map into a grid-based representation that can be combined with other sensor data. More importantly, modelling the digital map as an additional sensor allows to account for errors and uncertainties of the map database (Boucher and Noyer 2011).

[1] An on-line algorithm takes each point separately and updates the result as opposed to batch, or off-line, counterparts.
[2] Off-line processing is used here in the sense of batch processing.



(a) Certain. (b) Uncertain. (c) Imprecise uncertain.

Figure 4.11 – Map virtual sensor models. Example of a linear cut over map data. Blue polygons above represent buildings, gray ones represent road surface, diagrams show mass attribution.

A careful reader might ask why the map data should be used as a sensor and not used in a "classical" way, i.e. for conditioning. The answer is that conditioning supposes that the truth lies in the given evidence, i.e. in the map in our case. Yet the maps are known to be not always up-to-date or they can be simply flawed. The most common problems are misplaced elements, unmapped entities or "ghost" objects, that is such that are present in the map but do not exist in reality. Choosing the map to be handled as a sensor permits to perform information fusion (as opposed to conditioning) and to benefit from what it goes with — videlicet, management of uncertainty. That is crucial, because it is unreasonable to trust blindly in map data. This becomes even more important, when one uses maps that do not have any guarantee of completeness or exactitude, which is our case when using OpenStreetMap (OSM) data.

4.6.1 Using map data

The map data may be represented by two sets of polygons $\mathcal{B}$ and $\mathcal{R}$, see Equations 4.20 and 4.21. Each polygon b_i is described by m_i vertices in 2D Cartesian coordinates. x coordinate denotes longitude and y coordinate indicates latitude of a vertex. A polygon is composed of segments (x_1, y_1)–(x_2, y_2), (x_2, y_2)–(x_3, y_3), $\ldots$, $(x_{m_{i-1}}, y_{m_{i-1}})$–$(x_{m_i}, y_{m_i})$, (x_{m_i}, y_{m_i})–(x_{m_1}, y_{m_1}).

- Buildings:

$$\mathcal{B} = \left\{ b_i = \begin{bmatrix} x_1 x_2 \ldots x_{m_i} \\ y_1 y_2 \ldots y_{m_i} \end{bmatrix}, i \in [0, n_B] \right\} \tag{4.20}$$

- Road surface:

$$\mathcal{R} = \left\{ r_i = \begin{bmatrix} x_1 x_2 \ldots x_{m_i} \\ y_1 y_2 \ldots y_{m_i} \end{bmatrix}, i \in [0, n_R] \right\} \tag{4.21}$$

One of our obvious assumptions is that the polygons constituting the buildings and the road surface satisfy the condition:

$$\mathcal{B} \cap \mathcal{R} = \emptyset$$

Without supposing such a statement, one would admit that a building can stand in the middle of the road for example, which is absurd.

4.6.2 Certain map model

A certain (and precise) map model is conceptually equivalent to information conditioning. As a matter of fact, such an approach models the map as the omniscient and unmistakable source of information. Roads, buildings and intermediate classes are disjoint and do not overlap. Moreover, the attributed masses are always categorical. There is no doubt, so the unknown mass $\mathrm{m}(\Omega)$ is always equal to zero. Please see Figure 4.11a for an example of a certain map model.

4.6.3 Uncertain map model

As in the certain model, the roads, buildings and intermediate classes rest disjoint. The difference is that the attributed masses are no more categorical and for each class, a factor $\beta < 1$ defines the map's confidence about the given part of the environment. In this model, the imprecision and inaccuracy are not considered explicitly, as can be seen in the example in Figure 4.11b.

As stated above, the level of confidence β is defined for each map source and is possibly different for each context[1]. In order to define this map model, let us specify the centres of each cell as $\tilde{x} = \frac{x_- + x_+}{2}$ and $\tilde{y} = \frac{y_- + y_+}{2}$ in respective dimensions. With this notation, our model takes the following form:

$$\mathrm{m}_{GG}\{X, Y\}(B) = \begin{cases} \beta_B & \text{if } (\tilde{x}, \tilde{y}) \in b_i \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in [0, n_B] \tag{4.22}$$

$$\mathrm{m}_{GG}\{X, Y\}(R) = \begin{cases} \beta_R & \text{if } (\tilde{x}, \tilde{y}) \in r_i \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in [0, n_R] \tag{4.23}$$

$$\mathrm{m}_{GG}\{X, Y\}(T) = \begin{cases} 0 & \text{if } (\tilde{x}, \tilde{y}) \in b_i \text{ or } (\tilde{x}, \tilde{y}) \in r_j \\ \beta_T & \text{otherwise} \end{cases} \quad \forall i \in [0, n_B], \forall j \in [0, n_R] \tag{4.24}$$

$$\mathrm{m}_{GG}\{X, Y\}(\Omega) = \begin{cases} 1 - \beta_B & \text{if } (\tilde{x}, \tilde{y}) \in b_i \\ 1 - \beta_R & \text{if } (\tilde{x}, \tilde{y}) \in r_i \\ 1 - \beta_T & \text{otherwise} \end{cases} \quad \forall i \in [0, n_B], \forall j \in [0, n_R] \tag{4.25}$$

4.6.4 Uncertain and imprecise model

Uncertain and imprecise map model combines the above described two models in order to take into account both types of data imperfections that can possibly occur. When building this model, we assumed that the limits of the zones, i.e. polygons delimiting buildings or roads, may be wrongly modelled. The positions of polygon vertices have been thus described using two-dimensional Gaussian distribution. Due to this fact, the classes might overlap in this model, as can be seen in Figure 4.11c. The hypothesis of Gaussian distribution of vertex positions was used since at least a few approaches of map-building methods consider Gaussian distributions to describe the noise model in the data (Hammoudi 2011; Burochin 2012). Besides, no other piece of information is at hand to make different hypothesis.

Such a model, taking into account both the imprecision and the uncertainty of available data, is in a close concordance with the reality. Being advantageous at this point, it is nevertheless difficult to implement in practice and, what is more important, can be replaced by its simplified counterparts. Indeed, we have chosen to use the uncertain but precise map model for further processing. In this model, the imprecisions of map data are also taken into consideration but implicitly through the choice of the size of cells with respect to map imprecision.

4.7 Conclusion

Being able to "translate" the sensor data into a grid is the basis and the crucial part of the perception system. It is important that how this building blocks would be combined together in order to infer

[1] In our case, however, $\beta_B = \beta_R = \beta_T = \beta$ since all map data come from the same source and are supposed to be equally correct.

the relevant information about the vehicle surroundings. The fusion process allows us to obtain more information that any single source would ever be able to give.

This is however impossible if the components do not fit together. We solved this problem by using a homogeneous representation, evidential grids, that is able to model extremely different sensor data. Heterogeneous sensors: lidars, cameras, radars, etc. may be easily added to our perception system only by defining a corresponding sensor model. We have shown also that such a representation is able to model other types of data like maps, successfully conveying geometrical and contextual information.

[This page intentionally left blank.]

Chapter 5

Incorporation of prior knowledge into perception system

"Nie zgadzam się z matematyką.
Uważam, że suma zer daje groźną liczbę."
"I do not agree with mathematics.
The sum total of zeros is a frightening figure."

Stanisław Jerzy Lec, *Myśli nieuczesane (Unkempt thoughts)*

Contents

Having defined sensor and map models and created corresponding grids is only the first step in the processing involved in environment perception. The new patterns emerge only after successfully fusing all these pieces of information together. In this chapter, we therefore present the method for combining information about vehicle's environment from heterogeneous sources: on-board sensors and precomputed cartographic maps. The method combines these different type of information and additionally takes advantage of meta-knowledge about the context in which the intelligent car evolves; this meta-data being obtained from the same digital map. During the information fusion, the map data allows us to refine hypotheses provided by lidar scan data. Our approach

Contrary to the Simultaneous Localization, Mapping and Moving Object Tracking (SLAMMOT) methods that build a map of the environment on the fly, our approach uses only maps prepared beforehand by external providers. As more and more maps are accessible nowadays and they are even more precise, accurate and complete than ever, such an approach seems to be deeply founded and will be likely generalised and developed. In the proposed method, we try to enhance a perception system that deals with dynamic environments like crowded city centres.

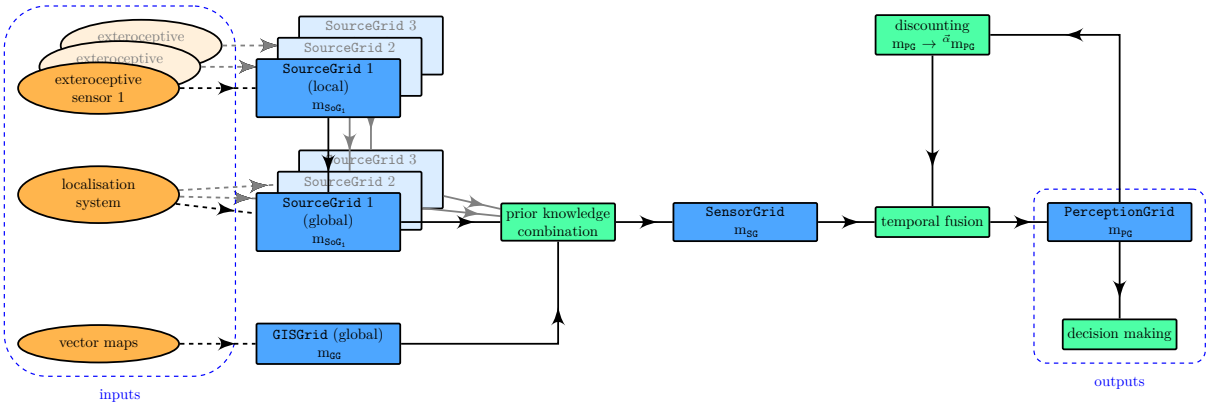


Figure 5.1 – Overview of the perception system.

In order to ameliorate and alleviate the deficiencies of available sensors, digital maps are considered as an additional source of information on a par with other sources, e.g. sensors. This fact is crucial for the method as the map cannot be regarded as an infallible source of absolute truth that always gives a correct result. If this were to be true, one could simply condition (e.g. in sense of Bayes' conditioning) the information obtained from sensors using the map data. Instead, we treat maps rather as just another source of information. In this manner, the handling of information is uniform and completely comprised in the data fusion process.

While designing our system, an important constraint was that a perception system must be easily adaptable for various configurations. This variety may be owed to a failing sensor, to a system modification or update as well as, as a common industrial practice, because of using the same system for a series of different vehicle models. An addition of a sensor should improve the overall performance of the system, whereas a removal could lead to its degradation. The logics of data processing should not nevertheless be altered. In Figure 5.1, we recall the architecture with which we achieved these goals.

The previous chapter dealt with processing that depends on the type of the sensor. In this chapter, we will handle invariant parts of our perception system that may seamlessly adapt themselves to different sensors or map providers given corresponding sensor models. This part of our perception system is depicted by Figure 5.2.

One of the reasons for which we employ map data is to deduce meta information about the environment and hence restrain possible types of objects to be detected. Secondly, we propose to control different perception dynamics in the same scene, thanks to the map. For instance, once a building is perceived by the perception module, this information should be stored for future use and not forgotten. On the contrary, mobile objects with low remanence in the scene should be updated rapidly, i.e. forgotten or discarded almost as soon as they disappear from the range of view.

In this context, we will use the term remanence to denote the persistence of a given piece of information. It is related to the time that an object is supposed to spend in the environment represented by a single cell of perception grid before being discarded if it is not seen again. As a consequence, this method allows to manage the occulted zones and objects. The notion of object remanence will be of high importance in our method and so we devote the whole Chapter 6 to the methods for information discounting.

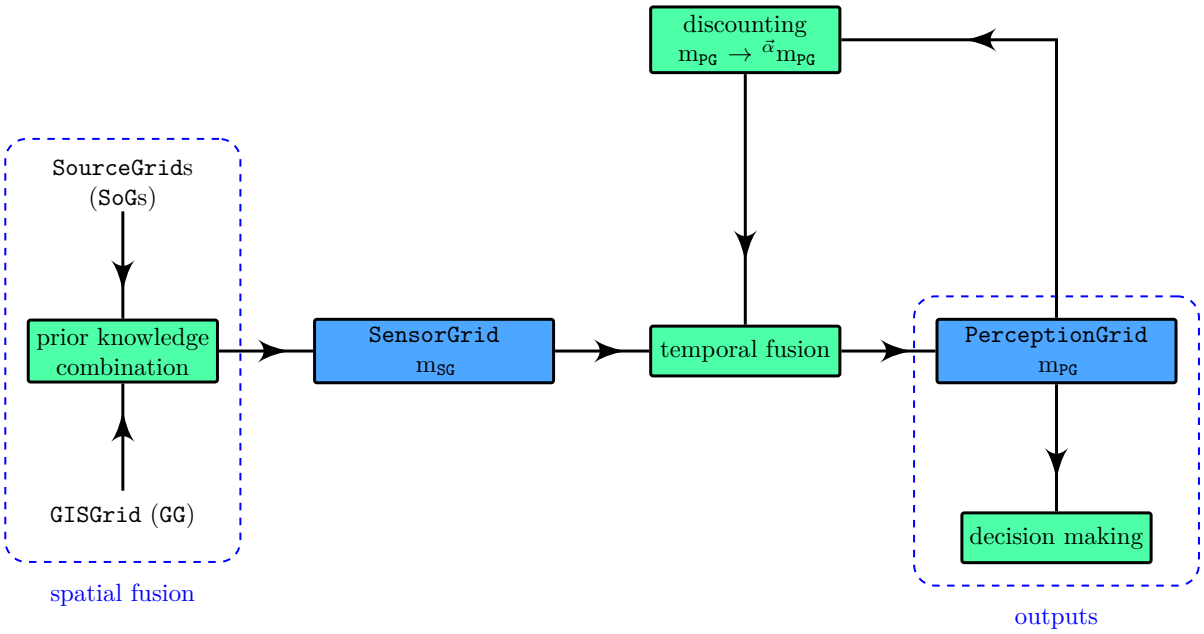


Figure 5.2 – Part of the perception system where the information fusion takes place independently of exact sensor types.

5.1 Spatial fusion: from SourceGrid to SensorGrid

The proposed method incorporates maps into sensor data in order to ameliorate the perception. The part responsible for this stage is schematically depicted in Figure 5.2 as *spatial fusion*. The maps are the source of prior information which can be used to gain more insight about the vehicle environment. The fusion of the information from GISGrid with the sensor data stored in the SourceGrids (SoGs) (cf. Figure 5.1) is performed on a cell-by-cell basis.

Grid transformations

At first, all grids are transformed into the same spatial reference frame and into a common frame of discernment Ω_{PG}. If a SoG is in polar coordinates, it has to be converted into Cartesian reference as shows Figure 5.3. In our approach, a bi-linear interpolation has been used for this transform. Each local SourceGrid (SoG) can be transformed into global reference framework using the pose provided by the proprioceptive sensor. Figure 5.3 illustrates the general idea of this process.

Next, a transformation of the Cartesian SoG is applied in order to obtain a world-referenced grid. This transformation consists of one rotation and one translation. The rotation is done with a bi-linear transformation, because one cell may be partially projected on many cells. Bi-linear transformation can interpolate values, so, in the transformed cell, masses are set to mean values of the neighbourhood of the polar cell. Such a method can cause a phenomena of edge smoothing, but a well-chosen grid size renders this effect negligible.

Incorporating map data

At this stage, the prior knowledge from maps is injected into sensor data. If multiple SoGs exist, the conjunctive rule of combination (denoted $\textcircled{\cap}$) is used to combine them before further processing, as expressed by Equation 5.1. Given that the masses in SoGs, indexed from 1 to N_s, are denoted m_{S_i},

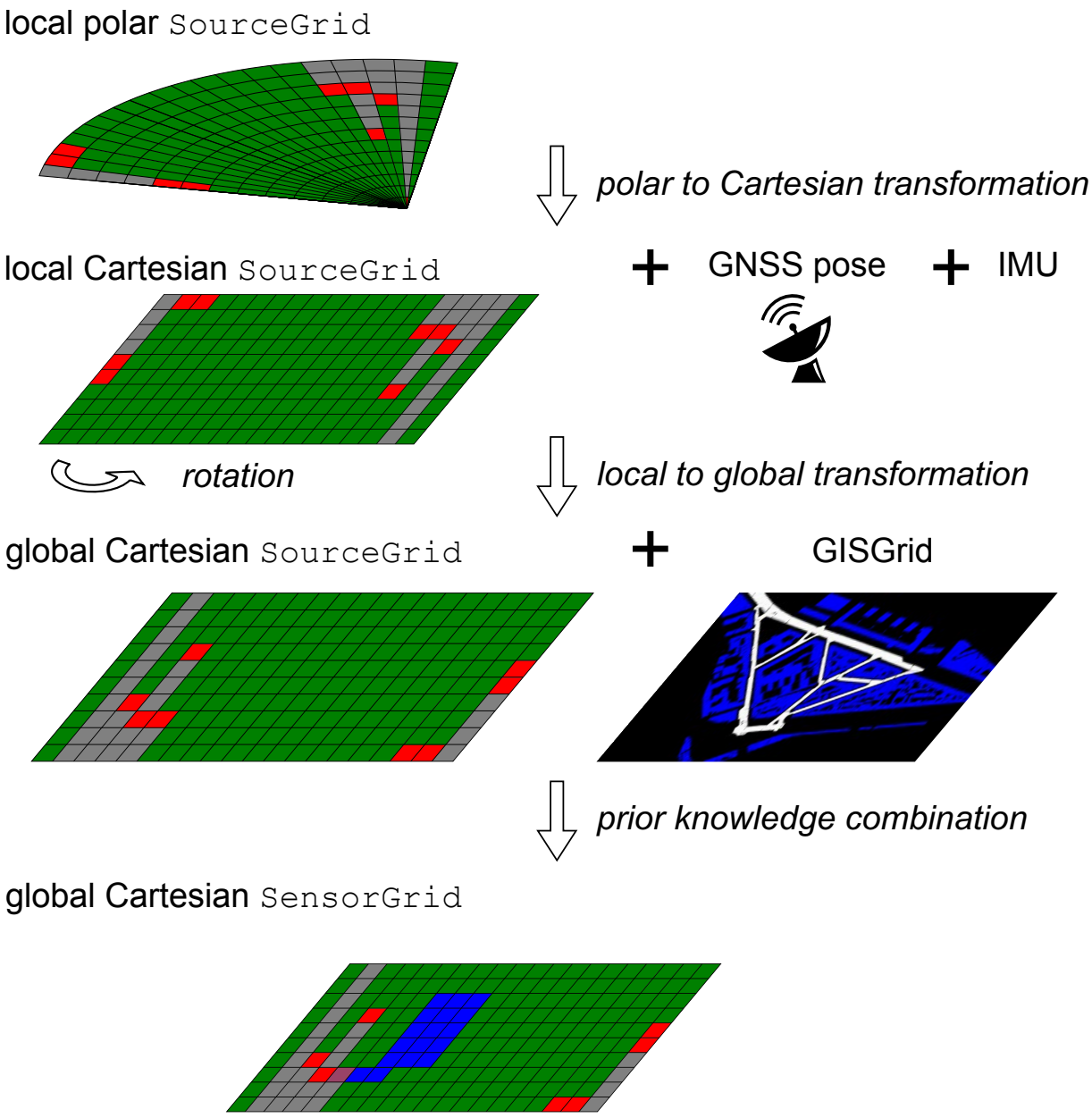


Figure 5.3 – Transformation from SourceGrid (SoG) to SensorGrid (SG).

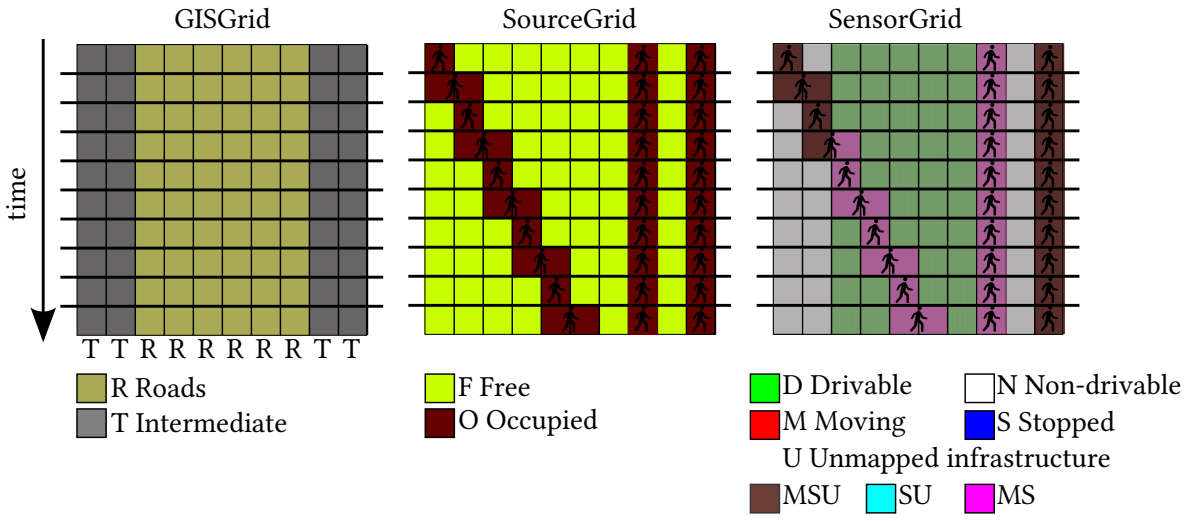


Figure 5.4 – Example of incorporating map data into SensorGrid (SG) on a 1D grid. T represents the intermediate space and R – road surface.

the combined mass function is m_S. At the next step, the conjunctive normalised operator is applied to combine this resulting grid (having cells with masses m_S) and the GISGrid (GG). Equation 5.2 describes this process of spatial fusion.

$$\mathrm{m}_S^{\Omega_{PG},(t)} = \bigcirc\!\!\!\!\!\!\bigcap_{i=1}^{N_s} \mathrm{m}_{S_i}^{\Omega_{PG},(t)} \tag{5.1}$$

$$\mathrm{m}_{SG}^{\Omega_{PG},(t)} = \mathrm{m}_S^{\Omega_{PG},(t)} \oplus \mathrm{m}_{GG}^{\Omega_{PG}} \tag{5.2}$$

The conjunctive normalised rule of combination (also called Dempster's rule and denoted $\oplus$) was chosen because the geodata from maps and the sensor data are considered to be independent. Furthermore, the sources are supposed reliable, even if errors are possible. At the end of this stage, the resulting grid, SensorGrid, is the combination of the sensor data from SourceGrids with the prior knowledge from GISGrid. SensorGrid is globally referenced and uses the same frame of discernment Ω_{PG} as the PerceptionGrid. SensorGrid, apart from being the output of the spatial fusion, is the input of the temporal fusion.

The Figure 5.4 gives an example of a SensorGrid. In the scene (see Figure 5.4), there are 3 pedestrians: one crossing the road, another staying on the road, and the last on the side-walk. We show in the figure three 1D grids evolving in time. The GISGrid (GG) represents our prior knowledge obtained from the digital map. Both sides are known to be side-walk (intermediate space $\{T\}$) and the centre is assumed to be the road surface $\{R\}$. The SourceGrid (SoG) is a representation of the current sensor data. In this situation the only information provided by the sensor is whether the space is free or occupied. This figure shows the result of spatial fusion, i.e. of map incorporated into sensor data, which is reflected in the SensorGrid (SG).

5.2 Temporal fusion

The temporal fusion serves the role of combining current sensor acquisition with preceding perception result. The sensor information input has been already combined with prior information as described before. The general form of information fusion operation is expressed by Equation 5.3.

$$\mathrm{m}_{PG}^{(t)} = {}^{\alpha}\mathrm{m}'^{\,(t-1)}_{PG} \circledast \mathrm{m}_{SG}^{(t)} \tag{5.3}$$

Fusion steps combine the PerceptionGrid from preceding epoch ${}^{\alpha}\mathrm{m}'^{\,(t-1)}_{PG}$ with the SensorGrid from current epoch $\mathrm{m}_{SG}^{(t)}$ using fusion operator $\circledast$ which belongs to the family of conjunctive operators. PerceptionGrid used for the fusion operation is the result of the process, including the conflict analysis[1] and mass function specialisation denoted by the apostrophe in m'_{PG}, as described in the following paragraphs.

[1]Actually, ${}^{\alpha}\mathrm{m}'^{\,(t-1)}_{PG}$ represents the result of the prediction model: grid from time $t-1$, is used to obtain the predicted state of the world at time t. Hence, both mass functions m_{SG} and m_{PG} describe the same instant of environment state, and so the conflict analysis is justified.

5.2.1 Importance of data discounting

In Equation 5.3, an important detail is the discounting of the mass contained in `PerceptionGrid` (PG). The mass $\mathrm{m}_{PG}^{(t)}$ at instant t is namely computed using the same mass at previous instant $t-1$. Apart from being specialised, as described in the next section, the mass ${}^{\alpha}\mathrm{m}'^{(t-1)}_{PG}$ is also discounted. The need of information discounting comes from the fact that the lapse $\Delta t = t-(t-1)$ is non-null. In other words, the environment state at time t is estimated on the preceding state at time $t-1$. One has to remark that during the period Δt, the environment around the vehicle (as well as the vehicle itself) might have changed. To model this possible change, we discount the information contained in $\mathrm{m}'^{(t-1)}_{PG}$ before combining it with the current sensor measurement $\mathrm{m}_{SG}^{(t)}$.

5.2.2 Mobile object detection by conflict analysis

To exploit dynamic characteristics of the scene, we propose the analysis of inflicted conflict masses. The idea presented in (Moras, Cherfaoui, and Bonnifait 2010) is used here to manage conflict masses. This need arises from the fact that the environment is dynamic. Some authors have elaborated different conflict management to detect changing areas (Ramasso et al. 2010).

Two types of conflict are therefore distinguished. In the proposed fusion scheme, $\emptyset_{FO}$ denotes the conflict induced when a free cell in `PerceptionGrid` is fused with an occupied cell in `SensorGrid`. Analogically, $\emptyset_{OF}$ indicates the conflict mass caused by an occupied cell in `PerceptionGrid` fused with a free cell in `SensorGrid`. Conflict masses are given by:

$$\begin{aligned}\mathrm{m}_{PG}^{(t)}(\emptyset_{OF}) &= \mathrm{m}_{PG}^{(t-1)}(O)\cdot\mathrm{m}_{SG}^{(t)}(F)\\ \mathrm{m}_{PG}^{(t)}(\emptyset_{FO}) &= \mathrm{m}_{PG}^{(t-1)}(F)\cdot\mathrm{m}_{SG}^{(t)}(O)\end{aligned} \tag{5.4}$$

where $\mathrm{m}(O) = \sum_{A\subseteq\{I,M,S,U\},\,A\neq\emptyset}\mathrm{m}(A)$ and $\mathrm{m}(F) = \sum_{A\subseteq\{D,N\},\,A\neq\emptyset}\mathrm{m}(A)$. In the ideal case where no noise is present in the input data, conflicts $\emptyset_{FO}$, $\emptyset_{OF}$ represent, respectively, appearance and disappearance of an object.

