Continuous Verification and Safe Localization in Semantic High Definition Maps for Automated Driving

Zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften (Dr.-Ing.)

von der KIT-Fakultät für Maschinenbau
des Karlsruher Instituts für Technologie (KIT)

angenommene

Dissertation

von

M.Sc. Jan-Hendrik Pauls

Tag der mündlichen Prüfung:	16.10.2024
Hauptreferent:	Prof. Dr.-Ing. Christoph Stiller
Korreferent:	Prof. Dr.-Ing. Klaus Dietmayer

Abstract

Semantic high definition (HD) maps enable safe and comfortable automated driving by providing information that extends far beyond the sensing range and surpasses on-board processing capabilities. However, if an autonomous vehicle falsely relies on outdated map information, this can have fatal consequences.

This thesis proposes a system which enables the continuous verification of map information *ahead* of the vehicle using only on-board sensors and real-time processing. To be able to react promptly to outdated maps, but also to safely exploit the information provided by verified maps, the verification needs to achieve exceptional certainty at distances greater than a comfortable breaking distance. This requires three major advancements over the previous state of the art.

The first is a method that *detects map elements with human-like accuracy at distances of up to* 180 m. Semantic instances are extracted from camera images using a deep neural network (DNN) and fused with high-resolution lidar data to estimate parametric representations. By tailoring the representations to each semantic class of map elements, they can be estimated robustly even at large distances. At the same time, they contain sufficient details to render meaningful reprojections into camera or range images, which is necessary to actively detect changes in the map. Moreover, they enable appearance-invariant association in parameter space, which can be trivially used for fully automated mapping.

At the core of this thesis lies the second major advancement, a *novel method for data association and localization*, called probabilistic correspondence graph (PCG). It offers previously unprecedented guarantees: With merely a coarse initial position and solely using sensor data of a single time step, it provides globally probabilistically optimal association results in real time. At the same

time, it can deal with outliers, *i.e*. clutter or outdated map elements, and offers a self-assessment capable of recognizing ambiguities.

When using PCG for data association and point cloud registration on smaller and medium-sized problems like the KITTI benchmark, it outperforms the state-of-the-art. Combining PCG with the proposed highly accurate parametric detections to localize within a semantic HD map achieves an accuracy of about $2\,\mathrm{cm}$ and $0.02°$, respectively. Such accuracy was previously only achievable with orders of magnitude more storage-intensive sensor-specific localization layers. When localizing in three years old and partially outdated maps, availability decreases but the error is almost unimpaired.

PCG as multi-hypothesis data association approach also enables the third major contribution, a *continuous ternary map verification* that tracks changes and verification independently. Detections assigned to a map element are interpreted as evidence for the element's up-to-dateness. Evidence for changes can be obtained through very fast ray casting in range images. Evidence theory enables aggregation over time, the combination of potentially contradictory evidence and an elegant incorporation of occlusions as complement to changes and verification.

The overall system enables highly reliable verification of significant shares of traffic lights and signs at ranges of more than $50\,\mathrm{m}$, the comfortable breaking distance in urban environments. As an outlook, a concept is drafted which allows the propagation of verification results from physical to abstract map elements, such as traffic rules, *without* the need to infer the abstract layer online.

Beyond mere localization and map verification, this work proposes several novel metrics to evaluate detections, map, and localization results without ground truth. The self-assessed localization and map verification makes it possible to safely use HD maps as pseudo ground truth. For this purpose, only the verified up-to-date parts of a map can be used, restricted to places where localization is accurate and unambiguous enough.

Kurzfassung

Semantische hochauflösende (HD) Karten ermöglichen sicheres und komfortables automatisiertes Fahren, da sie Informationen bereitstellen, die weit über das Sichtfeld der Bordsensorik und weit über die Verarbeitungsmöglichkeiten an Bord hinausgehen. Verlässt sich ein automatisiertes Fahrzeug aber auf veraltete Karteninformationen kann das tödliche Konsequenzen haben.

Diese Arbeit stellt daher ein Verfahren vor, das die zuverlässige Verifikation von Karteninformation *vorab* ermöglicht und hierfür nur bordeigene Sensorik und echtzeitfähige Verarbeitungsschritte nutzt. Um rechtzeitig auf Änderungen reagieren zu können, aber auch um von verifizierten Karten zu profitieren, muss eine kontinuierliche Verifikation weit über den komfortablen Bremsweg hinaus, aber dennoch mit höchster Zuverlässigkeit erfolgen. Dies erfordert drei wesentliche Weiterentwicklungen des Stands der Technik.