5.2.3 Static object detection

5.2.3.1 Occupancy accumulator

Distinguishing between static and dynamic obstacles is a crucial issue in dynamic environments. To meet this need, an accumulator ζ is introduced. Secondly, a mass function specialisation using ζ is performed to distinguish temporarily stopped objects from those that are moving.

ζ is defined in each cell in order to include temporal information on the cell occupancy. For this purpose, a gain $\delta \in [0,\,1]$ and an decrement-to-increment ratio γ have been chosen. Section 7.3 explains what factors influence the computation of these parameters and sheds some light on their physical interpretation.

$$\zeta^{(t)} = \zeta^{(t-1)} + \delta\cdot[\,\mathrm{m}_{PG}(O)\cdot(1-\mathrm{m}_{PG}(\emptyset)) - \gamma\cdot(1-\mathrm{m}_{PG}(O))] \tag{5.5}$$

Value of ζ is consequently clamped into range $[0,\ 1]$ so that it could be used in a specialisation matrix.

$$\zeta^{(t)} = \max\left(0, \min\left(1,\ \zeta^{(t)}\right)\right) \tag{5.6}$$

5.2.3.2 Mass specialisation

Accumulator ζ behaviour is described in Section 5.4. ζ brings a piece of evidence about a more specific set, here the static classes. ζ values are used to specialise mass functions in `PerceptionGrid` using Equation 5.7. Masses on elements of $\mathrm{m}_{PG}^{(t)}$ are transferred to $\mathrm{m'}_{PG}^{(t)}$ according to specialisation matrix $S^{(t)}$ as presented by Equation 5.7. It is noteworthy to mention that $S^{(t)}(A,\ B)$ represents the ratio of the mass attributed to set B that will be transferred to set A.

$$\mathrm{m'}_{PG}^{(t)}(A) = \sum_{B \subseteq \Omega_{PG}} S^{(t)}(A,\ B) \cdot \mathrm{m}_{PG}^{(t)}(B) \qquad \forall A \subseteq \Omega_{PG} \tag{5.7}$$

A specialisation matrix $S^{(t)}$ is used to do the mass transfer. Matrix $S^{(t)}$ is identically zero except for the following elements:

$$\left.\begin{aligned} S^{(t)}(A \setminus \{M\},\ A) &= \zeta^{(t)} \\ S^{(t)}(A,\ A) &= 1 - \zeta^{(t)} \\ S^{(t)}(A,\ A) &= 1 \end{aligned}\right\} \quad \begin{aligned} &\forall A \subseteq \Omega_{PG} \text{ and } A \ni \{M\} \\ &\forall A \subseteq \Omega_{PG} \text{ and } A \not\ni \{M\} \end{aligned} \tag{5.8}$$

The idea behind the specialisation matrix and the accumulator is that moving objects are differentiated from static or stopped objects. The mass attributed to sets $\{U,\ S,\ M\}$ will be transferred to $\{U,\ S\}$ and from $\{S,\ M\}$ to $\{S\}$, respectively. Additionally, the value of the transferred mass is proportional to the time that the cell in question stayed occupied.

5.2.4 Fusion rule

The fusion rule $\circledast$ from Equation 5.3 is based on the conjunctive rule of combination, but it has been influenced by Yager's fusion operator (Yager 1987). An effort has been made to adapt it for mobile object detection. Yager's operator has the advantage of not attributing more mass than given by the sources to any class except for the unknown Ω (which is the case with normalised rules).

As indicated above, some modifications to the conjunctive rule have to be performed in order to distinguish between moving and stationary objects. These changes consist in transferring the mass corresponding to a newly appeared object to the class of moving objects $\{M\}$ as described by Equation 5.9. Fusion rule $\circledast$ has no longer the commutative nor associative properties, but the temporal fusion is performed sequentially and the order is imposed.

$$\begin{aligned} (\mathrm{m}_1 \circledast \mathrm{m}_2)(A) &= (\mathrm{m}_1 \textcircled{\cap} \mathrm{m}_2)(A) && \forall A \subsetneq \Omega \text{ and } A \neq M \\ (\mathrm{m}_1 \circledast \mathrm{m}_2)(\{M\}) &= (\mathrm{m}_1 \textcircled{\cap} \mathrm{m}_2)(\{M\}) + (\mathrm{m}_1 \textcircled{\cap} \mathrm{m}_2)(\emptyset_{FO}) \\ (\mathrm{m}_1 \circledast \mathrm{m}_2)(\Omega) &= (\mathrm{m}_1 \textcircled{\cup} \mathrm{m}_2)(\Omega) + (\mathrm{m}_1 \textcircled{\cap} \mathrm{m}_2)(\emptyset_{OF}) \\ (\mathrm{m}_1 \circledast \mathrm{m}_2)(\emptyset_{FO}) &= 0 \\ (\mathrm{m}_1 \circledast \mathrm{m}_2)(\emptyset_{OF}) &= 0 \end{aligned} \tag{5.9}$$

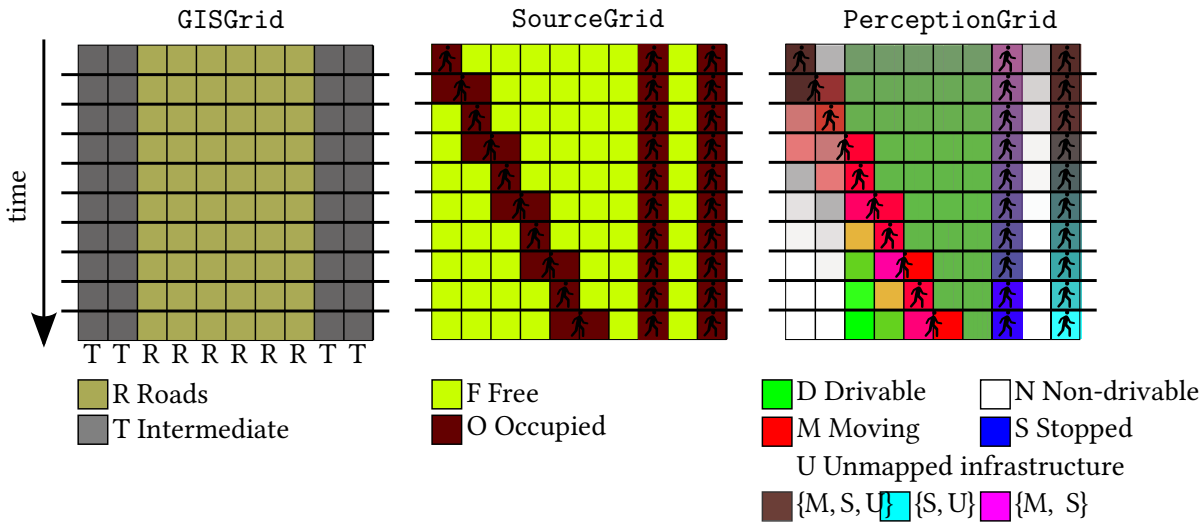


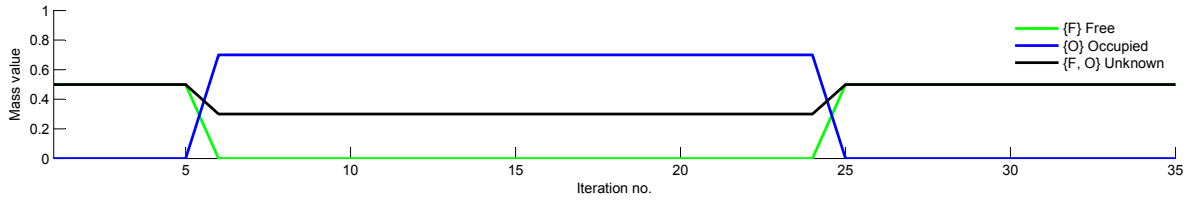
Figure 5.5 – Example of temporal fusion on a 1D grid. Please see Figure 5.4 for legend.

The above mentioned steps conduct to the construction of a `PerceptionGrid`, which is the system output. Such a `PerceptionGrid` contains rich information on the environment state. It includes the knowledge on mobile and static cells divided in classes.

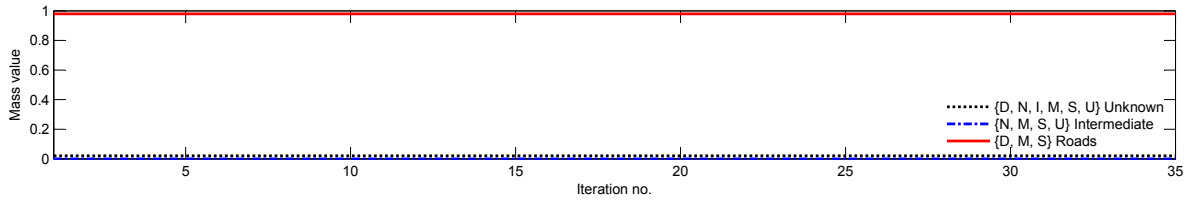
5.3 Illustrative examples

This section aims to illustrate the behaviour of our perception system with the aid of an instructive example. The example is composed of three 1D grids (for the sake of simplicity) that change over time, meaning that the scenario has to be read from top to bottom, line by line. Colours are determined by the mass function of the cell, as specified in the legend. It should be remarked that a mass function can contain more than one focal element, and for this reason the effective colour is a mix of corresponding classes.

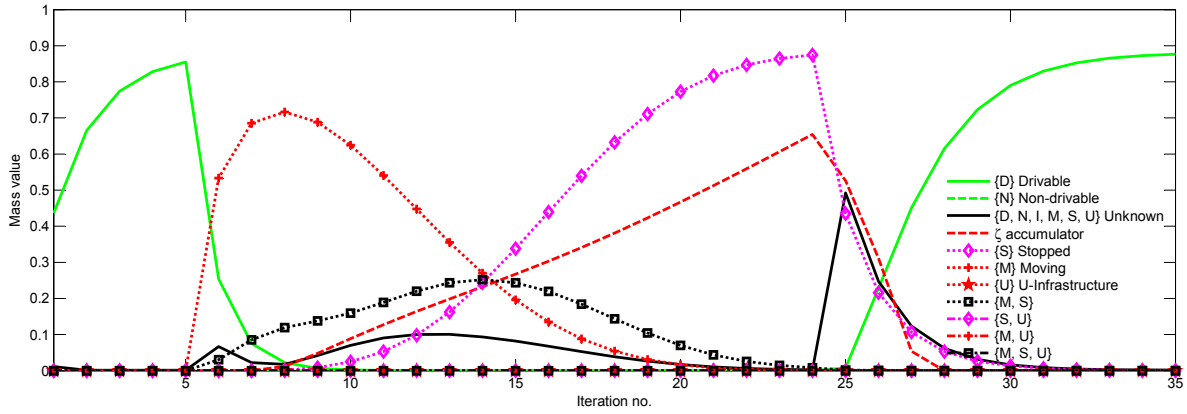
Figure 5.5 uses the same scenario and illustration conventions as the preceding Figure 5.4. The difference lies in the last grid `PerceptionGrid` (PG) that shows the step-by-step result of the perception system, i.e. it includes both the spatial and temporal fusion. Its initial state is complete ignorance (the entire mass assigned to unknown Ω). As the moving pedestrian walks, the movement is detected in different grid cells. Behind the pedestrian, previously occupied cells gradually become free. Free, drivable and non-drivable cells have their masses increased as the sensor confirms the same information. The information relative to the two stopped pedestrians is processed differently using the map information. The pedestrian on the road is treated as a moving or stopped object $\{M, S\}$ at first, since initially the cell was unknown. Then, as the situation develops over time, the pedestrian is seen as a stopped object, which illustrates how the accumulator and the specialization work. This change is visible as the colour changes from purple to blue. The other pedestrians on the side-walk are at first treated as either moving, stopped or unmapped obstacles $\{M, S, U\}$. By the end, the pedestrian who remained on the side-walk is detected as a stopped obstacle *or* as unmapped infrastructure $\{U\}$. Indeed, it is impossible to distinguish between a stopped pedestrian and, for instance, a lamppost. The pedestrian crossing the road is quickly detected as moving $\{M\}$.



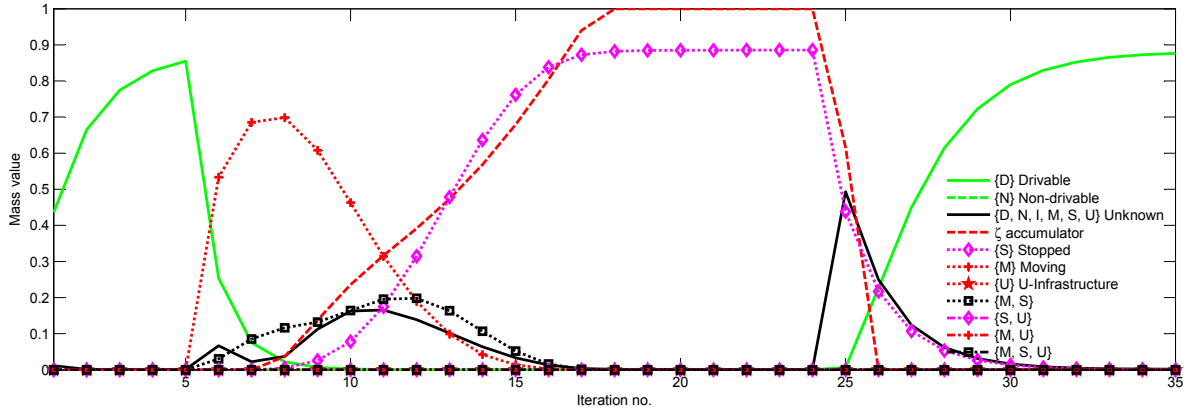
(a) `SourceGrid`.



(b) `GISGrid`.

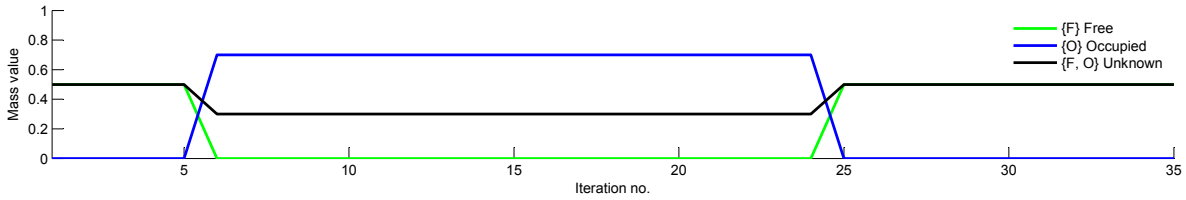


(c) `PerceptionGrid`. $\gamma = 6, \delta = 0.05$

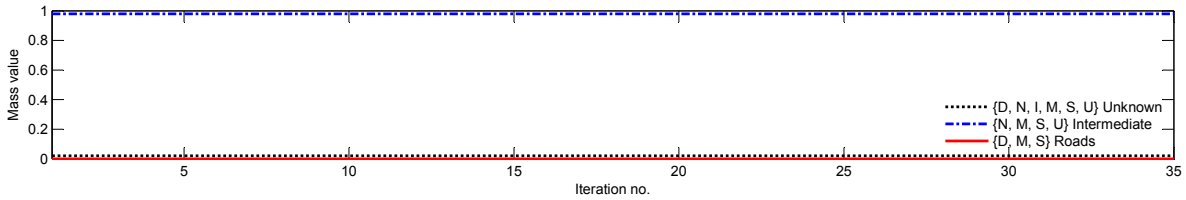


(d) `PerceptionGrid`. $\gamma = 6, \delta = 0.15$

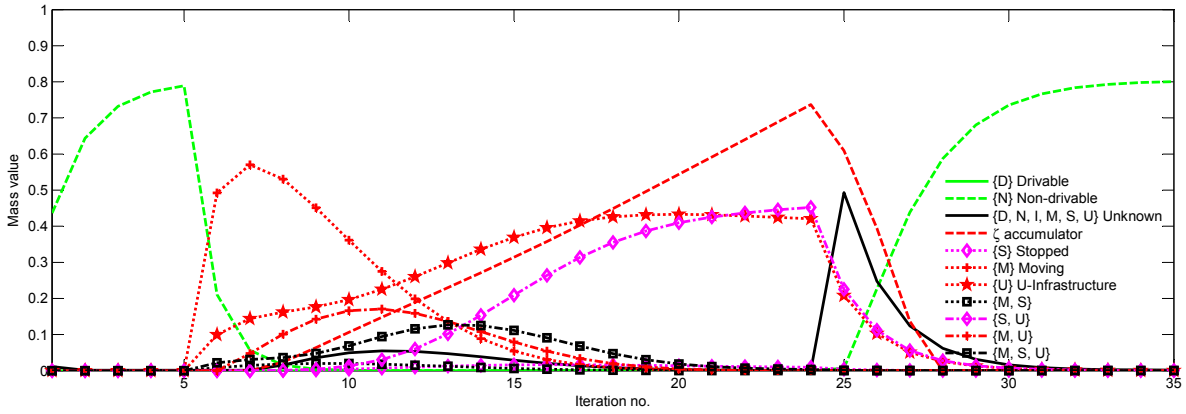
Figure 5.6 – Fusion rule behaviour in the road context.



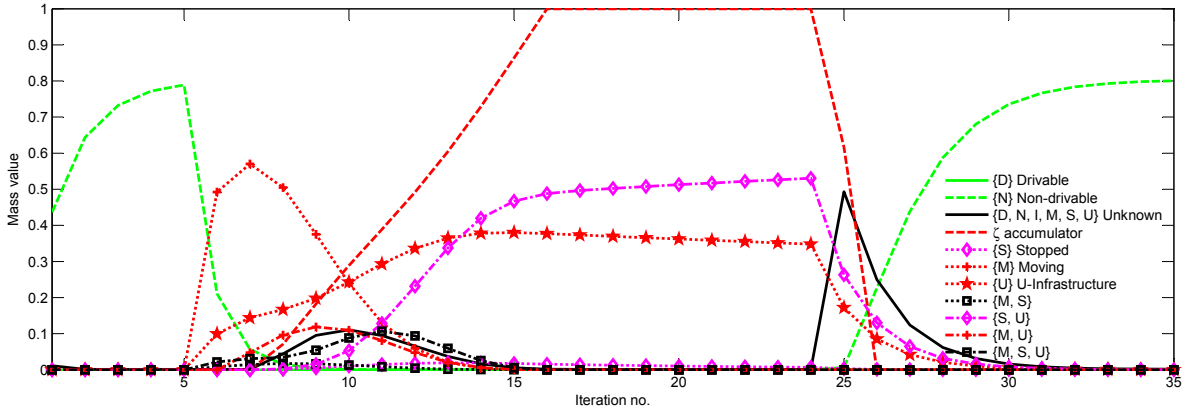
(a) `SourceGrid`.



(b) `GISGrid`.



(c) `PerceptionGrid`. $\gamma = 6, \delta = 0.05$



(d) `PerceptionGrid`. $\gamma = 6, \delta = 0.15$

Figure 5.7 – Fusion rule behaviour in the intermediate space context.

5.4 Temporal fusion behaviour analysis

The temporal fusion step described in the previous paragraphs is the core part of the proposed fusion scheme. This operation is also a complex one and requires additional comments. Figures 5.6 and 5.7 present the behaviour of the proposed fusion scheme in different contexts: road and intermediate space, respectively.[1]

In both figures, parts a show the evolution of the mass function of the `SourceGrid`, i.e. the sensor acquisition. Performed simulation models a single cell that stays free for 5 iterations, or sensor cycles. Then, the cell stays occupied until iteration no. 24 and, finally, becomes free again. This situation can be interpreted as an appearance of an object which stays for a longer time and leaves its place in the end.

Part b represents the mass function of the `GISGrid`, so the prior information obtained from maps. In the road context, the major part of the mass is attributed to the $\{D,\ M,\ S\}$ class (drivable free space, moving or stopped object), shown as dotted red line. In contrast, in the intermediate space context, the affected class is the $\{N,\ M,\ S,\ U\}$ (non-drivable free space, moving or stopped object, or unmapped infrastructure), depicted by dash-dot blue line.

Parts c and d constitute the most important part of the comparison. They present the evolution of the `PerceptionGrid` mass function with different values of parameter δ.

Looking at Figures 5.6c and 5.6d, one can observe that the evolution of free drivable space mass $\{D\}$ (solid green line) follows a well-known pattern. During the iterations 1 to 5, as the sensor detects the cell as free repeatedly, the mass on $\{D\}$ augments due to conjunctive rule behaviour. When the cell is detected as occupied in the 6th cycle, the pieces of information in `SourceGrid` and `PerceptionGrid` become contradictory and create a conflict, which is transferred to moving objects class $\{M\}$ (dotted red line with plus signs). The mass of movable (moving or stopped) class $\{M,\ S\}$ (dotted black line with squares) is affected as well due to the discounting operation.

The cell stays occupied until iteration 25. During this period, accumulator ζ increases progressively, which causes the mass transfer from $\{M,\ S\}$ to stopped class $\{S\}$ (dotted magenta line with diamonds)[2]. The $\{M\}$ mass diminishes as the cell stays occupied and finally approaches zero.

When the cell gets free again at iteration 25, a high peak in unknown $\Omega = \{D,\ N,\ I,\ M,\ S,\ U\}$ mass value is observed. This behaviour arises because of the transfer of $\emptyset_{OF}$ conflict mass to Ω set[3]. At the same moment, the accumulator ζ starts to decrease. In the next cycles, the free mass $\{D\}$ grows steadily and the other (occupied) classes diminish rapidly. ζ drops to zero at a much faster rate than it grew before. Such a behaviour is of course attended and should be interpreted as if a stopped object started to move.

Parts c and d present how the change in the accumulator gain δ impacts the mass evolution. Higher value of δ accelerates the classification of a cell as a stopped one, class $\{S\}$. Similarly, higher value of γ ratio would accentuate the drop of the accumulator value when a stopped object leaves the cell

[1] Not all the subsets of the frame of discernment have been shown on these figures, so the sum of visible masses may be less than 1.

[2] The mass transfer from class $\{M,\ S\}$ to $\{S\}$ is just one example. Actually, all the classes that contain class $\{M\}$ are affected and a part of their masses get transferred to their subset without the $\{M\}$ class, e.g., from $\{M,\ S,\ U\}$ to $\{S,\ U\}$.

[3] One could argue that the fusion rule is unnecessarily non-symmetrical here, and the $\emptyset_{OF}$ conflict could be transferred to the free drivable class $\{D\}$. However, it seems prudent to postpone the growth of the free mass, especially in the case of the presented application. Namely, an intelligent vehicle should compensate for aberrant input data obtained from not completely reliable sources.

(iteration 25 onwards). An important moment during the mass evolution is the instant at which the cell is no longer believed to be a moving object, but the mass attributed to stopped class is the highest. This happens at iteration 12 and 10 respectively when $\delta = 0.05$ and $\delta = 0.15$. That fact gives another clue about the impact of this parameter on the detection of stopped obstacles. A careful reader would point out that there is no study of behaviour when the parameter γ varies. This is due to the fact that this parameter has been defined analytically as described in Section 7.3 and has a physical interpretation.

The above description is valid for both Figures 5.6 and 5.7 with minor differences. Firstly, in the intermediate space, classes stopped $\{S\}$ and unmapped infrastructure $\{U\}$ cannot be distinguished by the fusion itself and the difference in values comes mostly from the discounting. Namely, the behaviour of masses attributed to $\{M, S\}$ and $\{S\}$ in the road context corresponds to the classes $\{M, S, U\}$, $\{S, U\}$ and $\{U\}$ in the intermediate space. Analogically, instead of drivable space $\{D\}$, it is the non-drivable free space $\{N\}$ that is present in Figure 5.7.

5.5 Conclusion

The fusion of information coming from heterogeneous sources needs special attention and adapted algorithms. In our case, the difficulty was to include the map data into a perception system. Another challenge is to reliably detect and distinguish mobile and static objects in a highly dynamic environment. To achieve these goals we have adapted existing fusion operators, notably Dempster's conjunctive rule and Yager's rule. The management of conflictual information allowed us to characterise dynamic obstacles. Mechanisms like the accumulation of static state and the specialisation of mass functions were used to differentiate stopped objects from moving ones. The preceding perception grid being the result of information fusion serves us to predict the new state of the environment on the next moment of fusion. Due to a non-zero lapse of time between these two instants, we have to compensate for the possible changes in the dynamic environment. For this purpose, the mass functions contained in evidential perception grids are discounted using a temporal discounting method. The importance of this procedure led us to study this subject more profoundly. We have therefore proposed a family of temporal discounting operators that are described in the next chapter.

Chapter 6

Management of ageing information

"Szerzenie niewiedzy o wszechświecie musi być także naukowo opracowane."
"Spreading the ignorance about the universe must be laid down scientifically as well."

Stanisław Jerzy Lec, *Myśli nieuczesane (Unkempt thoughts)*

Contents

The domain of information fusion concerns in great measure the combination of sensor data arriving successively with the passage of time. Past information is often useful and should not be discarded. However, one cannot disregard the fact that the information may worth less and less over time. In order to handle this variation in the subjective value of a piece of information, one needs to manage the age of information for example by applying a discounting operation. In general, the information has to be discounted when we do not have enough certainty in the information source.

In the case of the presented perception system, the need for handling the age of information arises for instance when predicting the future state of the world basing our judgement on out-dated evidence. In this situation, the state of environment at the moment of previous sensor scan is represented by a perception grid. When new data arrive, this grid is used to predict the current state. However, the information contained therein does not correspond to the present state and thus should be discounted. In general, the more a piece of information is out-dated, the more it should be discounted. The discounting operation expresses the fact that the previous perception grid represents the current state of the affair only partially.

Information discounting plays an important role in the theory of belief functions and, generally, in information fusion, for instance in many problems where there is a need to allow for the reliability of a source (Smets 2000). The level of confidence or reliability can depend only on the source, but it can as well depend on the type of evidence being discounted, e.g. for different classes of detected obstacles. Whereas the former case is easily handled by classical discounting operation (see Section 3.5.3), the latter is more complex. The belief functions theory (BFT) makes it possible to model one's opinion about the reliability of an information source using discounting, as described in Section 3.5.3. The point is that more reliable sources get assigned heavier weights than those more unreliable. The result of discounting of the basic belief assignment (bba) m^{Ω} is a new bba ${}^{\alpha}\mathrm{m}^{\Omega}$ (both defined on the frame of discernment (fod) Ω). Discounting factor α may be considered as the level of distrust in this particular source.

The problem shows up when there are various classes of information that do not become out of date at the same pace, one being more persistent than another. Unfortunately, existing methods such as contextual discounting (cf. Section 6.2.1) do not meet all possible use cases. Notably the temporal discounting cannot be modelled in all situations, as we will demonstrate further.

We have studied the considered the postulates that a discounting operation should meet. As a solution, new contextual discounting schemes, conservative, proportional and optimistic, are proposed. We examine the properties of these discounting operations and show, for instance, that the classical discounting is a special case of these schemes. Two motivating cases are discussed: modelling of source reliability and application to temporal discounting.

6.1 Temporal discounting

The trust assigned to the information coming from a source depends on the time elapsed since the acquisition. It comes from the fact that the environment is dynamic and it possibly changes between two consecutive sensor acquisitions. The term *temporal discounting* denotes this particular discounting operation. Temporal discounting can be used to partially "forget" information which is no longer valid. In the domain of mobile perception, the environment changes rapidly and discounting becomes indispensable to avoid keeping obsolete information.

In temporal discounting, the discount factor α does not model the level of reliability assigned to the source as in the classical discounting, but it serves another purpose. Here, α corresponds as well to the speed with which information becomes obsolete. This process is often called "information ageing". Some authors proposed that α is a function of elapsed time $\Delta t = t_{\text{current}} - t_{\text{acquisition}}$ and a remanence characteristic ρ of the event E (information arrival) (Cherfaoui, Denœux, and Cherfi 2010), expressed

as a time value in seconds:

$$\alpha = 1 - \exp \frac{\Delta t}{-\rho(E)} \tag{6.1}$$

When dealing with sensor acquisitions, data often arrive at regular intervals and the processing starts instantly, so α can be fixed to a constant value. In this particular formulation, the remanence ρ is a subjective quantification of the persistence of the information. Without giving it some physical sense, the choice of the function $\rho(E)$ is completely arbitrary. For this reason, we based our reflection on the following postulates, defining a precise sense to the information remanence.

Postulates for temporal discounting

The starting point for our discussion about the temporal discounting is to adopt the following convention. A piece of evidence, described by a mass function m, has a mean lifetime τ and the contained information can be regarded as a decaying quantity. Such a formulation allows us to consider the information in terms of time it rests usable and not in terms of some arbitrary decay constant λ.

Considering the value of information in such a manner may let the reader think about the process of radioactive decay described by Ernest Rutherford in early 1900's (Rutherford and Soddy 1903). As it will be shown further, the below stated postulates will imply that the temporally discounted information should be subject to *exponential decay*, Indeed, we opt for the solution where the information "decays", i.e. a piece of information becomes gradually obsolete. In the following paragraphs, A will denote a set, $A \subseteq \Omega$, about the reliability of which an additional piece of knowledge is available.

Mean lifetime τ and half-life time $t_{1/2}$ The operation of information discounting is at base without any connection to the physical properties of the information source or the described entity. In order to compensate for this lack, we propose to define a time after which a piece of data has lost the half of its value. Here, we use the word value as any objective criterion of usability in a data fusion process.

As the main assumption, an information is usable for a limited period of time. The mean value of this time for a given class of information will be called mean lifetime τ. When τ time has elapsed, the given piece of evidence will be, in average, of no use in the fusion process. The mean lifetime can also be expressed as a decay rate $\lambda = \frac{1}{\tau}$. We will use this rate in calculations only, without more ado about its interpretation. A more intuitive characteristics of exponential decay than the lifetime is the time required for a piece of information to lose a half of its value. In other terms, the mass attributed to a piece of information is two times smaller than the initial mass after half-life time $t_{1/2} = \tau \cdot \ln 2$.

Thanks to this postulate, one can compare the persistence of different information types by comparing their half-life times. As far as different information persistence measures are considered, it is noteworthy that choosing mean lifetime or "life expectancy", i.e. mean time after which a piece of information becomes completely irrelevant, would prohibit the use of exponential functions and so entail some computational complications.

$$^{t_{1/2}}\mathrm{m}(A) = \frac{\mathrm{m}(A)}{2} \qquad \forall A \subsetneq \Omega \tag{6.2}$$

$$\lambda = \frac{\ln 2}{t_{1/2}} \tag{6.3}$$

One can describe this quantity in more general terms. Then, the $1/N$*-life time* $t_{1/N}$ will denote the

time required for the mass attributed to a piece of information to decay to the Nth part of the initial mass after time.

$$ {}^{t_{1/N}}\mathrm{m}(A) = \frac{\mathrm{m}(A)}{N} \qquad \forall A \subsetneq \Omega \tag{6.4} $$

$$ \lambda = \frac{\ln N}{t_{1/N}} \tag{6.5} $$

$$ t_{1/N} = \tau \cdot \ln N \tag{6.6} $$

Order invariance Another important condition for a discounting operation is the order invariance. We claim that the result of discounting should be independent of the order of operations. Indeed, in the case where two discounting operations modify two different classes, it should not be important in which order we apply them.

$$ {}^{t_2}\left({}^{t_1}\mathrm{m}(A)\right) = {}^{t_1}\left({}^{t_2}\mathrm{m}(A)\right) \qquad \forall A \subsetneq \Omega \tag{6.7} $$

In other terms, two (unary) discounting operations should be commutative.

Only age-dependent As the last postulate, we consider that the value of discounted evidence should depend only on the age of the information. The number or order of discounting operations should not matter. Indeed, it is desirable that the frequency or the precise moment of intermediary discount operations at which a piece of information gets discounted, does not change the final result. This postulate is important in case of sensor data processing, as various types of sensing devices can have different frequencies of data acquisition. In the case where two sensors perceive the same scene, the discounted information should rest the same.

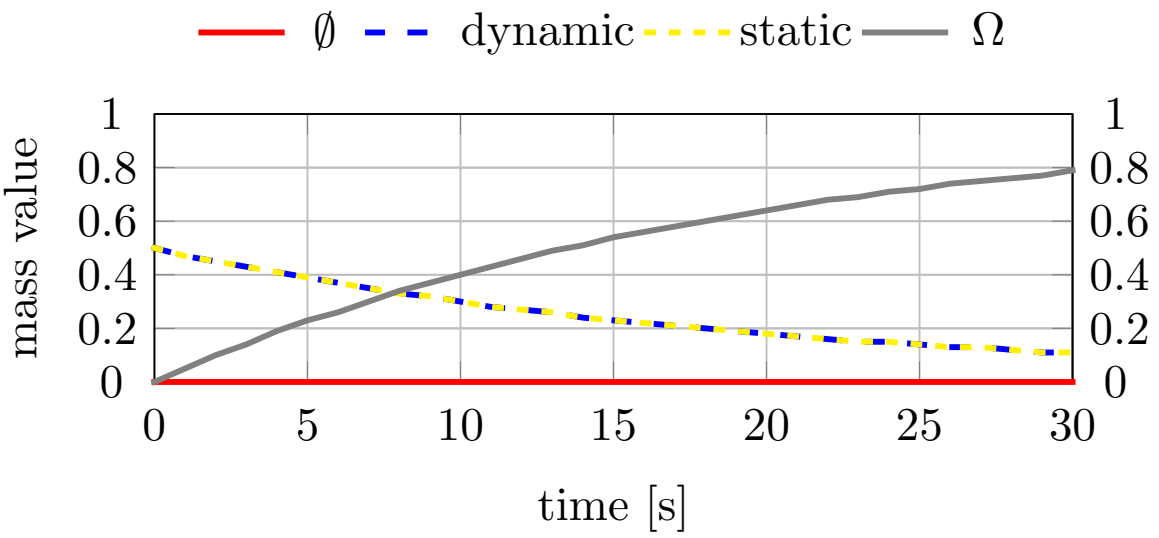
$$ {}^{t_2}\left({}^{t_1}\mathrm{m}(A)\right) = {}^{t_1+t_2}\mathrm{m}(A) \qquad \forall A \subsetneq \Omega \tag{6.8} $$

6.2 Existing methods

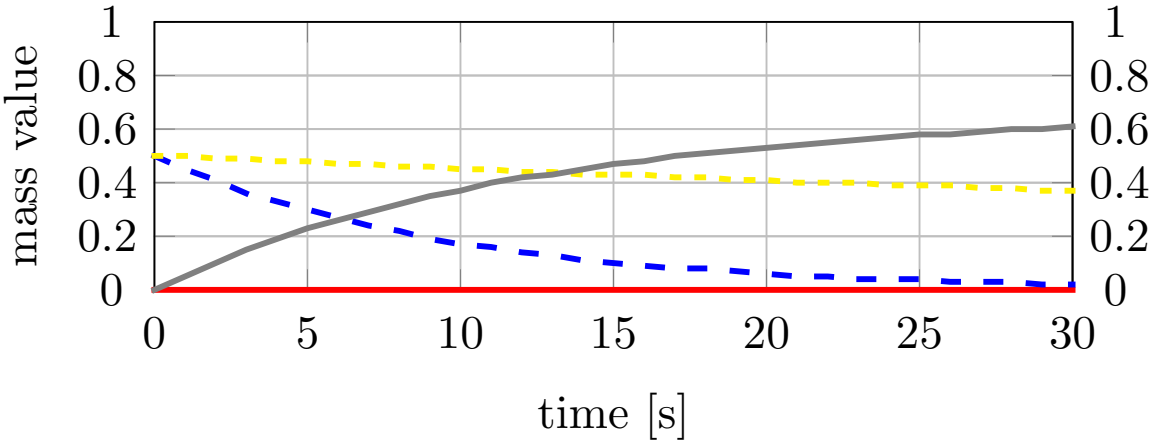
6.2.1 Contextual discounting

Mercier, Quost, and Denœux (2006) and Mercier, É. Lefèvre, and Delmotte (2012) introduced and further developed *contextual discounting* — a type of discounting that makes it possible to adapt the forgetting rate to the context. When more detailed information regarding the confidence ascribed to the sources is available, contextual discounting permits to model this fact. Together with temporal discounting, it can be modelled that different pieces of information become obsolete at different rates. Since meta-knowledge of the robot environment states that some objects (like buildings) do not change rapidly, whereas other do (mobile objects: cars, pedestrians), the contextual discounting process takes these facts into account.

This operation uses vector $\vec{\alpha}$ of discount factors α_θ attributed to elements θ of partition Θ of the frame



(a) Effect of uniform temporal discounting on different classes. Evolution of a mass as a function of time; $\alpha = 0.05$. The masses on the dynamic and static classes are superposed as they decrease at the same rate.



(b) Effect of contextual temporal discounting on different classes. Evolution of a mass as a function of time; $\alpha_{\text{dynamic}} = 0.01$, $\alpha_{\text{static}} = 0.1$. Despite the fact that decay factor for the dynamic class is lower than for the static one $\alpha_{\text{dynamic}} < \alpha_{\text{static}}$, the mass attributed to this class decreases more rapidly than the mass on the static class.

Figure 6.1 – Comparison of discounting behaviour.

of discernment Ω, i.e.:

$$\Theta \subseteq 2^{\Omega} \tag{6.9}$$

$$\Omega = \bigcup_{\theta \in \Theta} \theta \tag{6.10}$$

$$\forall \theta_i, \theta_j \in \Theta, i \neq j : \theta_i \cap \theta_j = \emptyset \tag{6.11}$$

A mass function m_i has to be created for each element θ of the partition Θ, i.e. for $i \in \{1, \ldots, N\}$, where N is the cardinality of Θ. m_i is then defined using corresponding discount factor α_i as follows:

$$\mathrm{m}_i = \begin{cases} 1 - \alpha_i & \text{if } A = \emptyset \\ \alpha_i & \text{if } A = \theta_i \\ 0 & \text{otherwise} \end{cases} \tag{6.12}$$

The discounted mass function ${}^{\vec{\alpha}}_{\cup,\Theta}\mathrm{m}$ of a bba m is then computed using the disjunctive combination (denoted $\textcircled{\cup}$) of the input mass function and each function m_i. By definition of the contextual discounting, factor α_i is responsible for discounting the masses on classes being the complement of θ.

$${}^{\vec{\alpha}}_{\cup,\Theta}\mathrm{m} = \mathrm{m} \,\textcircled{\cup}\, \mathrm{m}_\Theta \tag{6.13}$$

$$\mathrm{m}_\Theta = \mathrm{m}_1 \,\textcircled{\cup}\, \mathrm{m}_2 \,\textcircled{\cup} \ldots \textcircled{\cup}\, \mathrm{m}_i \,\textcircled{\cup} \ldots \textcircled{\cup}\, \mathrm{m}_N \tag{6.14}$$

We give an example of application using the frame of discernment $\Omega_{PG} = \{D,\ N,\ S,\ M,\ I,\ U\}$ used in the presented perception method and the coarsening $\Theta = \{\theta_{static},\ \theta_{dynamic}\}$. Factor $\alpha_{dynamic}$ was attributed to the free space as well as to the moving objects $\theta_{dynamic} = \{D,\ N,\ M\}$. Discount rate α_{static} was assigned to the static objects $\theta_{static} = \{S,\ I,\ U\}$.

In our case, the role of the contextual discounting is to control the remanence of different classes. Higher remanence is attributed to static, slowly evolving classes and lower level of persistence can be assigned to rapidly changing, dynamic contexts. Remanence level ρ, if known, can be used to calculate discount factor α for the complement of θ using some decreasing function $f : \mathcal{R} \to [0, 1]$ like $\alpha^c = \frac{1}{1+\rho}$. Nevertheless, discount factors $\vec{\alpha}$ can be learnt using the learning algorithm presented in (Mercier, Quost, and Denœux 2006). In this case, learnt factors $\vec{\alpha}$ would serve to quantify the remanence ρ. The discounted mass function ${}^{\vec{\alpha}}_{\cup,\Theta}\mathrm{m}$ is calculated using:

$${}^{\vec{\alpha}}\mathrm{m} = \mathrm{m} \,\textcircled{\cup}\, \mathrm{m}_{\theta_{static}} \,\textcircled{\cup}\, \mathrm{m}_{\theta_{dynamic}} \tag{6.15}$$

Figure 6.1b presents the evolution of various mass functions when contextual discounting is applied. The example uses partition $\Theta = \{\text{dynamic},\ \text{static}\}$ and discount factors $\alpha_{\text{dynamic}} = 0.01$, $\alpha_{\text{static}} = 0.1$. To compare, a classical discounting, presented in Figure 6.1a, behaves differently. With uniform discounting, all masses are being forgotten at the same rate. Contextual discounting allows to slowly forget classes of high remanence and to discount more rapidly less remanent classes. One of the inconveniences of this method is the fact that reliability factors are attributed to a partition of the frame of discernment, which excludes cases where reliability is known for intersecting subsets of Ω.

Generalised contextual discounting The limitation of the contextual discounting that discount factors may only be defined for a partition of the frame of discernment has been addressed in (Mercier,

Denœux, and Masson 2010; Mercier, É. Lefèvre, and Delmotte 2012) where *generalised contextual discounting* is proposed as a correction mechanism. Again, vector $\vec{\alpha}$ of discount factors is used, but here, they can be defined also for intersecting sets. The method employs the idea of discounting disjunctive weights ν of a canonical decomposition of a bba.

6.3 Conservative, optimistic and proportional discounting

The before formulated postulates make us arrive at the conclusion that the temporal discounting operation should be represented as a Poisson process with decay constant λ. In this way, we obtain a simple equation for information ageing:

$$^{t}\mathrm{m}(A) = \mathrm{m}(A) \cdot e^{(-\lambda t)} \qquad \forall A \subsetneq \Omega \tag{6.16}$$

As a departure point for the design of an operation of discounting, a few hypotheses have been set. First and foremost, information source S is supposed to excessively encourage set of solutions A and, therefore, $\mathrm{m}(A)$ should be discounted by factor α_A corresponding to A. The behaviour of the new discounting operation should be close to the behaviour of classical discounting. Mass of conflict $\mathrm{m}(\emptyset)$ shall get discounted and new schemes should generalise the classical one. Moreover, setting a non-zero discount factor for set θ should entail the discounting of mass attributed to θ, whereas masses of sets having no elements in common with θ should rest unchanged[1]. Such a behaviour is opposite to contextual discounting proposed by Mercier et al. (Mercier, Quost, and Denœux 2005) that retains mass attributed to θ and discounts *other* sets, which we judge counter-intuitive especially in case of many classes, but well-justified and conform to the proposed interpretation, see (Mercier, Quost, and Denœux 2006, Example 2). Finally, we postulate that discounted mass of set θ should be transferred to Ω and not to its other superset being a proper subset of Ω, since doing so would imply additional knowledge about the state of the represented entity.

6.3.1 Conservative discounting

Conservative discounting presents a pessimistic approach to the discounting. As stated before, the attribution of $\mathrm{m}_S(A)$ by source S is excessive and this mass should be discounted by α_A. Let us suppose now that some meta-knowledge states additionally that the affectation of masses to supersets of A by source S is highly dependent on class A. Bearing in mind the above statement, the mass attributed to AB should be discounted in the same manner as $\mathrm{m}(A)$.

Definition 28 (Conservative discounting)**.** *In conservative discounting, set θ, the empty set $\emptyset$ and all sets having at least one element in common (a non-empty intersection) with θ are discounted by factor α_θ.*

[1] Except for frame of discernment Ω, since masses are transferred to this set.

When Θ contains only one element, $\Theta = \{\theta\}$:

$$\overset{\alpha}{c}\mathrm{m}(\emptyset) = \mathrm{m}(\emptyset) \cdot (1 - \alpha_\theta) \tag{6.17}$$

$$\overset{\alpha}{c}\mathrm{m}(A) = \mathrm{m}(A) \cdot (1 - \alpha_\theta) \qquad \forall A \subsetneq \Omega,\ A \cap \theta \neq \emptyset \tag{6.18}$$

$$\begin{aligned}\overset{\alpha}{c}\mathrm{m}(\Omega) &= 1 - \sum_{B \subsetneq \Omega} \overset{\alpha}{c}\mathrm{m}(B) \\ &= \mathrm{m}(\Omega) \cdot (1 - \alpha_\theta) + \alpha_\theta \cdot \left[\mathrm{m}(\emptyset) + \sum_{\substack{B \subseteq \Omega \\ B \cap \theta \neq \emptyset}} \mathrm{m}(B) \right]\end{aligned} \tag{6.19}$$

Generalising this behaviour to any $\Theta \subseteq 2^\Omega$, one obtains:

$$\vec{\overset{\alpha}{c}}\mathrm{m}(\emptyset) = \mathrm{m}(\emptyset) \cdot \prod_{\theta \in \Theta} 1 - \alpha_\theta \tag{6.20}$$

$$\vec{\overset{\alpha}{c}}\mathrm{m}(A) = \mathrm{m}(A) \cdot \prod_{\substack{\theta \in \Theta \\ A \cap \theta \neq \emptyset}} 1 - \alpha_\theta \qquad \forall A \subsetneq \Omega, A \neq \emptyset \tag{6.21}$$

$$\begin{aligned}\vec{\overset{\alpha}{c}}\mathrm{m}(\Omega) = {} & \mathrm{m}(\Omega) \cdot \prod_{\theta \in \Theta} 1 - \alpha_\theta \\ & + \mathrm{m}(\emptyset) \cdot \prod_{\theta \in \Theta} \alpha_\theta \\ & + \sum_{A \subseteq \Omega} \left[\mathrm{m}(A) \cdot \prod_{\substack{\theta \in \Theta \\ A \cap \theta \neq \emptyset}} \alpha_\theta \right]\end{aligned} \tag{6.22}$$

One remarks that the most discounted mass is $\mathrm{m}(\emptyset)$ which is affected by all discount rates.

6.3.2 Optimistic discounting

Optimistic discounting is based on a hypothesis opposite to the one made in conservative discounting. This time, the meta-information about source S asserts that masses of supersets of A are affected independently of class A. These masses shall *not* be discounted by α_A. On the other hand, all subsets of A will be affected in the same way as A.

This type of discounting can be expressed for any $\Theta \subseteq 2^\Omega$ by:

$$\vec{\overset{\alpha}{o}}\mathrm{m}(\emptyset) = \mathrm{m}(\emptyset) \cdot \prod_{\theta \in \Theta} 1 - \alpha_\theta \tag{6.23}$$

$$\vec{\overset{\alpha}{o}}\mathrm{m}(A) = \mathrm{m}(A) \cdot \prod_{\substack{\theta \in \Theta \\ A \subseteq \theta}} 1 - \alpha_\theta \qquad \forall A \subsetneq \Omega, A \neq \emptyset \tag{6.24}$$

$$\begin{aligned}\vec{\overset{\alpha}{o}}\mathrm{m}(\Omega) = {} & \mathrm{m}(\Omega) \cdot \prod_{\substack{\theta \in \Theta \\ \Omega \subseteq \theta}} 1 - \alpha_\theta \\ & + \mathrm{m}(\emptyset) \cdot \prod_{\theta \in \Theta} \alpha_\theta \\ & + \sum_{A \subseteq \Omega} \left[\mathrm{m}(A) \cdot \prod_{\substack{\theta \in \Theta \\ A \subseteq \theta}} \alpha_\theta \right]\end{aligned} \tag{6.25}$$

6.3.3 Proportional discounting

The above proposed schemes represent two extremes of discounting strategies. Conservative one that demonstrates very cautious or even overcautious behaviour which can be resumed as: in case of doubt, do not exclude any possibilities. Indeed, discounting all supersets in the same way as the set in question means that one accepts a possibility that mass of a superset (e.g. AB) corresponds *entirely* to one of its constituents (e.g. A), which, incidentally, has been overestimated and should hence be discounted. Conversely, when one assumes that mass of superset AB depends on a set that has not been excessively evaluated (B), optimistic discounting is used. Such a behaviour can be seen as optimistic or bold, because any doubt about whether to discount a particular set or not implies a negative answer.

Since the above schemes are the extreme cases, a need of an in-between solution appears naturally. A manner of performing this without recurring to mass-dependent computation is to ponder the discount rate by some measure of dependence between a set and it supersets. The straightforward one is the inclusion criterion measuring the ratio between cardinalities of the set and the superset. On the basis of this idea, proportional discounting is expressed by: When Θ contains only one element, $\Theta = \{\theta\}$:

$$
\begin{aligned}
{}^{\alpha}_{p}\mathrm{m}(\emptyset) &= \mathrm{m}(\emptyset) \cdot (1 - \alpha_\theta) && && (6.26)\\
{}^{\alpha}_{p}\mathrm{m}(A) &= \mathrm{m}(A) \cdot (1 - \alpha_\theta) \cdot \frac{|A \cap \theta|}{|A|} && \forall A \subsetneq \Omega, A \cap \theta \neq \emptyset && (6.27)\\
{}^{\alpha}_{p}\mathrm{m}(\Omega) &= \mathrm{m}(\Omega) \cdot (1 - \alpha_\theta) \cdot \frac{|\Omega \cap \theta|}{|\Omega|} && && (6.28)\\
&\quad + \alpha_\theta \cdot \frac{|A \cap \theta|}{|A|} \cdot \left[\mathrm{m}(\emptyset) + \sum_{\substack{B \subseteq \Omega \\ B \cap \theta \neq \emptyset}} \mathrm{m}(B) \right]
\end{aligned}
$$

Generalising to any Θ, we obtain:

$$
\begin{aligned}
{}^{\vec{\alpha}}_{p}\mathrm{m}(\emptyset) &= \mathrm{m}(\emptyset) \cdot \prod_{\theta \in \Theta} 1 - \alpha_\theta && && (6.29)\\
{}^{\vec{\alpha}}_{p}\mathrm{m}(A) &= \mathrm{m}(A) \cdot \prod_{\substack{\theta \in \Theta \\ A \cap \theta \neq \emptyset}} 1 - \alpha_\theta \cdot \frac{|A \cap \theta|}{|A|} && \forall A \subsetneq \Omega, A \neq \emptyset && (6.30)\\
{}^{\vec{\alpha}}_{p}\mathrm{m}(\Omega) &= \mathrm{m}(\Omega) \cdot \prod_{\theta \in \Theta} 1 - \alpha_\theta \cdot \frac{|\Omega \cap \theta|}{|\Omega|} && && (6.31)\\
&\quad + \mathrm{m}(\emptyset) \cdot \prod_{\theta \in \Theta} \alpha_\theta \\
&\quad + \sum_{A \subseteq \Omega} \left[\mathrm{m}(A) \cdot \prod_{\substack{\theta \in \Theta \\ A \cap \theta \neq \emptyset}} \alpha_\theta \cdot \frac{|A \cap \theta|}{|A|} \right]
\end{aligned}
$$

6.4 Properties

6.4.1 Generalisation of classical discounting

Proposed discounting schemes generalise classical discounting in the case where $\Theta = \{\Omega\}$. Such a behaviour comes simply from the fact that for any $\theta \in \Theta$, all its subsets will get discounted. Since all

Table 6.1 – Comparative table of the proposed discounting methods. Mass attributed to Ω omitted for clarity, since for all mass functions $\mathrm{m}(\Omega) = 1 - \sum_{A \subsetneq \Omega} \mathrm{m}(A)$. For succinctness, $\beta_i = 1 - \alpha_i$.