Der erste Beitrag ist eine Methode, die *Kartenelemente in bis zu* $180\,\mathrm{m}$ *Entfernung mit einer Genauigkeit wahrnimmt*, die menschlichen Annotationen nahe kommt. Hierfür werden semantische Instanzen mit einem tiefen neuronalen Netz aus Kamerabildern extrahiert und mit hochauflösenden Lidar-Daten fusioniert um parametrische Repräsentationen zu schätzen. Die maßgeschneiderte Anpassung der Modelle an die jeweilige semantische Klasse erlaubt eine robuste Schätzung selbst in großen Entfernungen. Gleichzeitig enthalten die Repräsentationen ausreichend Details sie in Kamera- oder Lidarbilder zu projizieren, was notwendig ist, um aktiv Änderungen der Karte erkennen zu können. Darüber hinaus ermöglichen sie eine erscheinungsinvariante Assoziation im Parameterraum, die trivial für eine vollautomatisierte Kartierung genutzt werden kann.

Das Herz dieser Arbeit ist ein *neuartiges Verfahren zur Datenassoziation und Lokalisierung*, Probabilistic Correspondence Graph (PCG) genannt, das bislang

beispiellose Garantien bietet. Mit nur einer sehr groben initialen Position und nur mit den Sensordaten eines einzigen Zeitschritts ermöglicht es eine global probabilistisch optimale Assoziation in Echtzeit. Gleichzeitig kann es mit Ausreißern, bspw. Fehldetektionen oder veralteten Kartenelementen, umgehen und bietet eine Selbstschätzung, die Mehrdeutigkeiten erkennen kann.

Für die allgemeine Datenassoziation und Registrierung von Punktwolken auf kleineren und mittelgroßen Problemen wie dem KITTI-Benchmark übertrifft es den bisherigen Stand der Technik. Wenn das Verfahren genutzt wird um sich mittels hochgenauer parametrischer Detektionen in einer semantischen HD-Karte zu lokalisieren erreicht es eine Genauigkeit von ca. $2\,\mathrm{cm}$ bzw. $0.02°$. Solche Werte waren bislang nur mit um Größenordnungen speicheraufwändigeren sensorspezifischen Lokalisierungskarten möglich. Wenn drei Jahre alte und damit teils veraltete Karten zur Lokalisierung verwendet werden, nimmt die Verfügbarkeit der Lokalisierung ab, aber der Lokalisierungsfehler bleibt quasi unbeeinträchtigt.

Die Multi-Hypothesen-Datenassoziation ist auch die Grundlage des dritten Beitrages, einer *ternären evidenzbasierten kontinuierlichen Kartenverifikation*. Zuordnungen von Detektionen zu einem Kartenelement werden als Evidenz für die Aktualität interpretiert. Evidenz für Änderungen kann über ein sehr schnelles Ray Casting in Lidarbildern gewonnen werden. Ein ternäres Evidenzsystem ermöglicht eine Aggregation über die Zeit, die Kombination von potentiell widersprüchlichen Evidenzen und eine elegante Berücksichtigung von Verdeckungen als Komplement von Verifikation und Änderung.

Das Gesamtsystem ermöglicht eine höchst zuverlässige Verifikation eines großen Teils von Ampeln und Schildern in weit mehr als $50\,\mathrm{m}$ Entfernung, was als komfortabler Bremsweg im urbanen Raum angenommen werden kann. Auf dieser Basis wird ein Verfahren entworfen, das die Weitergabe von Verifikationsergebnissen von physischen auf abstrakte Elemente, wie Verkehrsregeln, ermöglicht, *ohne* die dafür normal nötige Inferenz durchzuführen.

Über die reine Lokalisierung und Verifikation von Karten hinaus werden in dieser Arbeit mehrere Metriken vorgestellt, um Detektions- und Kartenqualität sowie Lokalisierungsergebnisse auch ohne eine Ground Truth zu evaluieren.

Schließlich ermöglicht die selbstüberwachte Lokalisierung und Verifikation von Kartenelementen die sichere Verwendung von HD-Karten als Pseudo-Ground Truth für das maschinelle Lernen. Hierfür können nur die verifizierten Teile einer Karte als Pseudo-Ground Truth bereitgestellt werden, und zwar nur dort, wo eine Lokalisierung genau und eindeutig genug möglich ist.