A	${}^{\vec{\alpha}}_{o}\mathrm{m}(A)$	${}^{\vec{\alpha}}_{p}\mathrm{m}(A)$	${}^{\vec{\alpha}}_{c}\mathrm{m}(A)$
$\emptyset$	$\beta_1\beta_{2,3}\,\mathrm{m}(\emptyset)$	$\beta_1\beta_{2,3}\,\mathrm{m}(\emptyset)$	$\beta_1\beta_{2,3}\,\mathrm{m}(\emptyset)$
$\{\omega_1\}$	$\beta_1\,\mathrm{m}(\{\omega_1\})$	$\beta_1\,\mathrm{m}(\{\omega_1\})$	$\beta_1\,\mathrm{m}(\{\omega_1\})$
$\{\omega_2\}$	$\beta_{2,3}\,\mathrm{m}(\{\omega_2\})$	$\beta_{2,3}\,\mathrm{m}(\{\omega_2\})$	$\beta_{2,3}\,\mathrm{m}(\{\omega_2\})$
$\{\omega_1,\ \omega_2\}$	$\mathrm{m}(\{\omega_1,\ \omega_2\})$	$(1-\frac{1}{2}\cdot\alpha_1)(1-\frac{1}{2}\cdot\alpha_{2,3})\,\mathrm{m}(\{\omega_1,\ \omega_2\})$	$\beta_1\beta_{2,3}\,\mathrm{m}(\{\omega_1,\ \omega_2\})$
$\{\omega_3\}$	$\beta_{2,3}\,\mathrm{m}(\{\omega_3\})$	$\beta_{2,3}\,\mathrm{m}(\{\omega_3\})$	$\beta_{2,3}\,\mathrm{m}(\{\omega_3\})$
$\{\omega_1,\ \omega_3\}$	$\mathrm{m}(\{\omega_1,\ \omega_3\})$	$(1-\frac{1}{2}\cdot\alpha_1)(1-\frac{1}{2}\cdot\alpha_{2,3})\,\mathrm{m}(\{\omega_1,\ \omega_3\})$	$\beta_1\beta_{2,3}\,\mathrm{m}(\{\omega_1,\ \omega_3\})$
$\{\omega_2,\ \omega_3\}$	$\beta_{2,3}\,\mathrm{m}(\{\omega_2,\ \omega_3\})$	$\beta_{2,3}\,\mathrm{m}(\{\omega_2,\ \omega_3\})$	$\beta_{2,3}\,\mathrm{m}(\{\omega_2,\ \omega_3\})$

sets are subsets of Ω, all of them are affected in the same way (except for Ω itself as expected).

6.4.2 Order invariance

The result of the discounting operations over different classes is invariant to the order of these operations, equally for conservative, optimistic and for proportional discounting. The proof is omitted here, as it is trivial and is based on the commutative property of the multiplication.

$$ {}^{\vec{\alpha}}_{\Theta_2}\left({}^{\vec{\alpha}}_{\Theta_1}\mathrm{m}\right) = {}^{\vec{\alpha}}_{\Theta_1}\left({}^{\vec{\alpha}}_{\Theta_2}\mathrm{m}\right) \tag{6.32} $$

6.4.3 Operation grouping

For all the proposed schemes, the result of two discounting operations on sets Θ_1, Θ_2 and discount rate vectors $\vec{\alpha}_1$, $\vec{\alpha}_2$ done one after another is equal to a single discounting operation on combined discount rate vector $\vec{\alpha} = \text{concatenate}(\vec{\alpha_1}, \vec{\alpha_2})$.

$$ {}^{\vec{\alpha_1}}_{\Theta_1}\left({}^{\vec{\alpha_2}}_{\Theta_2}\mathrm{m}\right) = {}^{\vec{\alpha}}_{\Theta_1\cup\Theta_2}\mathrm{m} \qquad \text{if } \Theta_1 \cap \Theta_2 = \emptyset \tag{6.33} $$

This property can be easily generalised for any number of discounting operations.

$$ {}^{\vec{\alpha_K}}_{\Theta_K}\left(\ldots\left({}^{\vec{\alpha_1}}_{\Theta_1}\mathrm{m}\right)\ldots\right) = {}^{\vec{\alpha}}_{\Theta}\mathrm{m} \tag{6.34} $$

given that

$$ \Theta = \bigcup_{i\in\{1,\ \ldots,\ K\}} \Theta_i \tag{6.35} $$

$$ \vec{\alpha} = \text{concatenate}(\vec{\alpha}_1,\ \ldots,\ \vec{\alpha}_K) \tag{6.36} $$

and under the following condition:

$$ \forall i, j \in \{1,\ \ldots,\ K\},\ i \neq j\colon \Theta_i \cap \Theta_j = \emptyset \tag{6.37} $$

Table 6.2 – Discounted mass function for aerial target recognition example.

A	$\mathrm{m}(A)$	$\overset{\vec{\alpha}}{o}\mathrm{m}(A)$	$\overset{\vec{\alpha}}{p}\mathrm{m}(A)$	$\overset{\vec{\alpha}}{c}\mathrm{m}(A)$
$\emptyset$	0	0	0	0
$\{a\}$	0.5	0.5	0.5	0.5
$\{h\}$	0	0	0	0
$\{a, h\}$	0	0	0	0
$\{r\}$	0.5	0.5	0.3	0.3
$\{a, r\}$	0	0	0	0
$\{h, r\}$	0	0	0	0
Ω	0	0	0.2	0.2

6.5 Examples

6.5.1 Example 1: comparison of proposed discounting methods

Let $\Omega = \{\omega_1, \omega_2, \omega_3\}$ and let m be a bba defined on Ω. Table 6.1 presents the result which yield the proposed discounting schemes with $\Theta = \{\{\omega_1\}, \{\omega_2, \omega_3\}\}$ and discount rate vector $\vec{\alpha} = [\alpha_1, \alpha_{2,3}]$[1]. For clarity, we use $\beta_i = 1 - \alpha_i$. It is noteworthy that we can arrange the proposed discounting operations in incrementing order of total discounted mass: optimistic $\preceq$ proportional $\preceq$ conservative. For all mass functions and all discount rate vectors, the following equation holds:

$$\overset{\vec{\alpha}}{o}\mathrm{m}(A) \geq \overset{\vec{\alpha}}{p}\mathrm{m}(A) \geq \overset{\vec{\alpha}}{c}\mathrm{m}(A) \qquad \forall A \subsetneq \Omega \tag{6.38}$$

6.5.2 Example 2: source reliability modelling

Let us consider an example of a simplified aerial target recognition problem borrowed from (Elouedi, Mellouli, and Smets 2004; Mercier, Quost, and Denœux 2006). The frame of discernment $\Omega = \{a, h, r\}$ contains three classes: air-plane ($a = \omega_1$), helicopter ($h = \omega_2$) and rocket ($r = \omega_3$). Sensor S provides us with a bba m hesitating between classifying the target as an air-plane or a rocket:

$$\mathrm{m}(\{a\}) = 0.5 \qquad \mathrm{m}(\{r\}) = 0.5 \tag{6.39}$$

Let us now consider that the sensor is over-reliable when the source is a helicopter or a rocket with plausibility $\alpha_{2,3} = \alpha_{h,r} = 0.4$, while being reliable when the target is an air-plane. The conservatively discounted bba $\overset{\vec{\alpha}}{c}\mathrm{m}$ is:

$$\overset{\vec{\alpha}}{c}\mathrm{m}(\{a\}) = 0.5 \qquad \overset{\vec{\alpha}}{c}\mathrm{m}(\{r\}) = 0.3 \qquad \overset{\vec{\alpha}}{c}\mathrm{m}(\Omega) = 0.2 \tag{6.40}$$

It is to remark that a fraction 0.4 of the mass attributed to $\{r\}$ has been transferred to Ω, which can be interpreted as follows: if the target is a helicopter or a rocket, then the source is over-reliable and it might have quantified excessively its belief about target being a helicopter, a rocket or any of the two. Thus, the target reported as a rocket may in reality be of another type.

It is noteworthy that in order to obtain a similar result using contextual discounting by Mercier, Quost, and Denœux (2006), one has to apply the same discount factor to the complement set. That is, with

[1] The fact that Θ represents a partition of Ω is insignificant, since it could be any subset of 2^Ω.

$\alpha_1 = \alpha_a = 0.4$, cited from Mercier, Quost, and Denœux (2006, Example 2, Case 1), the discounting yields:

$$\overset{\vec{\alpha}}{\cup}\mathrm{m}(\{a\}) = 0.5 \qquad \overset{\vec{\alpha}}{\cup}\mathrm{m}(\{r\}) = 0.3 \qquad \overset{\vec{\alpha}}{\cup}\mathrm{m}(\{a,\ r\}) = 0.2 \tag{6.41}$$

This shows that the behaviour is almost inverse to conservative and proportional discounting and different than optimistic discounting. Namely, the discount factor being set to the same value but attributed to the set $\{a\}$, the resulting mass function is identical.

6.6 Case study: temporal discounting using proposed methods

In this section, an application of proposed discounted methods to *temporal discounting* is studied. The principal idea behind this discounting is the fact that a piece of information becomes partially obsolete with time. This can happen because the entity described by this particular information is dynamic, changes or is not observed any more. It is important to underline that different pieces of information become obsolete at possibly different rates. This example motivates why there is a need for introducing new contextual discounting schemes and why the existing one is not sufficient. The first part demonstrates some postulates about temporal discounting itself. Next, the existing contextual discounting scheme is applied to temporal discounting. Finally, the application of the proposed methods is demonstrated.

6.6.1 Temporal discounting using contextual discounting

This section will present an attempt to use contextual discounting as presented by Mercier, Quost, and Denœux (2005) and Mercier, Quost, and Denœux (2006) and a counter-example demonstrating that this discounting scheme is not adapted for this aim.

Computing discounting mass function Instead of calculating discounting mass function m_Θ by applying the disjunctive operator, one can compute it directly using (Mercier, Quost, and Denœux 2005, Proposition 7):

$$\mathrm{m}_\Theta(A) = \prod_{\substack{\theta\in\Theta\\ \theta\subseteq A}} \alpha_\theta \cdot \prod_{\substack{\theta\in\Theta\\ \theta\not\subseteq A}} (1-\alpha_\theta) \tag{6.42}$$

Computing discounted mass function Once again, direct computation is possible to obtain discounted mass function $\overset{\vec{\alpha}}{\cup,\Theta}\mathrm{m}$ using the results from Equations 6.13 and 6.42, which yields[1]:

$$\begin{aligned} \overset{\vec{\alpha}}{\cup,\Theta}\mathrm{m}(A) &= (\mathrm{m} \textcircled{\cup} \mathrm{m}_\Theta)(A) \\ &= \sum_{B\cup C=A} \mathrm{m}(B)\cdot \mathrm{m}_\Theta(C) \\ &= \sum_{B\subseteq A}\left[\mathrm{m}(B)\cdot \sum_{C\subseteq A,\ C\supseteq A\setminus B} \mathrm{m}_\Theta(C)\right] \end{aligned} \tag{6.43}$$

[1] It is supposed that no discount rate has been defined for the empty set.

Simplified computation Let us suppose that m is a normal mass function, i.e. the mass attributed to the empty set is null. This enables us to simplify Equation 6.43 for singletons to:

$$\begin{aligned}
{}^{\vec{\alpha}}_{\cup,\Theta}\mathrm{m}(\{\theta\}) &= \sum_{B\cup C=\theta} \mathrm{m}(B)\cdot \mathrm{m}_\Theta(C) \\
&= \sum_{C\subseteq\theta} \mathrm{m}(\{\theta\})\cdot \mathrm{m}_\Theta(C) \\
&= \mathrm{m}(\{\theta\})\cdot \sum_{C\subseteq\theta} \mathrm{m}_\Theta(C) \\
&= \mathrm{m}(\{\theta\})\cdot \mathrm{bel}_\Theta(\{\theta\})
\end{aligned} \tag{6.44}$$

Use for temporal discounting

In order to calculate discount rates $\vec{\alpha}$ of contextual discounting from parameters $\vec{\kappa}$ of temporal discounting, let us compare side by side temporal discounting (Equation 6.16) as obtained thanks to the above stated postulates:

$$\begin{aligned}
{}^{\vec{\alpha}}\mathrm{m}(\{\theta\}) &= \mathrm{m}(\{\theta\})\cdot e^{-\lambda_\theta t} \\
&= \mathrm{m}(\{\theta\})\cdot \kappa_\theta \qquad\qquad \forall\theta\in\Theta,\ 0<\kappa_\theta\leq 1
\end{aligned} \tag{6.45}$$

with the simplified expression of contextually discounted mass (Equation 6.44):

$${}^{\alpha}\mathrm{m}(\{\theta\}) = \mathrm{m}(\{\theta\})\cdot \mathrm{bel}_\Theta(\theta) \tag{6.46}$$

which, given that $\mathrm{m}(\{\theta\})\neq 0,\ \forall\theta\in\Theta$, yields:

$$\mathrm{m}(\{\theta\})\cdot\kappa_\theta \equiv \mathrm{m}(\{\theta\})\cdot \mathrm{bel}_\Theta(\theta) \qquad /:\ \mathrm{m}(\{\theta\}) \tag{6.47}$$

$$\kappa_\theta \equiv \mathrm{bel}_\Theta(\theta) \tag{6.48}$$

$$\kappa_\theta \equiv \prod_{\substack{B\in\Theta\\ B\not\subseteq\theta}} (1-\alpha_B) \tag{6.49}$$

Let $K=|\Omega|=|\Theta|$. Creating a system of equations for all $\theta\in\Theta$ using Equation 6.49 issues:

$$\begin{cases}
\kappa_{\theta_1} &= \prod\limits_{\substack{B\in\Theta\\ B\not\subseteq\theta_1}} (1-\alpha_B) \\
&\vdots \\
\kappa_{\theta_K} &= \prod\limits_{\substack{B\in\Theta\\ B\not\subseteq\theta_K}} (1-\alpha_B)
\end{cases} \tag{6.50}$$

By solving the above equation it, with the convention that $\prod\limits_{i\in\emptyset} x_i = 1$, one obtains:

$$\alpha_i = 1 - \sqrt[K-1]{\frac{\prod\limits_{j\neq i}\kappa_{\theta_j}}{\kappa_{\theta_i}^{K-2}}} \tag{6.51}$$

From Equations 6.45 and 6.51, we obtain:

$$\kappa_\theta(t) = e^{-\lambda_\theta t} \tag{6.52}$$

$$\alpha_i(t) = 1 - \sqrt[K-1]{\frac{\prod_{j \neq i} e^{-\lambda_j t}}{(e^{-\lambda_i t})^{K-2}}} \tag{6.53}$$

Example and counterexample

Let us consider source S that provides mass functions m_S^Ω defined on the frame of discernment $\Omega = \{\omega_1,\ \omega_2,\ \omega_3\}$. Relying on the information that we possess about this source, we can examine two cases C_1 and C_2 for which the half-life times differ. For each $\omega \in \Omega$, half-life times $t_{1/2}$ are known to be:

$$\begin{array}{cc} C_1 & C_2 \\ \vec{t}_{1/2,\,C_1} = [1, 4, 15]\ \text{s} & \vec{t}_{1/2,\,C_2} = [5, 4, 15]\ \text{s} \end{array}$$

Case C_1 can be interpreted as follows. Additional knowledge about source S is available and it states that classes ω_1, ω_2 and ω_3 become obsolete with different rates. Namely, ω_1 is known to be worth a half of its initial value[1] after 1 second, ω_2 and ω_3 — after 4 s and 15 s respectively. Analogical interpretation should be given to case C_2 with the sole difference that the half-life period of class ω_1 is longer and equal to 5 s.

Using Equation 6.3, decay parameters $\vec{\lambda}$ are computed:

$$\vec{\lambda}_{C_1} \approx [0.6931, 0.1733, 0.0462] \tag{6.54}$$

$$\vec{\lambda}_{C_2} \approx [0.1386, 0.1733, 0.0462] \tag{6.55}$$

Then, thanks to Equations 6.52 and 6.53, let compute parameters $\vec{\kappa}$ and discount factor vector $\vec{\alpha}$ for instant $t = 4$ s:

$$\vec{\kappa}_{C_1}(t) \approx [0.0625, 0.5000, 0.8312] \tag{6.56}$$

$$\vec{\kappa}_{C_2}(t) \approx [0.5743, 0.5000, 0.8312] \tag{6.57}$$

$$\vec{\alpha}_{C_1} \approx [-1.5787, 0.6777, 0.8061] \tag{6.58}$$

$$\vec{\alpha}_{C_2} \approx [0.1493, 0.0228, 0.4122] \tag{6.59}$$

Discounting for case C_1 The above steps demonstrate that the desired temporal discounting cannot be expressed in terms of contextual discounting as proposed in (Mercier, Quost, and Denœux 2005). Indeed, $\vec{\alpha}_{C_1}$ contains a negative value, which is incompatible with this method and the outcome of such a discounting would not satisfy the condition of a mass function as required in Equation 3.17.

[1] The word *value* corresponds to some subjective value of a piece of information from the point of view of the fusion system.

Discounting for case C_2

For case C_2, we can compute discounting mass function m_Θ using Equation 6.42:

$$\begin{aligned} \mathrm{m}_\Theta(\emptyset) &= 0.4886 & \mathrm{m}_\Theta(\{\omega_1\}) &= 0.0858 \\ \mathrm{m}_\Theta(\{\omega_2\}) &= 0.0114 & \mathrm{m}_\Theta(\{\omega_1,\ \omega_2\}) &= 0.0020 \\ \mathrm{m}_\Theta(\{\omega_3\}) &= 0.3427 & \mathrm{m}_\Theta(\{\omega_1,\ \omega_3\}) &= 0.0602 \\ \mathrm{m}_\Theta(\{\omega_2,\ \omega_3\}) &= 0.0080 & \mathrm{m}_\Theta(\{\omega_1,\ \omega_2,\ \omega_3\}) &= 0.0014 \end{aligned}$$

Given function m with masses attributed as follows:

$$\begin{aligned} \mathrm{m}(\emptyset) &= 0 & \mathrm{m}(\{\omega_1\}) &= 0.3 \\ \mathrm{m}(\{\omega_2\}) &= 0.4 & \mathrm{m}(\{\omega_1,\ \omega_2\}) &= 0 \\ \mathrm{m}(\{\omega_3\}) &= 0.2 & \mathrm{m}(\{\omega_1,\ \omega_3\}) &= 0 \\ \mathrm{m}(\{\omega_2,\ \omega_3\}) &= 0 & \mathrm{m}(\{\omega_1,\ \omega_2,\ \omega_3\}) &= 0.1 \end{aligned}$$

using the contextual discounting operation as expressed by Equation 6.13, we obtain discounted mass function ${}^t\mathrm{m} = \overset{\vec{\alpha}}{\cup}\mathrm{m}$:

$$\begin{aligned} {}^t\mathrm{m}(\emptyset) &= 0 & {}^t\mathrm{m}(\{\omega_1\}) &= 0.1723 \\ {}^t\mathrm{m}(\{\omega_2\}) &= 0.2 & {}^t\mathrm{m}(\{\omega_1,\ \omega_2\}) &= 0.0391 \\ {}^t\mathrm{m}(\{\omega_3\}) &= 0.1662 & {}^t\mathrm{m}(\{\omega_1,\ \omega_3\}) &= 0.15 \\ {}^t\mathrm{m}(\{\omega_2,\ \omega_3\}) &= 0.1442 & {}^t\mathrm{m}(\{\omega_1,\ \omega_2,\ \omega_3\}) &= 0.1281 \end{aligned}$$

6.6.2 Contextual temporal discounting inconveniences

Contextual temporal discounting such as presented presents some undesirable properties. Let us reuse the example of case C_2 from the section 6.6.1. Let $t = 60\,\mathrm{s}$. We would expect that the masses attributed to all classes will diminish 4096, 32 768 and 16 times respectively[1]. The resulting discounted mass function is equal to:

$$\begin{aligned} {}^t\mathrm{m}(\emptyset) &= 0 && (6.60)\\ {}^t\mathrm{m}(\{\omega_1\}) &= 0.0000732 && (6.61)\\ {}^t\mathrm{m}(\{\omega_2\}) &= 0.0000122 && (6.62)\\ {}^t\mathrm{m}(\{\omega_1,\ \omega_2\}) &= 0.000156 && (6.63)\\ {}^t\mathrm{m}(\{\omega_3\}) & && (6.64)\\ {}^t\mathrm{m}(\{\omega_1,\ \omega_3\}) &= 0.3410 && (6.65)\\ {}^t\mathrm{m}(\{\omega_2,\ \omega_3\}) &= 0.0405 && (6.66)\\ {}^t\mathrm{m}(\{\omega_1,\ \omega_2,\ \omega_3\}) &= 0.6058 && (6.67) \end{aligned}$$

As expected, we observe that ${}^t\mathrm{m}(\{\omega_1\}) = 7.32 \times 10^{-5} \approx \frac{\mathrm{m}(\{\omega_1\})}{4096}$. Similarly, our expectations are satisfied for ${}^t\mathrm{m}(\{\omega_2\})$ and ${}^t\mathrm{m}(\{\omega_3\})$. However, it is observed that e.g. ${}^t\mathrm{m}(\{\omega_1,\ \omega_2\})$ has a relatively

[1] These numbers come from a simple computation $2^{\frac{t}{t_{1/2}}}$.

Table 6.3 – Temporal discounting using the proposed discount schemes. Case 1.
$\vec{t}_{1/2,C_1} = [1, 4, 15]$ s

A	$\mathrm{m}(A)$	$\overset{\vec{\alpha}}{o}\mathrm{m}(A)$	$\overset{\vec{\alpha}}{p}\mathrm{m}(A)$	$\overset{\vec{\alpha}}{c}\mathrm{m}(A)$
$\emptyset$	0	0	0	0
$\{\omega_1\}$	0.3	0.281	0.281	0.281
$\{\omega_2\}$	0.2	0.1	0.1	0.1
$\{\omega_1, \omega_2\}$	0.2	0.2	0.1453125	0.09375
$\{\omega_3\}$	0.2	0.034	0.034	0.034
$\{\omega_1, \omega_3\}$	0	0	0	0
$\{\omega_2, \omega_3\}$	0	0	0	0
Ω	0.1	0.385	0.440	0.491

Table 6.4 – Temporal discounting using the proposed discount schemes. Case 2. Result of Mercier's contextual discounting in the rightmost column.
$\vec{t}_{1/2,C_2} = [5, 4, 15]$ s

A	$\mathrm{m}(A)$	$\overset{\vec{\alpha}}{o}\mathrm{m}(A)$	$\overset{\vec{\alpha}}{p}\mathrm{m}(A)$	$\overset{\vec{\alpha}}{c}\mathrm{m}(A)$	$\overset{\vec{\alpha}}{\cup}\mathrm{m}(A)$
$\emptyset$	0	0	0	0	0
$\{\omega_1\}$	0.3	0.12771	0.12771	0.12771	0.1723
$\{\omega_2\}$	0.2	0.1	0.1	0.1	0.1
$\{\omega_1, \omega_2\}$	0.2	0.2	0.1069275	0.04257	0.1391
$\{\omega_3\}$	0.2	0.03376	0.03376	0.03376	0.1662
$\{\omega_1, \omega_3\}$	0	0	0	0	0.15
$\{\omega_2, \omega_3\}$	0	0	0	0	0.074
Ω	0.1	0.53853	0.6316025	0.69596	0.1983

large value, only slightly smaller than $\frac{\mathrm{m}(\{\omega_1\})+\mathrm{m}(\{\omega_2\})+{}^{t}\mathrm{m}(\{\omega_1,\,\omega_2\})}{4096} = \frac{0.7}{4096} = 1.71 \times 10^{-4}$, but bigger than $\frac{0.7}{32768} = 2.14 \times 10^{-5}$.

6.6.3 Temporal discounting using proposed discounting schemes

On the contrary to contextual discounting, the proposed methods are expressive enough to reflect the desired behaviour of temporal discounting. Let us reuse the same two cases evoked in Section 6.6.1. The computation of decay parameters $\vec{\lambda}$ and $\vec{\kappa}$ is common to both methods. Moreover, discount rate vector values $\vec{\alpha}$ correspond directly to values of $\vec{\kappa}$ as shown by:

$$\vec{\alpha}_1 = [\omega_1 \mapsto 0.0625, \omega_2 \mapsto 0.5, \omega_3 \mapsto 0.8312] \tag{6.68}$$

$$\vec{\alpha}_2 = [\omega_1 \mapsto 0.5743, \omega_2 \mapsto 0.5, \omega_3 \mapsto 0.8312] \tag{6.69}$$

Tables 6.3 and 6.4 show 3 different discounting methods for the two analysed cases.

6.7 Conclusion

In this chapter, we have proposed and defined three types of contextual discounting: conservative, proportional and optimistic. These methods allow fine-grained modelling of the reliability of the sources. Moreover, the introduced techniques can be applied to temporal discounting which has been described as well. It has been demonstrated that the existing contextual discounting introduced by Mercier, Quost, and Denœux (2006) is not strong enough to model temporal discounting.

In addition to the already given applications, the authors consider the use of temporal discounting in

the context of intelligent transportation perception. Various object classes seen by a vehicle should not be forgotten at the same rate. For instance, information about objects recognised as buildings shall be kept longer than static but possibly mobile objects. In turn, mobile static objects would persist longer than moving objects.

As a practical advantage, one can mention that for a given discount rate vector, factors by which masses are multiplied to obtain discounted mass function can be precomputed and stored for later use. The computational complexity of such an algorithm grows linearly with the size of the power set 2^{Ω} equally for time and space.

It would be interesting to automatically or semi-automatically define which type of discounting has to be used in particular situation. Moreover, a profound study of the properties of the proposed discounting rules seems to be significantly important. These tasks are left for future research.

[This page intentionally left blank.]

Part IV
Experimental results

[This page intentionally left blank.]

Chapter 7

System & setup

"Z doświadczenia rozum się mnoży."
"Experience multiplies the reason."

Mikołaj Rej

Contents

7.1 Map-aided perception system architecture

The high complexity of a perception system deserves a profound consideration on the structure of the system. There are few works on the architecture of systems using grid-based approach, but some insightful studies of similar topics have been realised by Durekovic and Smith (2011) and by Benenson and Parent (2008b). Among many foreseeable architectures, the chosen one is considered to be the simple but effective enough for our needs. This component-based software architecture is roughly depicted by Figure 7.1.

Figure 5.1 presents in turn the general logics of our approach. The actual system, with the real sensors that were installed on the vehicle, is illustrated by Figure 7.2. Starting on the leftmost side, the diagram shows the data sources. The method employs different evidential grids for the storage of: prior information, sensor acquisitions and fusion result.

Necessary data

All system inputs mentioned on Figure 5.1 are obligatory in the proposed perception system:

- at least one exteroceptive sensor,
- localisation system (GNSS receiver, proprioceptive sensor, IMU),

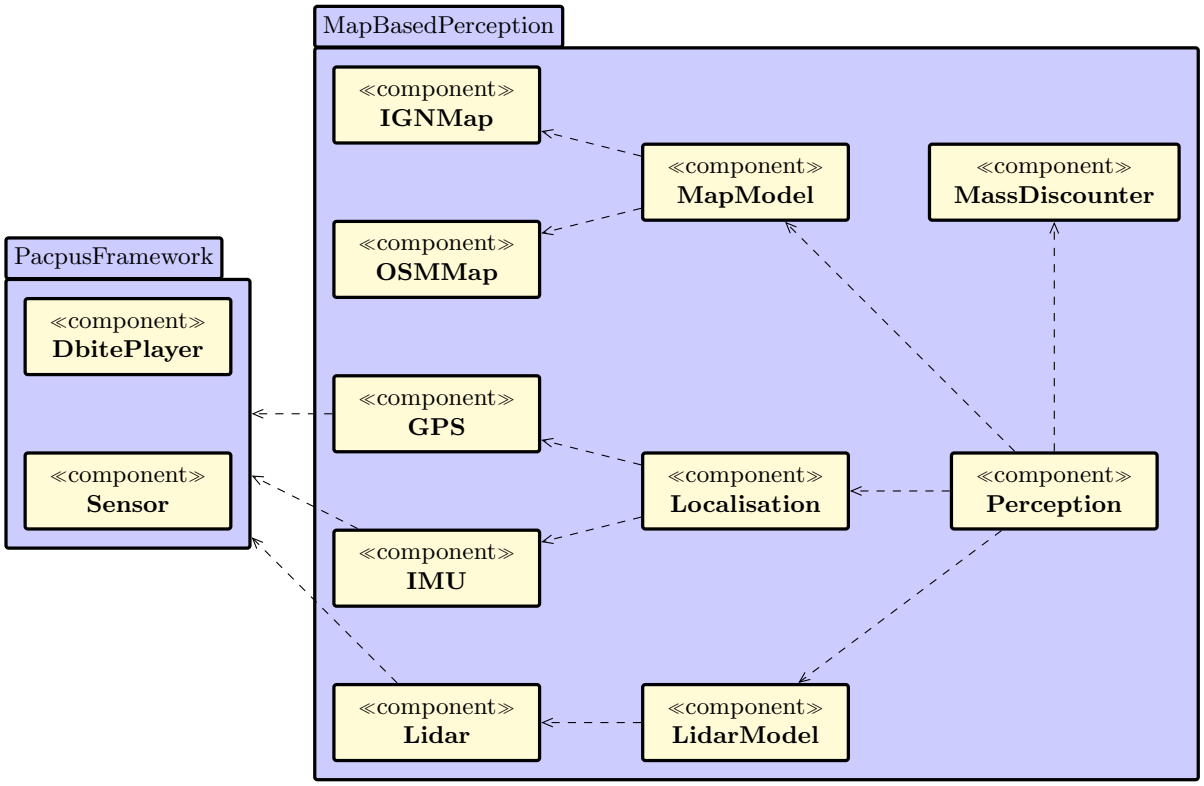


Figure 7.1 – Simplified view of components of Pacpus framework used for the implementation of the perception system.

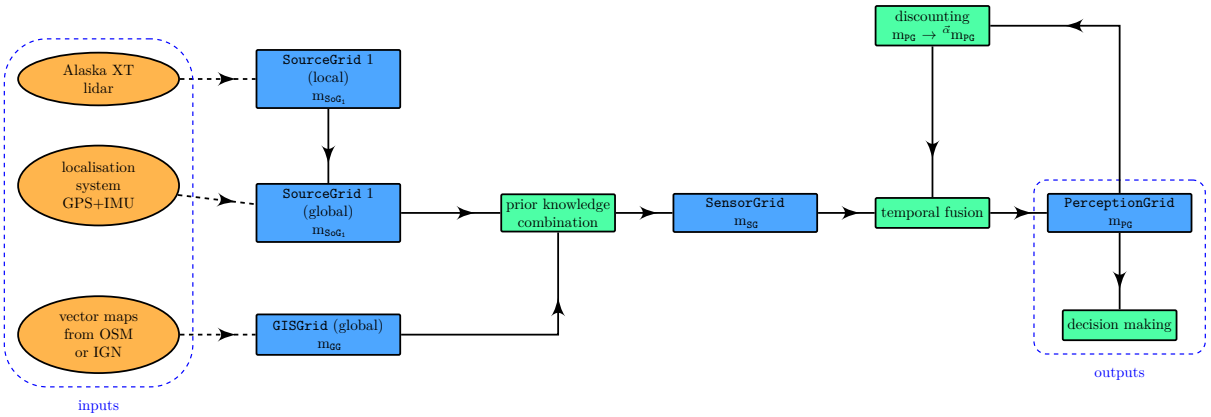


Figure 7.2 – Overview of the perception system as used on the Pacpus platform.

- vector map containing geometric information (roads, buildings).

The presented approach is based on the hypothesis that all information sources are available. If multiple sensors are at hand, the method is adapted to use them and takes advantage of this supplementary information.

An exteroceptive sensor gives a partial view of the vehicle environment. The sensor is assumed to distinguish free and occupied space and model it in 2D x, y or 3D x, y, z coordinates. The coordinates can be relative to the robot or world-referenced. A typical exteroceptive sensor capable of satisfying this assumption is a lidar (laser range scanner), a radar, or a stereo camera system.

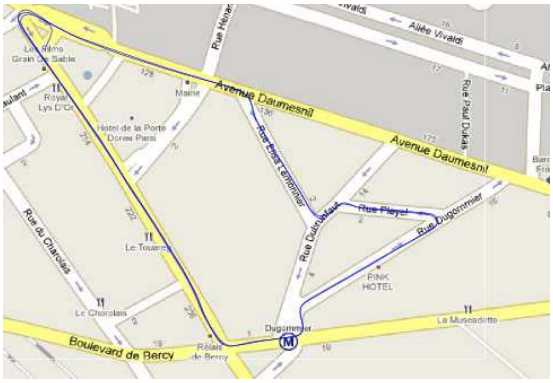
A proprioceptive sensor like an Inertial Measurement Unit (IMU) or an odometer hybridised with a Global Navigation Satellite System (GNSS) receiver are needed to provide the vehicle pose. We have used a highly-precise hybrid IMU-Global Positioning System (GPS) Applanix system for localisation Provided pose is assumed to be reliable, accurate and precise. The pose should be globally referenced and is needed to situate the vehicle in the environment. A hypothesis is made that the pose reflects accurately the real state of the vehicle.

Lastly, our method tries to exploit at large the information contained in geographical maps, so we assume that the maps are sufficiently detailed and contain valuable and accurate data. At the minimum, the map data have to contain buildings and road surface description. For this purpose, we have performed tests on two different maps. The first one has been prepared by the National Institute of the Geographic and Forest Information (IGN) and was supposed to be almost flawless. This fact is reflect by a very high map confidence level β attributed to this map. It contained precise 3-dimensional (3D) building models and 3D road surface model. The second map that we used was the freely available map from OpenStreetMap (OSM) project. This source contained only 2-dimensional (2D) building models and the road network. The roads were represented only by polylines supposed to be situated in the middle of the way. In order to define the road surface, we used the width of the road if this parameter was available; otherwise we made a hypothesis that its width depends on the type of the road. The confidence level β was lower than for the IGN map, since this map could have been edited by anybody and no verification has been performed over it.

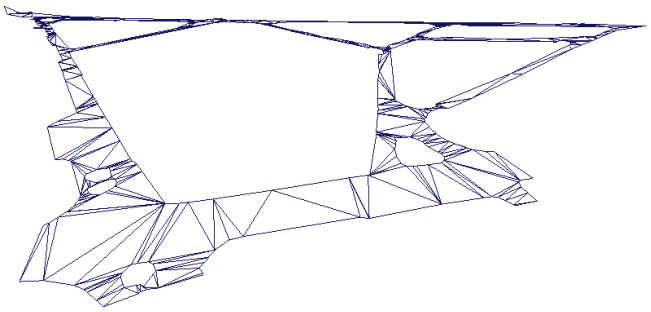
7.2 Dataset

The data used in this research were acquired thanks to the collaboration with the IGN during the CityVIP project (Soheilian, Tournaire, et al. 2013). Experiments took place in the 12th district of Paris and the overall length of the test trajectory was approximately 3 km. Other experiments have been performed using the PACPUS platform (Heudiasyc 2013) of the Heudiasyc laboratory and Carmen vehicle (shown in Figure 7.4). To sum up, the Carmen platform have used the following sensors. Applanix localisation module based on a GPS, an odometer and an IMU provided one of the system entries, namely the vehicle pose. The pose given by Applanix is supposed precise and of high confidence. As the exteroceptive sensor, an IBEO Alaska XT lidar was used. It provided a point cloud of 800 impacts at a frequency of 10Hz.

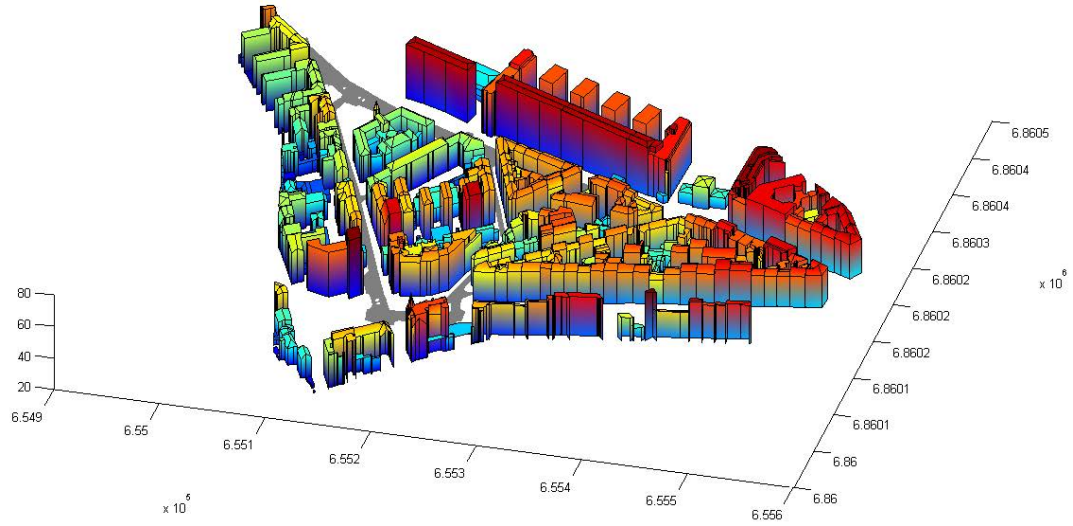
The vector maps were provided by the IGN and contain 3D models of road surfaces and buildings (see Figure 7.3). Tests were also performed with free editable 2D maps from the *OpenStreetMap* project (OSM 2013). The maps were supposed accurate and up-to-date.



(a) Vehicle trajectory.



(b) Road surface polygons.



(c) Buildings.

Figure 7.3 – Visual representation of the dataset.

Figure 7.4 – Test vehicle Carmen with the lidar sensor in front.

7.2.1 Using OpenStreetMap data

Our algorithm for creating `GISGrid` (GG) needs 2D surface model of the road. Since the roads in OSM are represented by multi-lines and not by polygons, we were obliged to transform the former into the latter. This is somehow problematic, because not all roads contain information about their width, their form might be complex or the width might change. We have therefore approximated the width of each road segment. This estimation was based on the segment type tag, which is present for almost all highways. For each class, a predefined value for width has been used to transform a multi-line into a polygon. Such an approach is almost identical to the one presented by Mandel and Laue (2010, Table I).

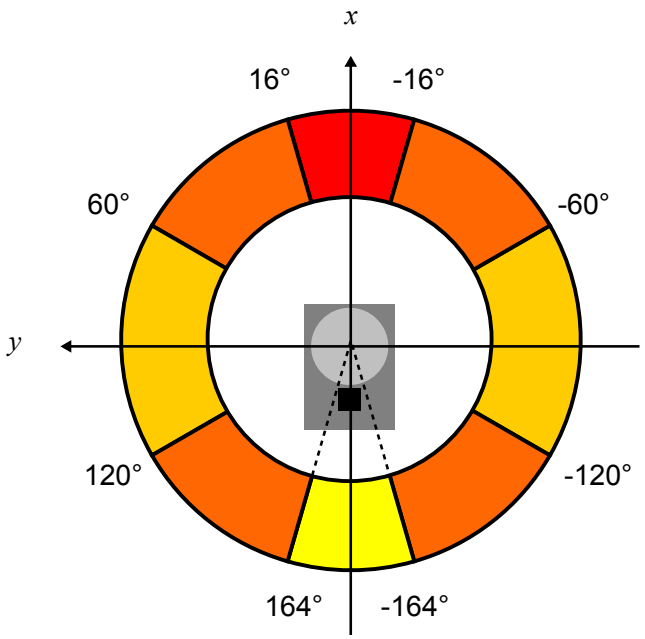
7.2.2 Using IGN data

We have as well used data furnished by the National Institute of the Geographic and Forest Information (IGN) composed of 3D models of buildings. The map data contain as well a road surface model also in 3 dimensions. All the preprocessing needed to make use of this data set was to transform 3D map into a two-dimensional one. This was achieved simply by applying a projection onto a locally tangent plane perpendicular to the Z-axis (down–up).

7.2.3 Lidar scan

It is a four-scan sensor which provides a 3D point cloud of the environment at each scan. This sensor can do measurements up to 200 m in a front field of 320°, with a rate from 8 Hz up to 40 Hz depending on the angular resolution. The lidar uses a 905 nm wavelength infra-red laser which has an aperture of 0.25°.

The angular resolution of the Alaska XT sensor is adaptive according to the angle as shown in Figure 7.5. This sensor is also able to provide several echoes per line of sight if, for instance, the laser beam is partially reflected by an obstacle. Another characteristic is that it can return no measurement in the considered line of sight. If there is no echo, two cases are possible: there is no object until the maximum range or there is an obstacle which does not reflect the laser beam. The sensor model that we used takes into account these particularities.



Resolution	
12.5 Hz	25 Hz
0.125°	0.25°
0.25°	0.5°
0.5°	1°
1°	2°

Figure 7.5 – Alaska XT angular resolution in function of the angle of measurement and of the frequency.

Table 7.1 – Parameter values used in the experimental setup.

Parameter name	Value
OSM map confidence level β	0.97
IGN map confidence level β	0.99
Cell size Δ	0.50 m
Grid size	80.00 mx80.00 m
Accumulator gain δ	0.02
Decrement-to-increment ratio γ	6.00
Free space confidence μ_F	0.70
Occupied space confidence μ_O	0.80
Dynamic classes discount factor $\alpha_{\{D,N,M\}}$	0.01
Static classes discount factor $\alpha_{\{S,I,U\}}$	0.10

7.3 Defining grid parameters

Table 7.1 lists all the parameter values used for the construction of sensor models (see Chapter 4) and the fusion operation (see Chapter 5).

Cell size The size of the grid cell in the occupancy grids was set to 0.5 m, which is sufficient to model a complex environment with mobile objects. Before choosing this value, we have tested our approach on a range of different cell sizes. They varied between 0.1 m and 1.0 m.

Lidar confidence level The confidence level for a lidar sensor model corresponding to free space μ_F and to occupied space μ_O have been defined empirically by testing the sensor in use.

Map confidence level We have specified the map confidence factor β by ourselves and set it to 0.98, but ideally, it should be given by the map provider. β describes data currentness (age), errors introduced

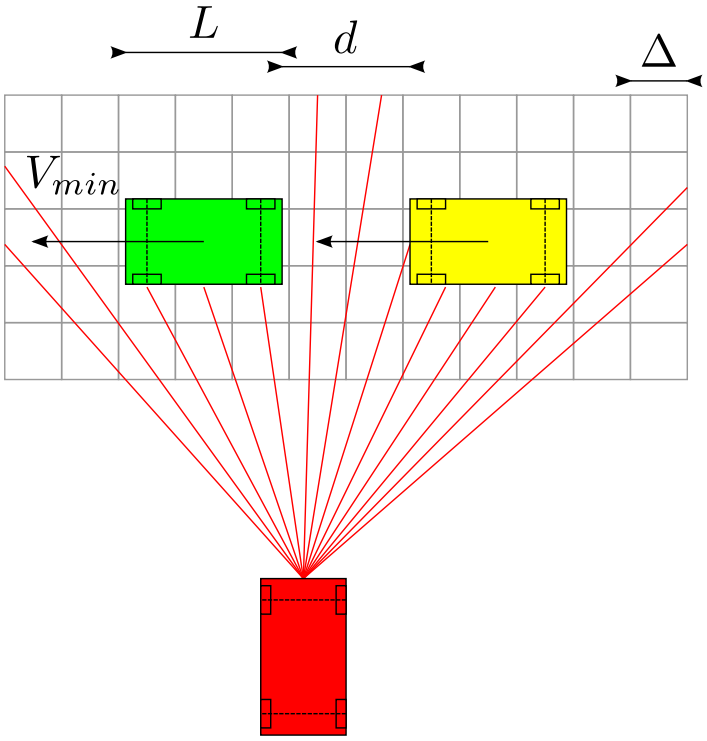


Figure 7.6 – Illustration of geometrical distances used for the parameter computation.

by geometry simplification and spatial discretisation. β can also be used to depict the localisation accuracy.

Remanence characteristics and discount factors The discount factors describe the persistence of the information of a given class, or in other words, the speed at which the information becomes obsolete. The higher is the discount factor, the lower the persistence of the corresponding object type. The discount factors can be learnt if a reference data with annotated objects is available. In our case, the discount rates $\vec{\alpha}$ were defined empirically. A learning method to obtained these factors has been proposed in (Mercier, Quost, and Denœux 2006).

Specialisation gains

Parameters δ (accumulator gain) and γ (decrement-to-increment ratio) used for static object detection determine the sensitiveness of this perception process.

To compute the values δ and γ, let consider an object of a length L moving at a speed V_{min} which is the minimum speed one would be able to detect. This configuration is illustrated by Figure 7.6.

The exteroceptive sensor provides scans with a constant frequency f. One can compute the maximum number of sensor cycles i_{point} necessary for a point of an object to pass through a cell of size Δ as follows:

$$i_{\text{point}} = \frac{\Delta \cdot f}{V_{\text{min}}} \tag{7.1}$$

The object of length L occupies at least $\frac{L}{\Delta}$ cells, so, analogically to Equation 7.1, we can compute the

number of cycles i_{object} during which this object occupies one cell:

$$i_{\text{object}} = \frac{L \cdot f}{V_{\text{min}}} \tag{7.2}$$

δ is computed in such a manner that the accumulator ζ grows up to 1 in i_{object} cycles.

$$\delta \cdot i_{\text{object}} = 1 \tag{7.3}$$

And then, from Equations 7.2 and 7.3:

$$\delta = \frac{V_{\text{min}}}{L \cdot f} \tag{7.4}$$

For a typical moving car at a minimal speed $V_{\text{min}} = 1\,\frac{\text{m}}{\text{s}}$ with a length $L = 3\,\text{m}$ and the sensor scan frequency $f = 15\,\text{Hz}$, one obtains $\delta = 0.02$.

Coefficient γ is determined in a similar way to δ, but in this case we consider the inter-object distance d between two successive objects. The fact that the accumulator ζ should reach 0 before the following object enters into this cell implies:

$$\delta \cdot \gamma = \frac{V_{\text{min}}}{d \cdot f} \tag{7.5}$$

By applying Equation 7.4 and simplifying, we obtain then:

$$\gamma = \frac{L}{d} \tag{7.6}$$

Finally, setting an inter-obstacle distance to $d = 0.5\,\text{m}$, one computes decrement-to-increment ratio $\gamma = 6$.

Chapter 8

Results

"Verba docent exempla trahunt."
"Words teach, examples attract."

Latin proverb

Contents

8.1 Contribution of map data

To assess the performance of our method, two cases have been considered. Firstly, when maps are present and prior information can be exploited. Secondly, when no maps are available and only mobile and static detection is done. A comparison of perception results for these two cases has then been shown. In this way, we show the interest of using a map-aided approach to the perception problem. All tests have been performed on real-world data recorded in an urban environment in Paris, France. The map that we used for this experiments came from National Institute of the Geographic and Forest Information (IGN).

The implemented perception system did not permit us to perform tests in real-time unfortunately. This objective is however attainable and could be achieved using techniques of parallel computing; this rests one of the perspectives for future work. The high computational complexity lies in the fact that the cardinality of frame of discernment Ω_{PG} in use was quite significant. It is to note that the information fusion using belief functions theory requires to operate on $2^{|\Omega_{PG}|}$ possible subsets of the frame of discernment. Evidential grids have therefore high computational requirements and important memory needs in comparison with their probabilistic counterparts. To be precise, an evidential grid is up to $2^{|\Omega_{PG}|}$ bigger than a probabilistic one. For this reason, and since the code has not been optimised, the tested program does not work in real time. A new lidar scan arrives at frequency of 10 Hz, but the processing of a single scan and all the fusion process takes approximatively 300 ms.

The perception results for a particular instant of the tested approach are presented on Figure 8.1. The visualisation of the `PerceptionGrid` has been obtained by attributing a colour to each class with

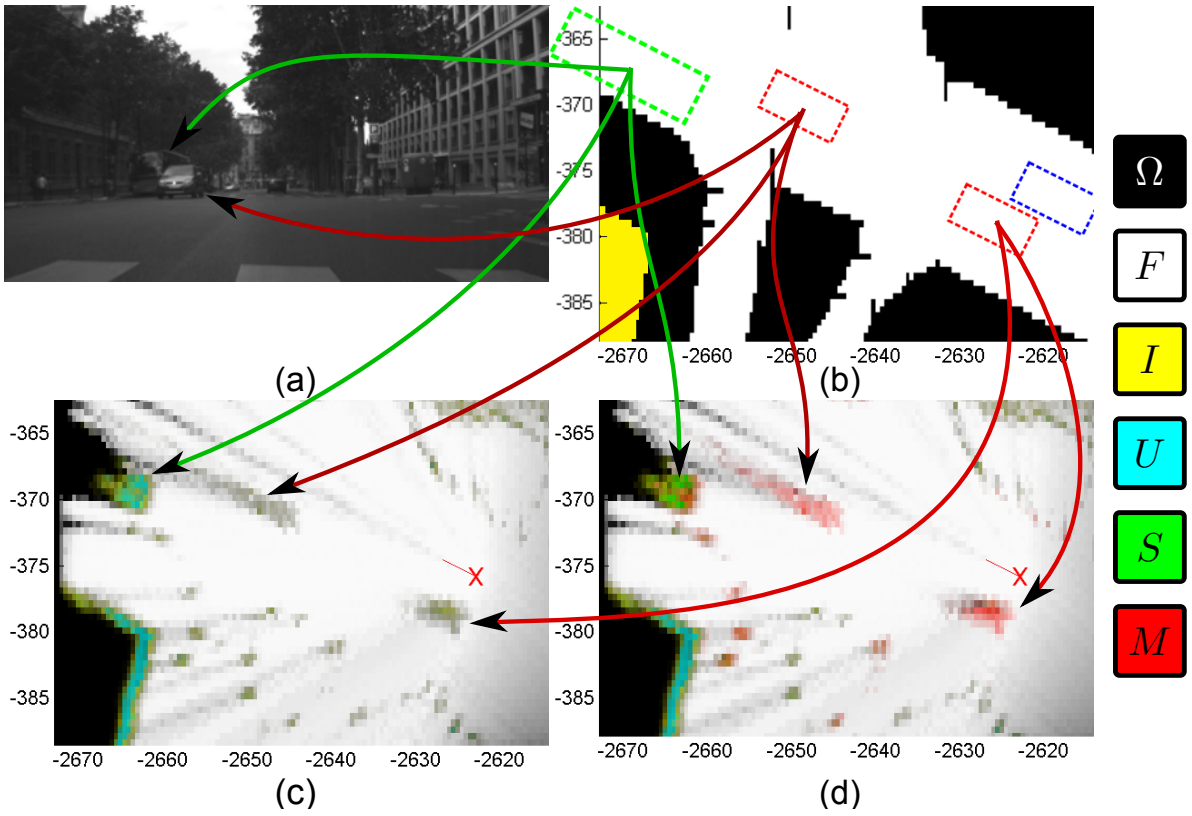


Figure 8.1 – (a) Scene. (b) `GISGrid` with superposed vehicle outlines. (c) `PerceptionGrid` without prior information. (d) `PerceptionGrid` with prior map knowledge. Right: colour code.

proportional to the mass value and calculating the mean colour. The presented scene contains two cars, one visible in the camera image and one invisible, both going in the direction opposite to the test vehicle, and a bus parked on the road edge. Bus and car positions are marked on the grids by green and red boxes, respectively. The position of the test vehicle is shown as a blue box. Classes of Ω_{PG} are represented by different colours as described at the right side of Figure 8.1. `GISGrid` in Figure 8.1b visualises the prior knowledge obtained from maps by showing the position of the road surface, in white, and buildings, in blue.

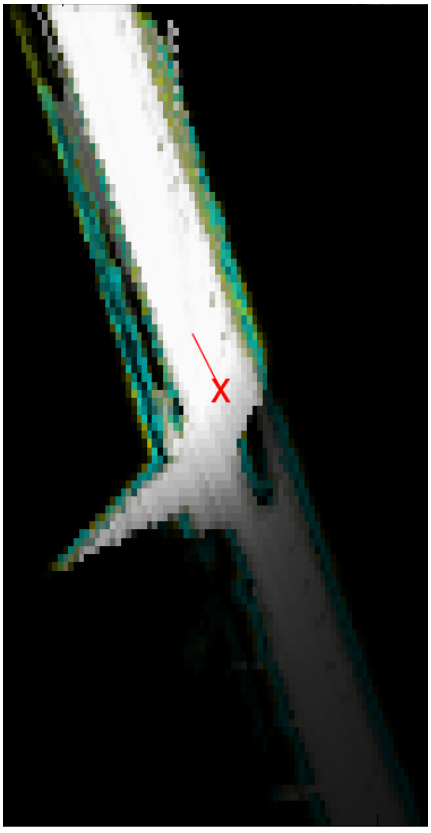
The advantage of using map knowledge is richer information on the detected objects. A difference between moving cells (red, car) and stopped ones (green, bus) is clearly visible. Also, stopped objects are distinct from infrastructure when prior map information is available (cf. Figures 8.1c and 8.1d). In addition, thanks to the prior knowledge, stationary cells, in cyan, modelling infrastructure are distinguished from stopped cells (road objects).

Figure 8.1 shows the effect of temporal discounting, which is particularly visible on the free space behind the vehicle. On the other hand, the parked bus is still in evidence despite being occluded by the passing car. Masses attributed to grid cells are being discounted, so the mass on the free class F diminishes gradually.

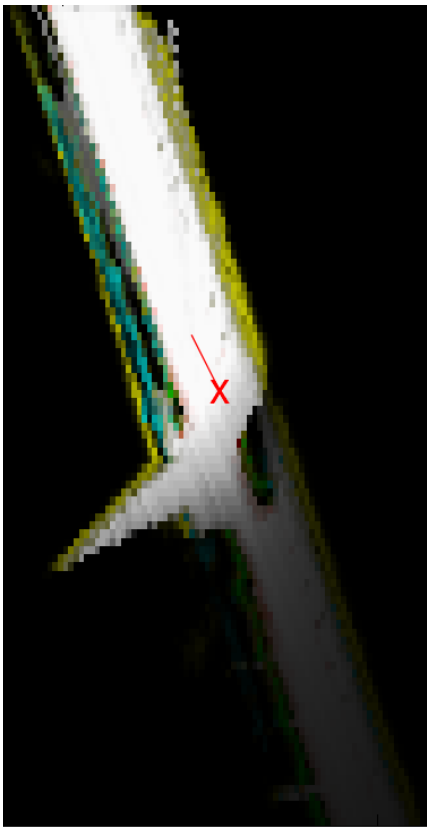
In Figure 8.2, there is a clear difference between the perception of buildings when the map data is available and when it is not. When no map is present, buildings are confused with barriers and bicycles. With prior information from the maps, the proposed approach easily distinguishes between the two.

Following paragraphs use the pignistic probability, described by Equation 3.40, as a method of transforming masses into probabilities. This mechanism is used in order to show an example of a decision rule that can be executed on an evidential grid.

(a) Camera acquisition



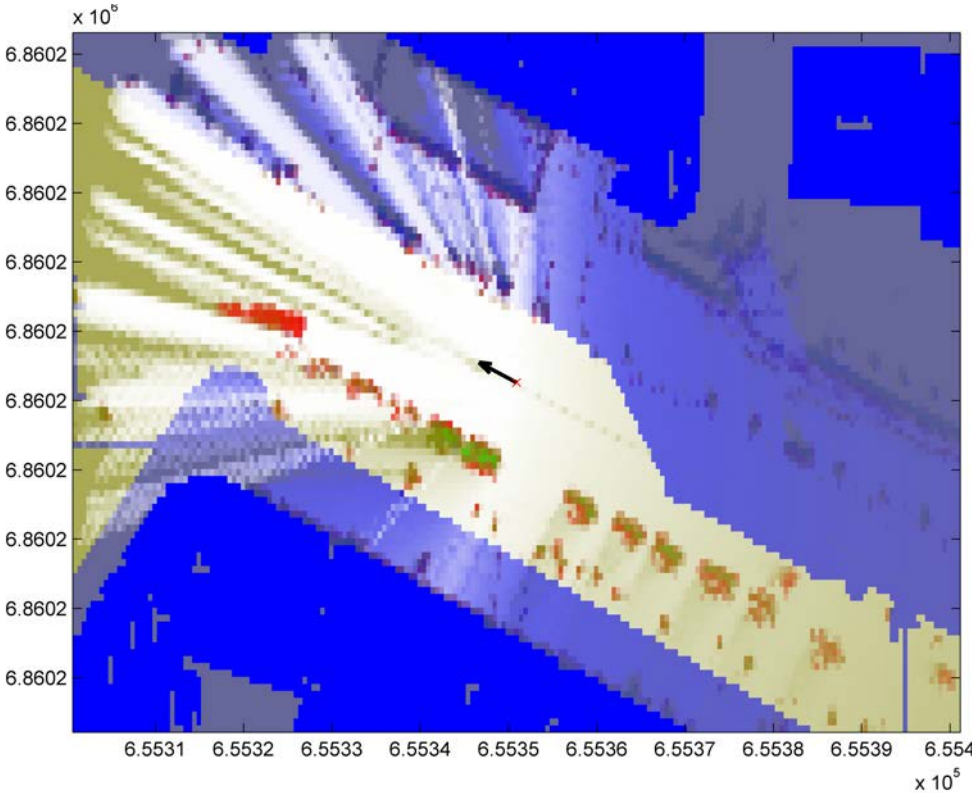
(b) `PerceptionGrid` obtained without map



(c) `PerceptionGrid` obtained with map

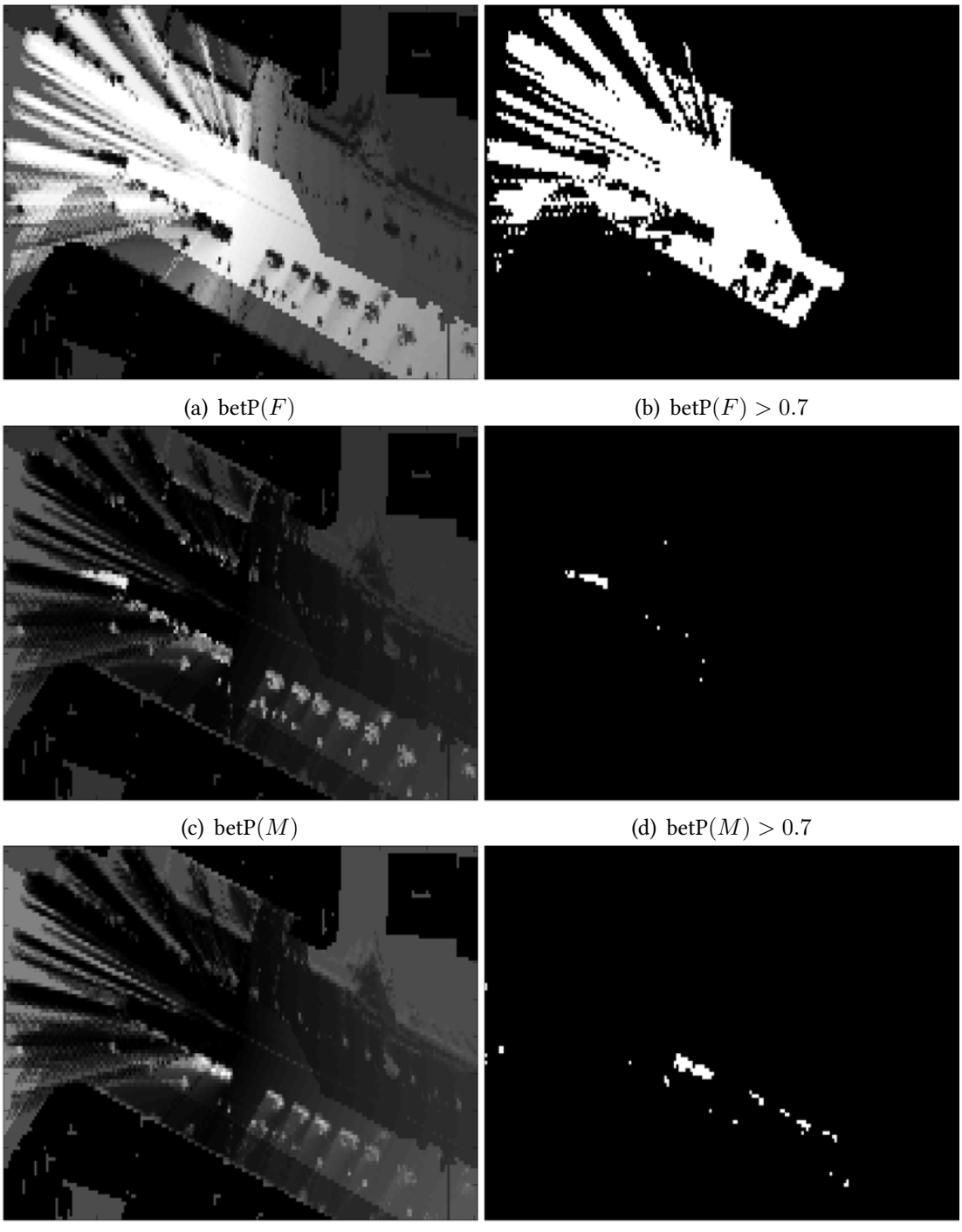
Figure 8.2 – `PerceptionGrid` comparison and scene snapshot. Using classical (uniform) discounting. Colour code as in Figure 8.1. Class F denotes any free space (drivable or not).

(a) Scene representation by a camera image.



(b) `PerceptionGrid` – pignistic probability.

Figure 8.3 – Camera acquisition and the pignistic probabilities of `PerceptionGrid`. Coordinates in meters in ENU frame.



(a) betP(F) (b) betP(F) > 0.7

(c) betP(M) (d) betP(M) > 0.7

(e) betP(S) (f) betP(S) > 0.35

Figure 8.4 – `PerceptionGrid`. Left column: pignistic probability (betP) for different classes. Right column: a simple decision rule example – threshold on pignistic probability. Classes denote respectively: F — free space, M — moving obstacles, S — stopped obstacles.

In the above results, we have filtered only the cells that have been reached by the sensor at least once in order to put forward the sensor information. Below, Figures 8.3 and 8.4 show all the cells. Both these figures present the same scene. Ahead at the left side, one can observe a moving white car, whereas closer at the left, a column of vehicles parked on the road surface. Figure 8.3a shows the scene captured by a FishEye camera. On Figure 8.3b, we present the singleton pignistic probabilities of the masses contained in `PerceptionGrid`. Vehicle position and its direction are represented on this figure by a red cross and a black arrow, respectively. We have changed the colour code, so that the classes of interest could be easily visible in the figure. The corresponding colours are as follows: F – white, I, U – blue, M – red and S – green. Other colours are the result of the fact that the pignistic probabilities (and their corresponding colours) are mixed together.

To better understand the information contained in the `PerceptionGrid`, we display, in Figure 8.4, the pignistic probabilities at a grid level of a few classes of interest. The left column contains images which are a visualisation of the pignistic probability for a given class. The right-column images give an example of a simple binary decision rule based on a threshold of the value of pignistic probability. These figures highlight the ability of the method to distinguish different classes.

One can spot the effect of the discounting, especially, in Figures 8.4a and 8.4b. Namely, the space far behind the ego-vehicle is no more recognised as free, since it has been observed for a long time. Figures 8.4d and 8.4f demonstrate the performance of the proposed approach in classifying moving cells M and stopped cells S. One can remark as well a few outliers due to the sensor noise.

Obtained results constitute only the first level of a perception system. Fully exploiting these data would mean performing further processing on the resulting grid. Clustering the cells into more meaningful object-level information and tracking these objects would be the next step towards the scene understanding.

8.2 Obstacle detection

The results for a particular instant of the approach tested on real data are presented in Figures 8.5 and 8.6. In this case, we have used the maps from OpenStreetMap (OSM) project for building models and IGN map to obtain road surface. The reported scenes were recorded while the vehicle was moving to illustrate the performance in real urban traffic conditions, typically at a speed of 40 km/h.

The topmost images show camera captures, while the central images present `PerceptionGrid` (PG) in a fixed Cartesian frame, zoomed in around the vehicle location. The visualization of PG was obtained by assigning to each class a color proportional to the pignistic probability betP and calculating the mean color. Images containing grids contain markers to show the vehicle position (a small red cross) and vehicle speed vector (a black arrow). Light dashed white lines show the approximate limits of the camera's field of view, in order to link the image with the grid. Note that the field of view of the lidar is wider than that of the camera, and for clarity is not shown. The bottommost images reflect the result of a decision rule that involved thresholding pignistic probabilities (see Equation 3.40). The different thresholds were set to 0.5 except for class S for which the threshold was 0.35, since we wanted to magnify the effect of detection of obstacles stopped momentarily.

Figure 8.5 presents quite a complex scene with multiple moving vehicles together with a few stopped vehicles. The two moving motorcycles and the moving car in the opposite lane are clearly detected, as shown by the red cells in the bottom figures. Behind these moving cells, the state of the space is unknown, which is consistent with the lidar capabilities. The car ahead, waiting at a traffic light, has

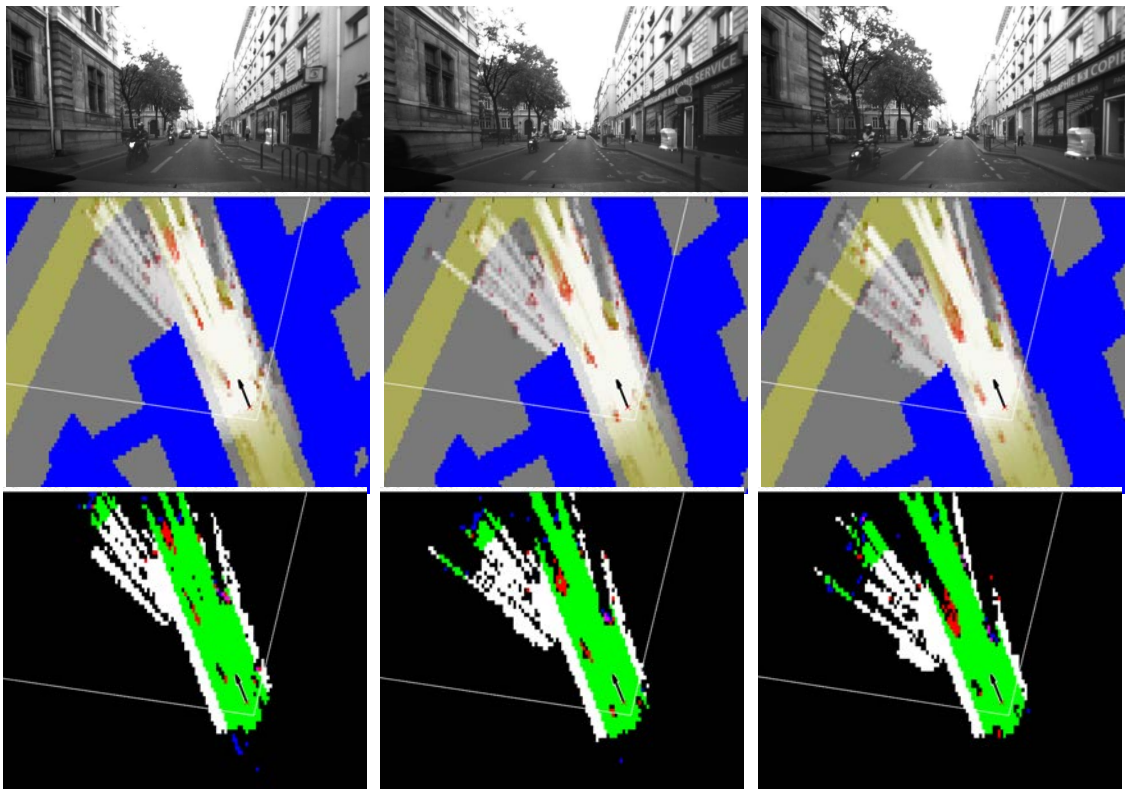


Figure 8.5 – Scene 1 as it unfolds: time goes from left to right. From top to bottom: (a) scene capture, (b) `PerceptionGrid` pignistic probability, (c) simple decision rule to detect free space, moving and stopped obstacles, Color code for Figures (3): green – drivable space D, white – non-drivable free space N, red – moving objects M, blue – stopped objects S, black – unknown Ω. Infrastructure (classes U and I) has not been visualised in the bottom figures.

been detected as stopped (see the blue cells on the right with respect to the direction of the arrow). Similarly, cars parked on the left side road are detected as stopped (blue cells in the bottom right of the grids). It will be remarked that even though these vehicles are hardly visible on the camera images, they have been detected by the perception system. When the size of the objects is substantially reduced, the lidar can miss them. This can create slightly odd effects in the perception scheme that may, for instance, cause traffic signs to oscillate between moving and stopped. This explains the isolated red/blue cells in the grid.

Figure 8.6 presents another complex scene containing three cars moving in the opposite direction (visible only in some photos), one parked car, one parked bus and a motorcycle going in the same direction as the equipped vehicle. Moving cars (in red) are clearly distinguished in the bottom images. Drivable (green) and non-drivable (white) spaces are well characterised and clearly separated. The partially visible bus and the car parked on the left (blue) are also successfully detected.

The additional information provided by the map clearly enhances the driving scene understanding. The system is able to make a clear difference between moving (red) and stopped (blue) objects. We have noticed from other sequences that stopped objects are perceived as distinct from infrastructure when prior map information is available. In addition, thanks to the prior knowledge, stationary objects such as infrastructure are distinguished from stopped objects on the road. This is a behaviour similar to that of the system using 3D city models proposed by Cappelle et al. (2012).

Finally, the effect of discounting is noticeable, particularly behind the vehicle, as the information about the environment is being forgotten with different rates thanks to the map. As the grid cells become discounted, the mass on the different classes diminishes gradually. The thresholded plots show that the

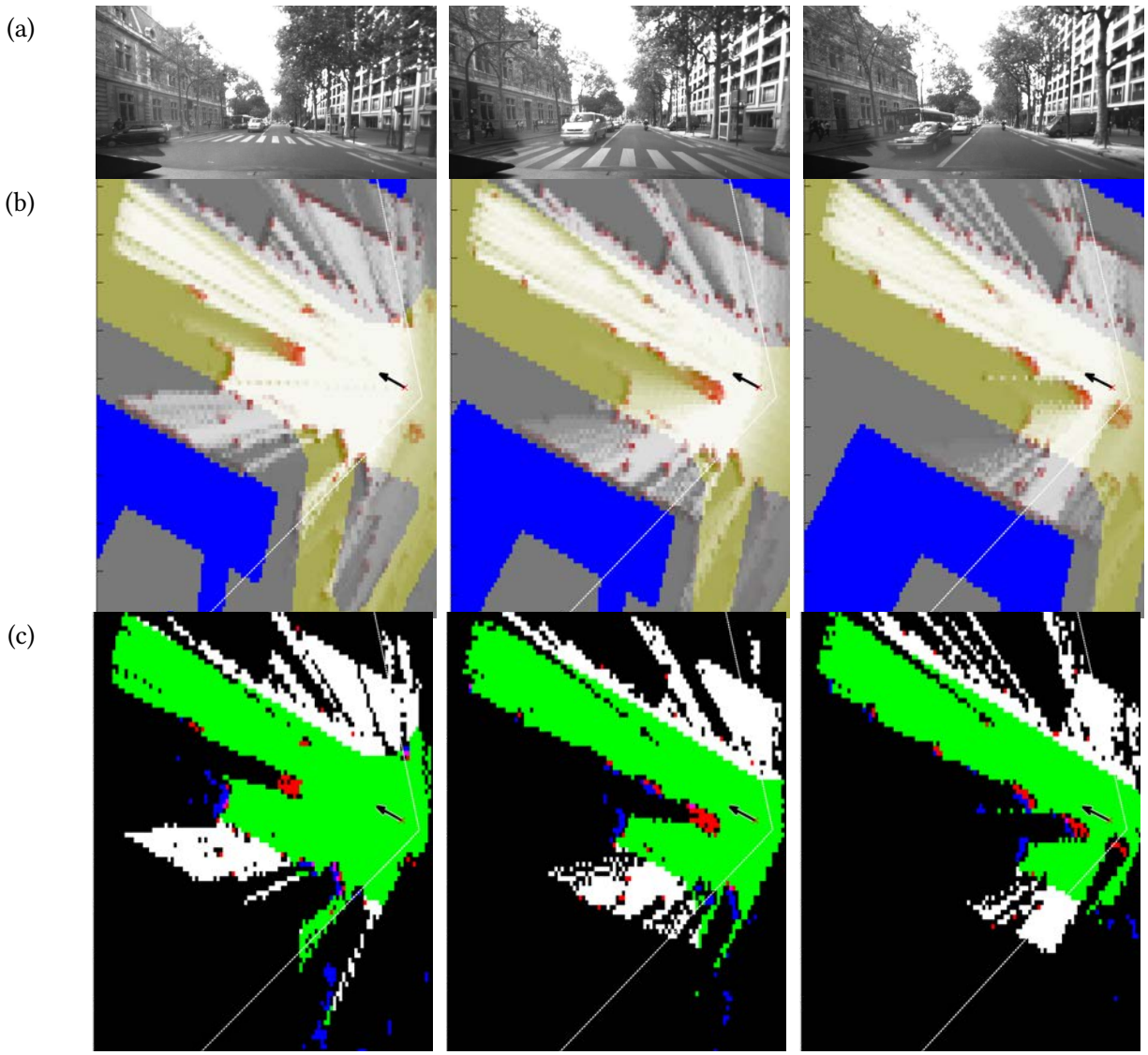
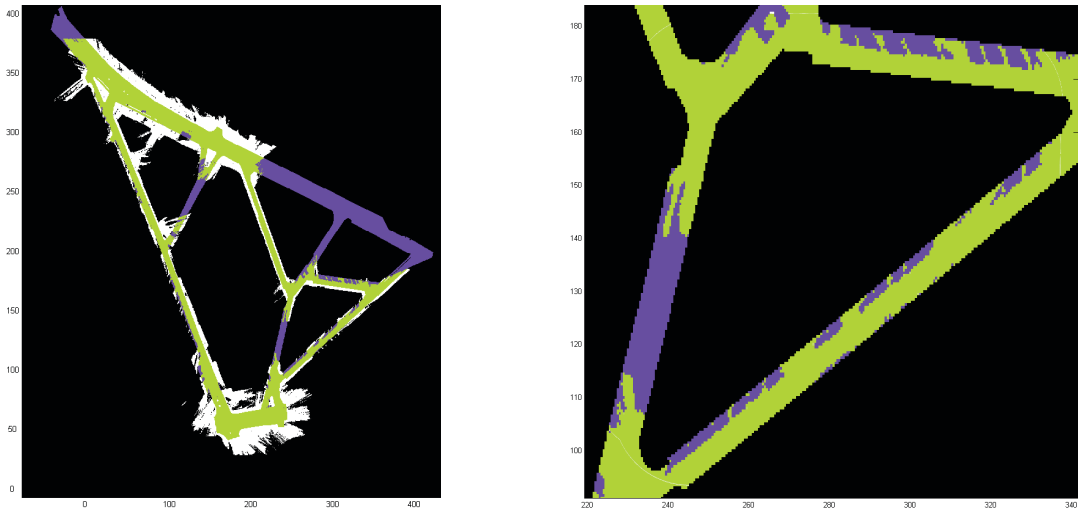


Figure 8.6 – Scene 2 as it unfolds. Analogous to Figure 8.5.



(a) Whole test sequence area.

(b) Zoom on drivable space D.

Figure 8.7 – Drivable space D (in green) and free non-drivable N (in white) accumulated over the complete test sequence. Road surface from maps in violet. Both axes in meters.

stopped information is more remanent as some blue cells are left behind.

8.3 Free space detection and characterisation

Our method is able to characterize the drivable space, i.e. the part of the road surface on which a wheeled vehicle may move. The navigable space refers usually to the capability of planning a feasible trajectory. Some other method, for instance the one proposed in (Schreier and Willert 2012), could be applied on the resulting evidential grid to define the navigable space. Figure 8.7 is the result of accumulating subsequent grids for drivable and non-drivable free spaces after having executed the pignistic decision rule. These figures show all the cells that were identified as free in at least one instant in the test sequence, either on the road or on the side-walk. These cells are shown in green for the drivable space and in white for the non-drivable space. Violet represents a superimposition of the prior map information about the road surface from the `GISGrid` (GG). It is thus possible to identify areas that the vehicle perceived during the test. More interestingly, in Figure 8.7b the places where cars or other vehicles are parked can be clearly recognised. On the other hand, the non-drivable free space in Figure 8.7a (in white) exhibits zones that are not normally used for driving, but that could be useful in special circumstances, such as when taking action to avoid colliding with a pedestrian.

8.4 Conclusion

The presented results are qualitative, but demonstrate the interest of the developed method. The use of map data, not surprisingly, improves the performance of the perception system by exploiting additional clues that this type of prior knowledge provides. Thanks to maps, we are able to distinguish the drivable free space from the non-drivable space. Mobile objects are separated into moving and stopped classes. Moreover, the infrastructure is detected as well.

An obvious remark and a possible enhancement would be to quantify the results and supply a comparison with other methods. If such results are strongly desired, the current problem is the lack of necessary reference data: it would have to contain annotations for each object at every sensor data acquisition.

On the other hand, using simulations to provide quantitative results is in our eyes simplistic and cannot handle the issues present in real driving environments.

We have presented the behaviour of the fusion rule, with varying parameters, in Chapter 5. In this chapter however, we present no parameter study and all the parameters have fixed values. We judge that this can be justified by the fact that an extensive manual tuning has been carried out during the experiments. Furthermore, a rationale has been given for the choice of all parameters in Chapter 7. And at last, many of these parameters have a clear physical interpretation.

Part V
Conclusion

This final part presents a conclusion of the realised research. Apart from outlining proposed approaches, Chapter 9 discusses the utility of such methods, their advantages and inconveniences. We mention theoretical and technical difficulties encountered while developing a perception system for intelligent vehicles. Finally, possible enhancements and improvements are enumerated in order to give directions for further in the promising domain of intelligent vehicles.

[This page intentionally left blank.]

Chapter 9

Conclusion & perspectives

"Litterarum radices amarae sunt, fructus dulces."
"The roots of scholarship are bitter, its fruits are sweet."

Marcus Tullius Cicero, *Amicitia (64,8)*

Contents

9.1 Conclusion

9.1.1 Map-based perception using unified framework: belief functions theory

This thesis has presented a mobile perception scheme for intelligent vehicles. The novelty of the approach is to extract prior knowledge from digital maps. The method exploits geographic information in order to refine the hypotheses induced by data of an exteroceptive sensor. To permit the fusion of data coming from such different sources and to manage their levels of uncertainty, we applied the method based on evidential grids. Such data structures use the belief functions theory (BFT) and are an extension of occupancy grids.

In our approach, the map data allows to infer some contextual information about the environment. The possible contexts are: roads, infrastructure and intermediate space. Each context favours the existence of different types of objects. For instance, one should not find buildings on the road, nor drivable space outside of the road context.

A modified fusion rule, based on the classic Dempster's conjunctive operator, has been elaborated. It takes into account the existence of moving and static objects. The detection of mobile obstacles is

performed thanks to the analysis of conflictual information. On the other hand, the objects that are possibly moving but currently stopped are distinguished from other non-mobile objects with the aid of an accumulator and mass specialisation. Other static parts of environment, like infrastructure are detected as well. The developed approach attempts to characterise the free space by separating it into two classes: drivable and non-drivable. This distinction is important in order to use the elaborated perception grids for trajectory planning.

An important part of the fusion rule concerns the use of previous result to predict the current state. Due to the dynamic character of modelled environment, the previous state is no longer totally reliable when it comes to describe the current state. For this reason, the outdated information has to be discounted. The perception grids are used to distinguish many classes of objects and free space, each one with its own characteristics. We have therefore opted to study the remanence of these classes and to discount them accordingly. In order to represent the variation in information lifetime of such objects, the contextual discounting has been used.

9.1.2 Contextual temporal discounting

Any information fusion system that tries to reason about the state of a dynamic system should take into consideration its evolution in time. In a part of such systems, one would observe that a new piece of information adds new knowledge that remains valid infinitely, or for a period of time that is long enough to be modelled as infinite in practice. There are however other fusion systems where it would be judged necessary to forget the old information as time passes. We have given the example of our perception system, but there are others that are similar. For instance, all the systems where the estimation of the future state is based on historical data or statistics should take into account the up-to-dateness of processed information. A stock exchange prediction system would be strongly biased and probably inefficient if it considered information from last year to predict tomorrow's ratings.

In order to manage the age of information in a fusion system, methods for temporal discounting were proposed. We demonstrated that existing algorithms, such as uniform discounting, contextual discounting as well as the generalised form of the latter do not meet the requirements needed for temporal information fusion. Namely, the former method, albeit the classical and the most widely used approach for information decay, cannot model varying persistence of different types of knowledge. Contextual variants proposed by Mercier go a step further. The original approach provides a way to define discount factor for disjoint sets of classes. The generalised method made it possible to specify these factors even if they are known for sets with a non-empty intersection. Unfortunately, even those contextual discounting methods fail when trying to apply them to temporal discounting.

To address this problem, we proposed a family of discounting operations. Variants called conservative, proportional and optimistic correspond to different knowledge about dependencies between discounted classes. The conservative approach represent the first extreme case where masses affected to a class depend on each of its subclasses and are hence discounted in the same manner. Inversely, the optimistic discounting is to be used in the case where a set of classes do not affect its supersets. An intermediate solution, the proportional operator, discounts the supersets proportionally to the cardinality of the subset concerned by the discount factor. A generalisation of this family of operations allows us to adopt a method that benefits from a fine-grained knowledge about the dependencies between object types.

9.1.3 Implementation and results

9.1.3.1 Results of perception on real data

The presented results prove that the use of map data in a perception system for self-driving vehicles is going to thrive in the near future. The importance of handling both aspects mentioned above, namely the fusion of map and sensor data along with the management of temporal information, has been demonstrated using real data. We have performed multiple tests on our test-bed vehicle.

9.1.3.2 Real-time implementation on a test-bed vehicle

An important part of this research work was the implementation of tested approaches on C++ Pacpus framework in an equipped car (Heudiasyc 2013). The vehicle, called Carmen, is presented in Figure 7.4. Described methods have been implemented using a modular approach. Proceeding in this way permitted us to test various parts of the developed system by a simple update, addition or removal of concerned components. The component-based architecture has been presented in Figure 7.1.

One of the advantages of such a structure of our system was that multiple sensor models and discount methods could have been tested. Moreover, the fusion algorithm was made independent of the exact type of sensor in use. Such a decomposition of our system permits to easily develop the method by adding supplementary modules, e.g. a trajectory planner or a new sensor.

9.2 Perspectives

9.2.1 Method validation

Reference data One of the perspectives for future work is the use of reference data to validate the results. This is somehow problematic as no reliable and precise data set for perception of intelligent vehicles exist so far. One of the attempts, the KITTI dataset, which has become popular recently (Geiger, Lenz, et al. 2013), could be a potential candidate for a reference data set. This database provides both camera and lidar data synchronised properly with localisation information from a hybrid Global Positioning System (GPS)-Inertial Measurement Unit (IMU) system. Moreover, the data are annotated, e.g. objects are described by their class (car, van, truck, pedestrian, etc.), bounding box and 3-dimensional (3D) dimensions. Each object is annotated with its translation and rotation with respect to the reference frame. Unfortunately, the KITTI has a major flow preventing the use for the presented perception system. Namely, the object data is obtained using the Velodyne lidar sensor that is used as well to provide the point cloud data. Such a situation means that the learning and test data are obtained from the same sensor and hence, using them for validation purposes would be biased and lead to overestimating the capabilities of the tested system.

Learning parameters In any case, a reference data set, even a biased one, would be a valuable resource to improve the described system. First of all, algorithm parameters could be learnt automatically or semi-automatically. As presented in Chapter 6, the remanence of different object classes can be inferred through a learning process when enough data is available. Another parameter that could be learnt from reference data is the most appropriate discounting scheme. That is to say, without any knowledge about the interdependencies between various object classes, one could find which type of

discounting method should be used. This could help to decide between optimistic and conservative approaches or find the proper parametrisation of the general discounting rule.

Algorithm validation Apart from defining the best parameters of the perception system, a reference data set could help to choose the most appropriate fusion rule. It would be of high value to determine whether proposed fusion rule based on Yager's operator behaves better than other rules, like conjunctive rule or cautious rule. What is more, possessing a reference data set offers a possibility to quantitatively validate the results. The verification that we have performed and which would be interesting to develop is the projection of the resulting perception grid into the camera image. This approach permits to visually verify the algorithm.

9.2.2 Map-based localisation

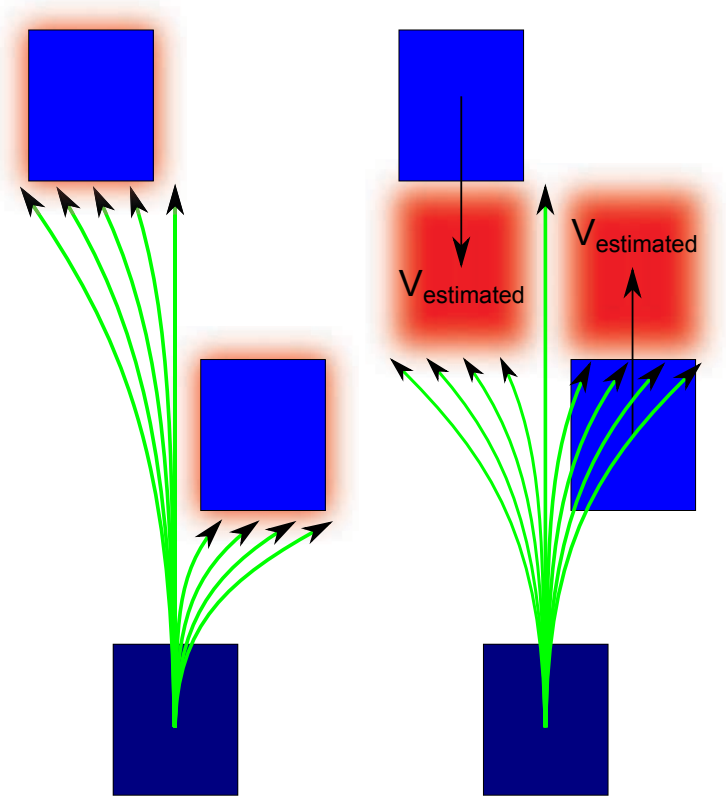
Map data It is envisioned that the hypothesis of accurate maps will be removed. Considerable work on creating appropriate error models for the data source will be needed. Such an improvement will be a step towards the use of our approach for navigation system in autonomous vehicles. Map information will be used to predict object movements. Lastly, more work is to be done to fully explore and exploit 3D map information.

Localisation An already started work is to perform a map-based localisation module using map data and a lidar. Such an approach would possibly make a Global Navigation Satellite System (GNSS) module unnecessary or limit its usage only to get the initial guess about the vehicle position. The basic idea is to find the correspondence between the buildings and other elements of infrastructure obtained from two sources. The first source being a lidar-based perception method and the second one — a map. The research on this subject has started and preliminary results have been described in (Mendes de Farias 2013).

Object-level description An important stage would be to pass from the cell-level grid description to a higher level one. For instance, it would be a huge benefit for the robustness of the perception system to describe detected obstacles at the object level. This is possible through segmentation of the perception grid, extracting object information from it and tracking these objects. While tracking, one can also estimate and predict the speed and the direction of each object. Furthermore, the predicted position of an object may be used to improve the next detection step by limiting the conflictual information. Of course, such additional processing needs computational resources, which is to be taken into account when building a real-time system.

9.2.3 Vehicle navigation

Trajectory planning Coupling the perception module with a system for trajectory planning. An immediate candidate would be the so called tentacle method. Tentacle methods, e.g. as the one described by Hundelshausen et al. (2008), use input occupancy grids to define the driving corridor. The presented perception system would further improve such approaches. Firstly, describing explicitly the drivable and non-drivable free space would avoid problems with the distinction of. Secondly, object-level description is another enhancement that would permit to predict the movement of other road users and to adapt accordingly the trajectory. One should imagine that detecting two objects, one going in the



(a) Without movement prediction.

(b) With prediction of object movement.

Figure 9.1 – Comparison of a tentacle-based trajectory planning algorithm. Dark blue: ego-vehicle, blue: detected vehicles, red: estimated positions of vehicles at the next execution of trajectory planning algorithm. Note that this figure does not take into account the geometric model of the ego-vehicle, thus allowing situation where the middle tentacle engages in a very narrow corridor between vehicles.

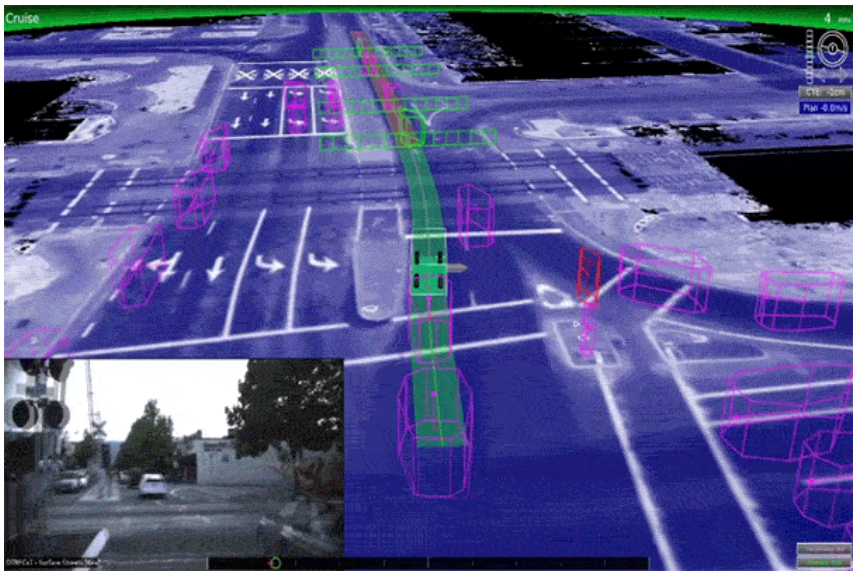


Figure 9.2 – Illustration of using virtual fences to limit possible car trajectory. Fences in green visible in front of the vehicle. Source: http://www.extremetech.com/extreme/189486-how-googles-self-driving-cars-detect-and-avoid-obstacles.

same direction as the intelligent vehicle, the other going in the opposite direction would need different handling. The former would be expected to advance, so that the candidate tentacles go further up to the new, predicted, position of the object. For the latter object, the candidate trajectories would be shorter, as the estimated position limits the possible manoeuvres. A schematic illustration of such behaviour can be seen in Figure 9.1. Additionally, an important modification of the tentacle approach would be to use fully the rich information encoded in perception grids. Namely, such an algorithm should take into account the certainty that cells on a candidate trajectory are free and use this quantity as an additional score for judging the tentacle's suitability. Besides, a Ph.D. thesis is starting on a similar subject starts at the University of Technology of Compiègne in Autumn 2014.

Emergency mode Another possibility of enhancing the trajectory planning system would be to add an emergency driving mode. One of important differences between this mode and the normal cruise mode would be to permit the vehicle to drive on (normally) non-drivable space N. This can be motivated by the behaviour of human drivers. For instance in case of an emergency vehicle approaching a car, the car's driver would move its vehicle to let the emergency move on. If necessary, he would need to drive on pavements or grass, otherwise risking to block the approaching vehicle and possibly lead to catastrophic consequences. Another situation where the necessity of going onto the non-drivable space shows up is the obstacle avoidance. More precisely, the manoeuvre of critical obstacle avoidance that is decisive in the choice between injury and death. One can easily imagine the situation where the driver (human or autonomous) dodges away on the emergency lane or on the grass in order to avoid the head-on collision with an oncoming vehicle. If we can ever dream of intelligent vehicles on our roads, such scenarios have to be taken into account.

Incorporating traffic rules Another idea would be to add the map information about road lanes and traffic rules to the resulting grid through conditioning. For example, a turn restriction would modify the grid provided to the trajectory planning module; this will happen in the way so that the grid itself restricts the drivable free space. This technique is conceptually similar to existing methods in trajectory planning, where virtual barriers limit the possible movements of the vehicle, as illustrated in Figure 9.2.

9.2.4 Technicalities

Implementation improvements A grid-based approaches hugely benefit from parallel processing, as the computations performed on each cell are identical. An efficient implementation would take this fact into account and use techniques of programming on massively parallel processors, like General-Purpose computing on Graphics Processing Unitss (GPGPUs). We have already performed successful tests using CUDA and OpenCL libraries for this purpose. These preliminary tests have shown that by applying this technique, one could largely decrease necessary computation times.

[This page intentionally left blank.]

Part VI
Appendices

[This page intentionally left blank.]

Appendix A

Proofs

A.1 Discounting

Proof A.1.1 (Classical discounting expressed in terms of conservative discounting)**.**

$$\overset{\alpha}{c}\mathrm{m}(\emptyset) = \mathrm{m}(\emptyset) \cdot (1 - \alpha_\Omega) \tag{A.1}$$

is obtained directly from Equation 6.20.

Then, from Equation 6.21, we have $A \cap \theta = A \cap \Omega \neq \emptyset$ for any given A. That provides us with:

$$\overset{\alpha}{c}\mathrm{m}(A) = \mathrm{m}(A) \cdot (1 - \alpha_\Omega) \qquad \forall A \subsetneq \Omega, A \neq \emptyset \tag{A.2}$$

Eventually,

$$\overset{\alpha}{c}\mathrm{m}(\Omega) = \mathrm{m}(\Omega) \cdot (1 - \alpha_\Omega) + \alpha_\Omega \tag{A.3}$$

This is obtained thanks to Equation 6.22, which leads us to:

$$\begin{aligned} \mathrm{m}(\emptyset) + \sum_{\substack{B \subseteq \Omega \\ B \cap \theta \neq \emptyset}} \mathrm{m}(B) &= \mathrm{m}(\emptyset) + \sum_{\substack{B \subseteq \Omega \\ B \cap \Omega \neq \emptyset}} \mathrm{m}(B) \\ &= \sum_{B \subseteq \Omega} \mathrm{m}(B) = 1 \end{aligned} \tag{A.4}$$

Development A.1.2 (Comparison between κ and α)**.** *Using Equation 6.42 and the fact that $A \subseteq \theta =$*

$\{\emptyset,\ \theta\}$ *for a singleton* θ*, we obtain:*

$$\kappa_\theta \equiv \text{bel}_\Theta(\theta) = \sum_{A \subseteq \theta} \text{m}_\Theta(A) \tag{A.5}$$

$$= \sum_{A \subseteq \theta} \left[\prod_{\substack{B \in \Theta \\ B \subseteq A}} \alpha_B \cdot \prod_{\substack{B \in \Theta \\ B \not\subseteq A}} (1 - \alpha_B) \right] \tag{A.6}$$

$$= \prod_{B \in \Theta} (1 - \alpha_B) + \alpha_\theta \cdot \prod_{\substack{B \in \Theta \\ B \not\subseteq \theta}} (1 - \alpha_B) \tag{A.7}$$

$$= (1 - \alpha_\theta) \cdot \prod_{\substack{B \in \Theta \\ B \not\subseteq \theta}} (1 - \alpha_B) + \alpha_\theta \cdot \prod_{\substack{B \in \Theta \\ B \not\subseteq \theta}} (1 - \alpha_B) \tag{A.8}$$

$$= (1 - \alpha_\theta + \alpha_\theta) \cdot \prod_{\substack{B \in \Theta \\ B \not\subseteq \theta}} (1 - \alpha_B) \tag{A.9}$$

$$\kappa_\theta \equiv \prod_{\substack{B \in \Theta \\ B \not\subseteq \theta}} (1 - \alpha_B) \tag{A.10}$$

Proof A.1.3 (Postulate 6.1)**.** *Using Equation 6.16 and 6.5, we have directly:*

$$^{t_{1/N}}\text{m}(A) = \text{m}(A) \cdot e^{-\lambda t_{1/N}} \tag{A.11}$$

$$= \text{m}(A) \cdot e^{-\frac{\ln N}{t_{1/N}} t_{1/N}} \tag{A.12}$$

$$= \text{m}(A) \cdot e^{-\ln N} \tag{A.13}$$

$$= \text{m}(A) \cdot \frac{1}{N} \tag{A.14}$$

Proof A.1.4 (Postulate 6.1)**.**

$$^{t_2}\left(^{t_1}\text{m}(A)\right) = {}^{t_2}\left(\text{m}(A) \cdot e^{(-\lambda t_1)}\right) \tag{A.15}$$

$$= \text{m}(A) \cdot e^{(-\lambda t_2)} e^{(-\lambda t_1)} \tag{A.16}$$

$$= {}^{t_1}\text{m}(A) \cdot e^{(-\lambda t_2)} \tag{A.17}$$

$$= {}^{t_1}\left(^{t_2}\text{m}(A)\right) \tag{A.18}$$

Proof A.1.5 (Postulate 6.1)**.**

$$^{t_2}\left(^{t_1}\text{m}(A)\right) = {}^{t_2}\left(\text{m}(A) \cdot e^{(-\lambda t_1)}\right) \tag{A.19}$$

$$= \text{m}(A) \cdot e^{(-\lambda t_2)} e^{(-\lambda t_1)} \tag{A.20}$$

$$= \text{m}(A) \cdot e^{(-\lambda (t_1 + t_2))} \tag{A.21}$$

$$= {}^{t_1 + t_2}\text{m}(A) \tag{A.22}$$

Appendix B

Implementation notes

B.1 Software analysis

This section presents the analysis of functional and non-functional requirements that should be met by a perception system of an intelligent vehicle. These requirements were not necessarily all met in the implemented system, but this analysis helped to a great extent in the development process. A production-ready system would inevitably follow similar analysis requirements and therefore, we present them for educational purposes.

B.1.1 Functional requirements

1. ***Reading map data***
   System MUST[1] read following formats of map databases: National Institute of the Geographic and Forest Information (IGN), OpenStreetMap (OSM). System MAY read formats: Geography Markup Language (GML), City Geography Markup Language (CityGML). System MAY read COLLADA, KML, 3DS, BeNomad formats. System MAY fetch online data from Google Maps.
   Rationale: Map databases that we possess are in IGN format. OSM data are freely available (OSM 2013).

2. ***Map download***
   System MAY download map data on-the-fly when the map of the terrain is needed.
   Rationale: OSM project make map data available for download, both raster map images and vector maps.

3. ***Data filtering***
   preprocessing data, leaving only necessary information
   Rationale: Needed for handling real-world data.

4. ***Reading lidar data***
   System MUST read lidar point clouds (scans).
   Rationale: Principal sensor.

5. ***Reading position data***
   System MUST read the global position (or pose if available) of the vehicle from a GNSS sensor

[1] The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

such as GPS.
Rationale: Localisation.

6. ***Reading orientation, speed and acceleration data***
System SHOULD read the vehicle orientation, speed and acceleration from available sensors such as IMU or Inertial Navigation System (INS) *if* they are available.
Rationale: Necessary for grid combination.

7. ***Graphical User Interface (GUI)***
User SHOULD dispose of a GUI to access to all functions of the system.
Rationale: Easier communication with user, fast validation.

B.1.2 Non-functional requirements

101. ***Compatibility***
Software MUST be compatible with PACPUS platform.
Rationale: Collaborative development.

102. ***Compliance***
System MAY be compliant with ISO 26262 standard. System MAY be compliant with AUTomotive Open System ARchitecture (AUTOSAR).
Rationale: An emerging standard of functional safety in road vehicles (Langheim et al. 2010)[1].

103. ***Documentation***
System MUST be well documented. System MUST possess a general design rationale, software analysis and architecture documentation. Code MUST be documented in computer-readable format conforming to Doxygen[2] automatic documentation generator tool.
Rationale: The whole or parts of the system will be reused and/or modified, so it is necessary that a sound documentation be available.

104. ***Extensibility***
Software SHOULD be easily extensible to add new functionalities or enhancements.
Rationale: Changing requirements, new devices and algorithms.

105. ***Maintainability***
Software SHOULD be easily maintainable. Maintenance tasks such as defect isolation and correction, changing or new requirements SHOULD be easy to achieve. It can be achieved by a modular, high-cohesion and low-coupling design and change prediction.
Rationale: Changing requirements.

106. ***Modifiability***
Software SHOULD be easily modifiable. Clarity of code SHOULD have priority over performance.
Rationale: Changing requirements.

107. ***Performance***
Soft real-time timing requirements needed. Update rate $\rho >= 10Hz$ (depends on sensor frequency). Update — reading and processing data, returning information.
Rationale: A perception and navigation systems should be responsive to the situation on the road. Triebel et al. define a responsive vehicle behaviour as one when planning and replanning of the

[1] http://www.iso.org/iso/fr/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=54591
[2] Doxygen documentation can be found at http://www.stack.nl/~dimitri/doxygen/.

path can be performed at the frequency of 10 Hz ("Proceedings of the IROS 2009 3rd Workshop: Planning , Perception and Navigation for Intelligent Vehicles (PPNIV)" 2009).

108. ***Portability***
Software MUST be portable to Windows and Linux platforms.
Rationale: Ease of development. Possibility of change in platform.

109. ***Fault tolerance***
When no position data from localisation system is available.
Rationale: Passenger safety.

B.2 Algorithms

In our implementation, occupancy grid transformations (rotation, translation and interpolation) have been performed using the image processing library OpenCV (Itseez 2012).

B.2.1 Data structures

Efficient spatial data storage and queries were done thanks to R-Trees (Guttman 1984) implementation in Boost.Geometry library. These trees are an index mechanism created for geo-data applications in order to help them retrieve data items quickly according to their spatial locations. This data structure became very popular and had several followers in research. Just to mention of them, so called priority R-Trees focused on improving the worst-case efficiency (Arge et al. 2004).

B.3 Third-party libraries

- Boost[1]
  Portable C++ source libraries.
- Compute Unified Device Architecture (CUDA)[2]
  Proprietary but free parallel computing architecture.
- OpenCV[3]
  Real-time computer vision library.
- CMU 1394 Camera [4]
  Driver and software library for cameras that comply with the 1394 Digital Camera Specification (FireWire).
- Perception et Assistance pour une Conduite Plus Sure (PACPUS)[5]
  Research platform for intelligent vehicle systems. C++ platform created in the Heudiasyc laboratory for Advanced Driver Assistance System (ADAS) and perception systems. The author is co-creator of this library.

[1] Available at http://www.boost.org.
[2] Available at http://developer.nvidia.com/cuda-zone.
[3] Available at http://opencv.org.
[4] Available at http://www.cs.cmu.edu/~iwan/1394.
[5] Available at http://www.hds.utc.fr/pacpus.

- Point-Cloud Library (PCL)[1]
  Open-source project for 3D point cloud processing. Library for manipulation of clouds of points, such as lidar data.
- Qt[2]
  Cross-platform application and user interface framework.
- Robot Operating System (ROS)[3]
  Open-source middleware framework for robot applications used by PACPUS. A framework for robotic real-time systems. (Quigley et al. 2009)

[1] Available at http://pointclouds.org.
[2] Available at http://qt-project.org.
[3] Available at http://www.ros.org.

Bibliography

Aitken, V. (2005). "Evidential Mapping for Mobile Robots with Range Sensors". *IEEE Instrumentation and Measurement Technology Conference*. Vol. 3, pp. 2180–2185.

Alcantarilla, P. F., Ni, K., Bergasa, L. M., and Dellaert, F. (2011). "Visibility Learning in Large-Scale Urban Environment". *IEEE International Conference on Robotics and Automation (ICRA)*. Shanghai, pp. 6205–6212.

Arge, L., Berg, M. de, Haverkort, H. J., and Yi, K. (2004). "The Priority R-Tree: A Practically Efficient and Worst-Case Optimal R-Tree". *SIGMOD Electronic Proceedings*. Paris: ACM.

Ayoun, A. and Smets, P. (2001). "Data association in multi-target detection using the transferable belief model". *International Journal of Intelligent Systems* 16.10, pp. 1167–1182.

Baig, Q. (2012). "Multisensor Data Fusion for Detection and Tracking of Moving Objects From a Dynamic Autonomous Vehicle". Ph.D. thesis. Université de Grenoble.

Baig, Q., Aycard, O., Vu, T.-D., and Fraichard, T. (2011). "Fusion Between Laser and Stereo Vision Data For Moving Objects Tracking In Intersection Like Scenario". *IEEE Intelligent Vehicles Symposium (IV)*, pp. 362–367.

Barnett, J. A. (2008). "Computational Methods for A Mathematical Theory of Evidence". *Classic Works of the Dempster-Shafer Theory of Belief Functions*, pp. 197–216.

Bayes, M. and Price, M. (1763). "An Essay towards Solving a Problem in the Doctrine of Chances. By the Late Rev. Mr. Bayes, F. R. S. Communicated by Mr. Price, in a Letter to John Canton, A. M. F. R. S." *Philosophical Transactions of the Royal Society of London* 53, pp. 370–418.

Benenson, R. and Parent, M. (2008a). "Design of an urban driverless ground vehicle". *IEEE International Conference on Intelligent Robots Systems (IROS)*.

Benenson, R. and Parent, M. (2008b). "Design of an urban driverless ground vehicle". *IEEE International Conference on Intelligent Robots Systems (IROS)*, pp. 16–21.

Ben-Moshe, B., Carmi, P., and Friedman, E. (2014). "Modeling GNSS Signals in Urban Canyons using Visibility Graphs and 3D Building Models".

Bernoulli, J. (1713). *Ars Conjectandi*. Basel: Thurnisius.

Boucher, C. and Noyer, J.-C. (2011). "GNSS-based Unscented Filtering for Road Map Databases Management". *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1398–1403.

Broggi, A., Bombini, L., Cattani, S., Cerri, P., and Fedriga, R. I. (2010). "Sensing Requirements for a 13,000 km Intercontinental Autonomous Drive". *IEEE Intelligent Vehicles Symposium (IV)*, pp. 500–505.

Broggi, A., Caraffi, C., Porta, P. P., and Zani, P. (2006). "The Single Frame Stereo Vision System for Reliable Obstacle Detection used during the 2005 DARPA Grand Challenge on TerraMax TM". *International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Toronto, pp. 745–752.

Buehler, M., Iagnemma, K., and Singh, S. (2007). *The 2005 DARPA Grand Challenge*. Springer.

Burochin, J.-P. (2012). "Segmentation d'images de façades de bâtiments acquises d'un point de vue terrestre". Ph.D. thesis. Université Paris-Est.

Cao, Q., Gu, J., and Huang, Y. (2008). "Rapid Traversability Assessment in 2.5D Grid-based Map on Rough Terrain". *International Journal of Advanced Robotic Systems (IJARS)* 5.4, pp. 389–394.

Cappelle, C., El Najjar, M. E. B., Charpillet, F., and Pomorski, D. (2007). "Outdoor obstacle detection and localisation with monovision and 3D geographical database". *IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 1102–1107.

Cappelle, C., El Najjar, M. E. B., Charpillet, F., and Pomorski, D. (2008). "Obstacle detection and localization method based on 3D model: distance validation with Ladar". *IEEE International Conference on Robotics and Automation (ICRA)*. Pasadena, CA, pp. 4031–4036.

Cappelle, C., El Najjar, M. E. B., Charpillet, F., and Pomorski, D. (2012). "Virtual 3D City Model for Navigation in Urban Areas". *Journal of Intelligent Robotic Systems* 66.3, pp. 377–399.

Cheng, Y. (1995). "Mean Shift , Mode Seeking , and Clustering". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17.8, pp. 790–799.

Cherfaoui, V., Denœux, T., and Cherfi, Z. L. (2010). "Confidence Management in Vehicular Network". *Confidence Management in Vehicular Network*. Ed. by Y. Zhang and M. H. CRC Press, pp. 357–378.

CityVIP (2012). *CityVIP project: http://projet_-cityvip.byethost33.com*.

CMU (2014). *Tartan Racing @ Carnegie Mellon: www.tartanracing.org*.

Comport, A. I., Malis, E., and Rives, P. (2010). "Real-time Quadrifocal Visual Odometry". *International Journal of Robotics Research* 29.2-3, pp. 245–266.

Cooman, G. de, Quaeghebeur, E., and Miranda, E. (2009). "Exchangeable lower previsions". *Bernoulli* 15.3, pp. 721–735.

Coué, C., Pradalier, C., Laugier, C., Fraichard, T., and Bessière, P. (2006). "Bayesian Occupancy Filtering for Multitarget Tracking: An Automotive Application". *The International Journal of Robotics Research* 25.1, pp. 19–30.

Dawood, M., Cappelle, C., Najjar, M. E. E., Khalil, M., and Pomorski, D. (2011). "Vehicle geo-localization based on IMM-UKF data fusion using a GPS receiver, a video camera and a 3D city model". *IEEE Intelligent Vehicles Symposium (IV)*, pp. 510–515.

Dempster, A. P. (1968). "A Generalization of Bayesian Inference". *Journal of the Royal Statistical Society* 30, pp. 205–247.

Denœux, T. (1989). "Fiabilité de la prévision de pluie par radar en hydrologie urbaine". Ph.D. thesis. École Nationale des Ponts et Chaussées.

Denœux, T. (2006). "The cautious rule of combination for belief functions and some extensions". *International Conference on Information Fusion*. Florence, Italy: IEEE.

Denœux, T. (2008). "Conjunctive and disjunctive combination of belief functions induced by nondistinct bodies of evidence". *International Journal of Artificial Intelligence* 172.2-3, pp. 234–264.

Destercke, S. (2008). "Représentation et combinaison d'informations incertaines : nouveaux résultats avec applications aux études de sûreté nucléaires". Ph.D. thesis. Université de Toulouse.

Drevelle, V. and Bonnifait, P. (2011). "Global Positioning in Urban Areas with 3-D Maps". *IEEE Intelligent Vehicles Symposium (IV)*, pp. 764–769.

Dubois, D. and Prade, H. (1985). "Evidence Measures Based on Fuzzy Information". *Automatica* 21.5, pp. 547–562.

Dubois, D. and Prade, H. (1988). "Representation and combination of uncertainty with belief functions and possibility measures". *Computer Intelligence* 4, pp. 244–264.

Durekovic, S. and Smith, N. (2011). "Architectures of Map-Supported ADAS". *IEEE Intelligent Vehicles Symposium (IV)*, pp. 207–211.

Elfes, A. (1989). "Using Occupancy Grids for Mobile Robot Perception and Navigation". *Computer Journal* 22.6, pp. 46–57.

Elouedi, Z., Mellouli, K., and Smets, P. (2004). "Assessing sensor reliability for multisensor data fusion with the transferable belief model". *IEEE Transactions on Systems, Man, and Cybernetics* 34.1, pp. 782–787.

Eskandarian, A., ed. (2012). *Handbook of Intelligent Vehicles*. London: Springer London.

Floros, G., Zander, B. van der, and Leibe, B. (2013). "OpenStreetSLAM: Global Vehicle Localization Using OpenStreetMaps". *IEEE International Conference on Robotics and Automation (ICRA)*. Karlsruhe.

Fouque, C., Bonnifait, P., and Bétaille, D. (2008). "Enhancement of Global Vehicle Localization using Navigable Road Maps and Dead-Reckoning". *Position, Location and Navigation Symposium*. Monterey, CA: IEEE, pp. 1286–1291.

Geiger, A., Lauer, M., Moosmann, F., Ranft, B., Rapp, H., Stiller, C., and Ziegler, J. (2012). "Team AnnieWAY's Entry to the Grand Cooperative Driving Challenge 2011". *IEEE Transactions on Intelligent Transportation Systems* 13.3, pp. 1008–1017.

Geiger, A., Lauer, M., Wojek, C., Stiller, C., and Urtasun, R. (2014). "3D traffic scene understanding from movable platforms". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.5, pp. 1012–1025.

Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). "Vision meets Robotics: The KITTI Dataset". *International Journal of Robotics Research (IJRR)*.

Google (2014a). *Google Earth*.

Google (2014b). *Google Maps: maps.google.com*.

Guttman, A. (1984). "R-trees: A dynamic index structure for spatial searching". *ACM*. Vol. 14. 2, pp. 47–57.

Hähnel, D., Triebel, R., Burgard, W., and Thrun, S. (2003). "Map Building with Mobile Robots in Dynamic Environments". *IEEE International Conference on Robotics and Automation (ICRA)*.

Hammoudi, K. (2011). "Contributions to the 3D city modeling". Ph.D. thesis. Université Paris-Est.

Harris, C. and Stephens, M. (1988). "A Combined Corner and Edge Detector". *Procedings of the Alvey Vision Conference 1988* 15, pp. 23.1–23.6.

Hentschel, M. and Wagner, B. (2010). "Autonomous robot navigation based on OpenStreetMap geodata". *International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Madeira Island: IEEE, pp. 1645–1650.

Hentschel, M., Wulf, O., and Wagner, B. (2008). "A GPS and laser-based localization for urban and non-urban outdoor environments". *IEEE International Conference on Intelligent Robots Systems (IROS)*, pp. 149–154.

Heudiasyc (2013). *PACPUS plaftorm: www.hds.utc.fr/pacpus*.

Himmelsbach, M., Luettel, T., Hecker, F., Hundelshausen, F. von, and Wünsche, H.-J. (2011). "Autonomous Off-Road Navigation for MuCAR-3". *Künstliche Intelligenz (KI)* 25.2, pp. 145–149.

Himmelsbach, M., Müller, A., Lüttel, T., and Wünsche, H.-J. (2008). "LIDAR-based 3D Object Perception". *International Workshop on Cognition for Technical Systems* 1.

Hundelshausen, F. von, Himmelsbach, M., Hecker, F., Mueller, A., and Wünsche, H.-J. (2008). "Driving with Tentacles: Integral Structures for Sensing". *Journal of Field Robotics* 25.9, pp. 640–673.

IGN (2014). *IGN Géoportail: www.geoportail.gouv.fr*.

Irie, K. and Tomono, M. (2013). "Road Recognition from a Single Image using Prior Information". *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Tokyo, pp. 1938–1945.

Itseez (2012). *OpenCV vision library: opencv.org*.

Jaffray, J.-Y. (2008). "Bayesian Updating and Belief Functions". *Classic Works of the Dempster-Shafer Theory of Belief Functions*. Vol. 22. 5, pp. 555–576.

Kammel, S., Ziegler, J., Pitzer, B., Werling, M., Gindele, T., Jagzent, D., Schröder, J., Thuy, M., Goebl, M., Hundelshausen, F. von, Pink, O., Frese, C., and Stiller, C. (2008). "Team AnnieWAY's Autonomous System for the 2007 DARPA Urban Challenge". *Journal of Field Robotics* 25.9, pp. 615–639.

Khaleghi, B., Khamis, A., Karray, F. O., and Razavi, S. N. (2013). "Multisensor data fusion: A review of the state-of-the-art". *Information Fusion* 14.1, pp. 28–44.

Kim, K., Summet, J., Starner, T., Ashbrook, D., Kapade, M., and Essa, I. (2008). "Localization and 3D reconstruction of urban scenes using GPS". *IEEE International Symposium on Wearable Computers (ISWC)*. IEEE, pp. 11–14.

Klein, J. and Colot, O. (2010). "Automatic discounting rate computation using a dissent criterion". *Workshop on the theory of belief functions*. Brest.

Klein, J. and Colot, O. (2011). "Singular sources mining using evidential conflict analysis". *International Journal of Approximate Reasoning (IJAR)* 52.9, pp. 1433–1451.

Klein, J., Lecomte, C., and Miché, P. (2010). "Hierarchical and conditional combination of belief functions induced by visual tracking". *International Journal of Approximate Reasoning (IJAR)* 51.4, pp. 410–428.

Konrad, M., Nuss, D., and Dietmayer, K. C. J. (2012). "Localization in Digital Maps for Road Course Estimation using Grid Maps". *IEEE Intelligent Vehicles Symposium (IV)*, pp. 87–92.

Kruse, R. and Klawonn, F. (1994). "Mass distributions on L-fuzzy sets and families of frames of discernment". *Advances in the Dempster-Shafer Theory of Evidence*. New York, NY: John Wiley \& Sons, Inc., pp. 239–250.

Kwon, H., Ahmad Yousef, K. M., and Kak, A. C. (2013). "Building 3D visual maps of interior space with a new hierarchical sensor fusion architecture". *Robotics and Autonomous Systems*.

Langheim, J., Guegan, B., Maaziz, K., Zeppa, G., Philippot, F., Boutin, S., Aboutaleb, H., and David, P. (2010). "System architecture, tools and modelling for safety critical automotive applications – the R\&D project SASHA". *Embedded Real-Time Software and Systems*.

Larnaout, D., Gay-Bellile, V., Bourgeois, S., and Dhome, M. (2013). "Vehicle 6-DoF localization based on SLAM constrained by GPS and digital elevation model information". *IEEE International Conference on Image Processing (ICIP)*.

Lategahn, H., Beck, J., Kitt, B., and Stiller, C. (2013). "How to Learn an Illumination Robust Image Feature for Place Recognition". *IEEE Intelligent Vehicles Symposium (IV)*. June. Gold Coast, pp. 285–291.

Lategahn, H., Schreiber, M., Ziegler, J., and Stiller, C. (2013). "Urban Localization with Camera and Inertial Measurement Unit". *IEEE Intelligent Vehicles Symposium (IV)*. Gold Coast, pp. 719–724.

Lategahn, H. and Stiller, C. (2012a). "City GPS using Stereo Vision". *IEEE International Conference on Vehicular Electronics and Safety (ICVES)*. July. Istanbul, pp. 1–6.

Lategahn, H. and Stiller, C. (2012b). "Experimente zur hochpräzisen landmarkenbasierten Eigenlokalisierung in unsicherheitsbehafteten digitalen Karten". *Workshop Fahrerassistenzsysteme (FAS)*. September. Walting, pp. 39–46.

Lefèvre, S., Laugier, C., and Ibañez-Guzmán, J. (2011). "Exploiting Map Information for Driver Intention Estimation at Road Intersections". *IEEE Intelligent Vehicles Symposium (IV)*, pp. 583–588.

Lothe, P., Bourgeois, S., Royer, E., Dhome, M., and Naudet-Collette, S. (2010). "Real-time vehicle global localisation with a single camera in dense urban areas: Exploitation of coarse 3D city models". *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 863–870.

Lowe, D. G. (2004). "Distinctive image features from scale-invariant keypoints". *International Journal of Computer Vision (IJCV)* 60.2, pp. 91–110.

Ma, J., Liu, W., Dubois, D., and Prade, H. (2011). "Bridging Jeffrey's rule, AGM revision and Dempster conditioning in the theory of evidence". *International Journal of Artificial Intelligence Tools*, pp. 1–29.

Mahler, R. P. S. (2007). *Statistical Multisource-Multitarget Information Fusion*. Boston, MA: Artech House.

Mandel, C. and Laue, T. (2010). "Particle Filter-based Position Estimation in Road Networks using Digital Elevation Models". *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Taipei: IEEE, pp. 5744–5749.

Mendes de Farias, T. (2013). "Utilisation des données Velodyne pour se positionner dans une carte". M.Sc. thesis. Université de Technologie de Compiègne.

Mercier, D., Denœux, T., and Masson, M.-H. (2006). "Refined sensor tuning in the belief function framework using contextual discounting". *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU)*. Paris, France, pp. 1443–1450.

Mercier, D., Denœux, T., and Masson, M.-H. (2010). "Belief function correction mechanisms". *Studies in Fuzziness and Soft Computing* 249, pp. 203–222.

Mercier, D., Lefèvre, É., and Delmotte, F. (2012). "Belief Functions Contextual Discounting and Canonical Decompositions". *International Journal of Approximate Reasoning (IJAR)* 53.2, pp. 146–158.

Mercier, D., Quost, B., and Denœux, T. (2005). "Contextual discounting of belief functions". *European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU)*. Ed. by L. Godo. Barcelona: Springer-Verlag, pp. 552–562.

Mercier, D., Quost, B., and Denœux, T. (2006). "Refined modeling of sensor reliability in the belief function framework using contextual discounting". *Information Fusion* 9.1, pp. 246–258.

Merriam-Webster Dictionary.

Microsoft (2014). *Bing Maps: maps.bing.com*.

Miranda, E. (2008). "A survey of the theory of coherent lower previsions". *International Journal of Approximate Reasoning (IJAR)* 48.2, pp. 628–658.

Montemerlo, M., Becker, J., Bhat, S., Dahlkamp, H., Dolgov, D., Ettinger, S., and Haehnel, D. (2008). "Junior: The Stanford Entry in the Urban Challenge". *Journal of Field Robotics* 25.9, pp. 569–597.

Moras, J. (2013). "Grilles de perception évidentielles pour la navigation robotique en milieu urbain". Ph.D. thesis. Université de Technologie de Compiègne.

Moras, J., Cherfaoui, V., and Bonnifait, P. (2010). "A lidar Perception Scheme for Intelligent Vehicle Navigation". *International Conference on Control, Automation, Robotics and Vision*. Singapore: IEEE, pp. 1809–1814.

Moras, J., Cherfaoui, V., and Bonnifait, P. (2011a). "Credibilist Occupancy Grids for Vehicle Perception in Dynamic

Environments". *IEEE International Conference on Robotics and Automation (ICRA)*. Shanghai, pp. 84–89.

Moras, J., Cherfaoui, V., and Bonnifait, P. (2011b). "Moving Objects Detection by Conflict Analysis in Evidential Grids". *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1120–1125.

Moravec, H. P. (1988). "Sensor Fusion in Certainty Grids for Mobile Robots". *AI Magazine* 9.2, pp. 61–74.

Obst, M., Bauer, S., Reisdorf, P., and Wanielik, G. (2012). "Multipath detection with 3D digital maps for robust multi-constellation GNSS/INS vehicle localization in urban areas". *IEEE Intelligent Vehicles Symposium (IV)*, pp. 184–190.

Ogawa, T., Sakai, H., Suzuki, Y., Takagi, K., and Morikawa, K. (2011). "Pedestrian Detection and Tracking using in-vehicle Lidar for Automotive Application". *IEEE Intelligent Vehicles Symposium (IV)*. Iv, pp. 734–739.

Ogawa, T. and Takagi, K. (2006). "Lane Recognition Using On-vehicle LIDAR". *IEEE Intelligent Vehicles Symposium (IV)*.

Orponen, P. (1990). "Dempster's Rule of Combination is #P-complete". *Artificial Intelligence* 44, pp. 245–253.

OSM (2013). *OpenStreetMap: www.openstreetmap.org*.

Pagac, D., Nebot, E. M., and Durrant-Whyte, H. (1998). "An evidential approach to map-building for autonomous vehicles". *IEEE Transactions on Robotics and Automation* 14.4, pp. 623–629.

Paparoditis, N., Papelard, J.-P., Cannelle, B., Devaux, A., Soheilian, B., David, N., and Houzay, E. (2012). "Stereopolis II: A multi-purpose and multi-sensor 3D mobile mapping system for street visualisation and 3D metrology". *Revue française de photogrammétrie et de télédétection* 200, pp. 69–79.

Peng, C., Guiot, J., Wu, H., Jiang, H., and Luo, Y. (2011). "Integrating models with data in ecology and palaeoecology: advances towards a model–data fusion approach". *Ecology letters* 14.5, pp. 522–536.

Perrollaz, M. (2008). "Détection d'obstacles multi-capteurs supervisée par stéréovision". Ph.D. thesis. Université Pierre et Marie Curie - Paris VI.

Pichon, F., Dubois, D., and Denœux, T. (2012). "Relevance and Truthfulness in Information Correction and Fusion". *International Journal of Approximate Reasoning (IJAR)* 53.2.

Predimap project (2011).

Pretiv project (2012).

"Proceedings of the IROS 2009 3rd Workshop: Planning , Perception and Navigation for Intelligent Vehicles (PPNIV)" (2009). *IEEE International Conference on Intelligent Robots Systems (IROS)*. St Louis, MO.

Quigley, M., Gerkey, B. P., Conley, K., Faust, J., Foote, T., Leibs, J., Berger, E., Wheeler, R., and Ng, A. Y. (2009). "ROS: an open-source Robot Operating System". *IEEE International Conference on Robotics and Automation (ICRA) Workshop on Open Source Software in Robotics*. Open-Source Software workshop.

Ramasso, E., Panagiotakis, C., Rombaut, M., and Pellerin, D. (2010). "Belief Scheduler based on model failure detection in the TBM framework. Application to human activity recognition". *International Journal of Approximate Reasoning (IJAR)* 51.7, pp. 846–865.

Royer, E. (2006). "Cartographie 3D et localisation par vision monoculaire pour la navigation autonome d'un robot mobile". Ph.D. thesis. Université Blaise Pascal - Clermont II.

Rusu, R. B. and Cousins, S. "3D is here: Point Cloud Library (PCL)".

Rutherford, E. and Soddy, F. (1903). "Radioactive change". *Philosophical Magazine and Journal of Science* 5.6, pp. 576–591.

Schreier, M. and Willert, V. (2012). "Robust free space detection in occupancy grid maps by methods of image analysis and dynamic B-spline contour tracking". *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 514–521.

Schubert, J. (2011). "Conflict management in Dempster–Shafer theory using the degree of falsity". *International Journal of Approximate Reasoning (IJAR)* 52.3, pp. 449–460.

Shafer, G. R. (1976). *A Mathematical Theory of Evidence*. Princeton University Press, pp. 1–297.

Siciliano, B. and Khatib, O., eds. (2008). *Handbook of Robotics*. Springer.

Smets, P. (1994). *What is Dempster-Shafer's model?* Ed. by R. R. Yager, M. Fedrizzi, and J. Kacprzyk. John Wiley & Sons, pp. 5–34.

Smets, P. (1995). "The Canonical Decomposition of a Weighted Belief". *International Joint Conference on AI (IJCAI)*. Montréal, pp. 1896–1901.

Smets, P. (1997). "Imperfect information: Imprecision and Uncertainty". *Uncertainty Management in Information Systems*. Springer, pp. 225–254.

Smets, P. (1999). "Jeffrey's rule of conditioning generalized to belief functions". *Annual Conference on Uncertainty in Artificial Intelligence (UAI)* 93, pp. 500–505.

Smets, P. (2000). "Data fusion in the Transferable Belief Model". *International Conference on Information Fusion*. Vol. 1. Paris, pp. 21–33.

Smets, P. (2005). "Decision Making in the TBM: the Necessity of the Pignistic Transformation". *International Journal of Approximate Reasoning (IJAR)* 38.2, pp. 133–147.

Smets, P. (2008). "Belief Functions: The Disjunctive Rule of Combination and the Generalized Bayesian Theorem". *Classic Works of the Dempster-Shafer Theory of Belief Functions*. Vol. 3085, pp. 633–663.

Smets, P. and Kennes, R. (1994). "The Transferable Belief Model". *Artificial Intelligence* 66, pp. 191–234.

Soheilian, B., Paparoditis, N., and Vallet, B. (2013). "Detection and 3D reconstruction of traffic signs from multiple view color images". *ISPRS Journal of Photogrammetry and Remote Sensing* 77, pp. 1–20.

Soheilian, B., Tournaire, O., Paparoditis, N., Vallet, B., and Papelard, J.-P. (2013). "Generation of an integrated 3D city

model with visual landmarks for autonomous navigation in dense urban areas". *IEEE Intelligent Vehicles Symposium (IV)*. Gold Coast: Ieee, pp. 304–309.

Stiller, C. and Ziegler, J. (2012). "3D Perception and Planning for Self-Driving and Cooperative Automobiles". *IEEE International Multi-Conference on Systems and Devices*. March, pp. 1–7.

Takagi, K., Morikawa, K., Ogawa, T., and Saburi, M. (2006). "Road Environment Recognition Using On-vehicle LIDAR". *IEEE Intelligent Vehicles Symposium (IV)*. Tokyo, pp. 120–125.

Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. Cambridge, Massachusetts, USA: MIT Press.

Toledo-Moreo, R., Bétaille, D., Peyret, F., and Laneurit, J. (2009). "Fusing GNSS , Dead-Reckoning , and Enhanced Maps for Road Vehicle Lane-Level Navigation". *IEEE Journal of Selected Topics in Signal Processing* 3.5, pp. 798–809.

Tournaire, O. (2007). "Extraction 3D de marquages routiers à partir d'images aériennes multi-vues et quelques applications". PhD thesis.

Urmson, C., Anhalt, J., Bagnell, D., Baker, C., and Bittner, R. (2008). "Autonomous Driving in Urban Environments: Boss and the Urban Challenge". *Journal of Field Robotics* 25.8, pp. 425–466.

Velaga, N. R., Quddus, M. A., and Bristow, A. L. (2009). "Developing an enhanced weight-based topological map-matching algorithm for intelligent transport systems". *Transportation Research*. Emerging Technologies 17.6, pp. 672–683.

Wald, L. (1999). "Definitions and terms of reference in data fusion". *International Archives of Photogrammetry and Remote Sensing*. Vol. 32. Valladolid, pp. 3–4.

Wang, L., Groves, P. D., and Ziebart, M. K. (2012). "GNSS Shadow Matching: Improving Urban Positioning Accuracy Using a 3D City Model with Optimized Visibility Prediction Scoring". *International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS)*. Nashville, TN, pp. 423–437.

Wilson, N. (1991). "A Monte-Carlo Algorithm for Dempster-Shafer Belief". *Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., pp. 414–417.

Wolf, D. F. and Sukhatme, G. S. (2005). "Mobile Robot Simultaneous Localization and Mapping in Dynamic Environments". *Autonomous Robots* 19, pp. 53–65.

Wurm, K. M., Hornung, A., Bennewitz, M., Stachniss, C., and Burgard, W. (2010). "OctoMap: A Probabilistic, Flexible, and Compact 3D Map Representation for Robotic Systems". *IEEE International Conference on Robotics and Automation (ICRA) Workshop on Best Practice Algorithms in 3D Perception and Modeling for Mobile Manipulation*.

Xiao, W., Xu, S., Oude Elberink, S., and Vosselman, G. (2012). "Change detection of trees in urban areas using multi-temporal airborne lidar point clouds". *Remote Sensing of the Ocean, Sea Ice, Coastal Waters, and Large Water Regions (SPIE)*. Ed. by C. R. Bostater, S. P. Mertikas, X. Neyt, C. Nichol, D. Cowley, and J.-P. Bruyant. Vol. 8532.

Xie, J., Nashashibi, F., Parent, M., and Favrot, O. G. (2010). "A real-time robust global localization for autonomous mobile robots in large environments". *International Conference on Control Automation Robotics and Vision*, pp. 1397–1402.

Xu, P., Davoine, F., Bordes, J.-B., and Den\oe{}ux, T. (2014). "Fusion d'informations pour la compréhension de scènes". *Traitement du Signal*, pp. 9–31.

Yager, R. R. (1987). "On the Dempster-Shafer framework and new combination rules". *International Journal of Information Sciences* 41.2, pp. 93–137.

Yi, W.-J., Sarkar, O., Mathavan, S., and Saniie, J. (2014). "Wearable sensor data fusion for remote health assessment and fall detection". *IEEE International Conference on Electro/Information Technology (EIT)*.

Yoneda, K., Tehrani, H. N. N., Ogawa, T., Hukuyama, N., and Mita, S. (2014). "Lidar Scan Feature for Localization with Highly Precise 3-D Map". *IEEE Intelligent Vehicles Symposium (IV)*.

Yu, C., Cherfaoui, V., and Bonnifait, P. (2014). "An Evidential Sensor Model for Velodyne Scan Grids". *International Conference on Control, Automation, Robotics and Vision (ICARCV)*.

Yu, Y., Zhao, H., Davoine, F., Cui, J., and Zha, H. (2013). "Monocular Visual Localization using Road Structural Features". *IEEE International Conference on Intelligent Robots Systems (IROS)*.

Zadeh, L. A. (1978). "Fuzzy sets as a basis for a theory of possibility". *Fuzzy Sets and Systems* 1, pp. 3–28.

Ziegler, J., Bender, P., Schreiber, M., Lategahn, H., Strauss, T., Stiller, C., Dang, T., Franke, U., Appenrodt, N., Keller, C. G., Kaus, E., Herrtwich, R. G., Rabe, C., Pfeiffer, D., Lindner, F., Stein, F., Erbs, F., Enzweiler, M., Knoppel, C., Hipp, J., Haueis, M., Trepte, M., Brenk, C., Tamke, A., Ghanaat, M., Braun, M., Joos, A., Fritz, H., Mock, H., Hein, M., and Zeeb, E. (2014). "Making Bertha Drive—An Autonomous Journey on a Historic Route". *IEEE Intelligent Transportation Systems Magazine (ITS Mag)* 6.2, pp. 8–20.

Zinoune, C., Bonnifait, P., and Ibañez-Guzmán, J. (2012a). "A sequential test for autonomous localisation of map errors for driving assistance systems". *International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Vol. 2. 1. Anchorage, Alaska: IEEE, pp. 1377–1382.

Zinoune, C., Bonnifait, P., and Ibañez-Guzmán, J. (2012b). "Detection of missing roundabouts in maps for Driving Assistance Systems". *IEEE Intelligent Vehicles Symposium (IV)*. Vol. Passenger. Alcala de Henares: IEEE, pp. 123–128.

[This page intentionally left blank.]

Index