Danksagung

Die vorliegende Arbeit entstand während meiner Tätigkeit am Institut für Mess- und Regelungstechnik (MRT) des Karlsruher Instituts für Technologie (KIT). Zunächst möchte ich mich bei Prof. Christoph Stiller für die Betreuung dieser Arbeit, das schon früh geschenkte Vertrauen und die Möglichkeit bedanken, so frei und umfassend in einem großartigen Umfeld forschen zu können. Ebenso danke ich Prof. Klaus Dietmayer für die Übernahme des Korreferats.

Diese Arbeit wäre aber auch ohne zahlreiche andere Menschen so nicht möglich gewesen. Bei Carsten Hasberg bedanke ich mich neben der inspirierenden Kooperation für die Empfehlung des MRT als Promotionsstätte.

Meinen aktuellen und ehemaligen Kollegen am MRT danke ich für die tolle Zusammenarbeit und die Vielzahl von sozialen Unternehmungen weit über das Arbeitsleben hinaus. Neben Mitgliedern meiner Forschungsgruppe, namentlich Fabian Poggenhans, Johannes Janosovits, Haohao Hu, Frank Bieder, Richard Fehler, Fabian Immel, Nils Rack und Alexander Blumberg, möchte ich mich insbesondere auch besonders bei Martin Lauer, Eike Rehder, Johannes Beck, Sven Richter, Johannes Fischer, Kevin Rösch und Nick Le Large für die zahlreichen Inspirationen, spannenden Diskussionen und das Feedback zu dieser Arbeit sowie weit darüber hinaus bedanken.

Bei den nicht-wissenschaftlichen Kollegen, vor allem Erna Nagler und Werner Paal, möchte ich mich für die oft übersehene und doch herausragende Arbeit bedanken. Ich möchte mir nicht vorstellen, wie das MRT ohne sie aussähe.

Weiterhin möchte ich mich bei meinen Hiwis und Abschlussarbeitern bedanken, insbesondere bei Jana Aberham, Mario Boxheimer, Kürsat Petek, Benjamin Schmidt und Yu Fang, deren Betreuung kleine Höhepunkte meiner Promotionszeit waren und ohne die Teile dieser Arbeit so nicht möglich gewesen wären.

Ein augenzwinkernder Dank gebührt zahlreichen Medizinstudenten, die mit audiovisuellen Beiträgen zu Fußballturnieren vor allem die harten Jahre meiner Promotionszeit zu versüßen wussten.

Von ganzem Herzen möchte ich mich bei meinen Eltern bedanken, die mich stets gefördert und mir so viel ermöglicht haben. Ohne sie wäre ich nicht der Mensch, der ich heute bin.

Mein letzter und dennoch ganz besonderer Dank gilt Annika Meyer für die ausdauernde Unterstützung, die Geduld und das Verständnis auch wenn wieder einmal ein Wochenende mit dem Schreiben eines Papers oder der Dissertation gefüllt wurde.

Karlsruhe, im März 2024 *Jan-Hendrik Pauls*

Contents

Abstract . . . . . **i**

Kurzfassung . . . . . **iii**

Danksagung . . . . . **vii**

Notation . . . . . **xv**

1 Maps as Static Images of a Dynamic World **1**
1.1 Motivation 1
1.2 Goals 2
1.3 Contributions and Outline 4

2 HAD Maps and Map Changes **7**
2.1 Related Work 8
2.1.1 HD Maps 8
2.1.2 Map Changes 10
2.2 Definitions 11
2.2.1 HAD Maps 11
2.2.2 Map Change 13
2.3 Quantitative Map Change Analysis 16
2.3.1 Aerial Imagery as Data Source 16
2.3.2 Map Change Detection 17
2.3.3 Spatial and Temporal Extent 19
2.3.4 Results 20
2.4 Qualitative Map Change Analysis 21
2.4.1 Causes of Map Changes 21

2.4.2 Correlations 23
2.5 Insights, Interpretation and Conclusion 24

3 Detecting and Mapping Semantically Tailored Parametric Landmarks 27
3.1 Foundations and Notation 30
3.2 Related Work 31
3.2.1 Map Perception 31
3.2.2 Map Element Representations 38
3.2.3 Data Association 39
3.3 Sensors, Synchronization and Calibration 40
3.3.1 2020 “Bertha” Sensor Setup 41
3.3.2 2023 “Joy” Sensor Setup 41
3.3.3 Synchronization and Calibration 42
3.3.4 Sensor Requirements and Generalization 42
3.4 Preprocessing 43
3.4.1 Semantic Detections 43
3.4.2 Lidar Odometry 45
3.4.3 Lidar Motion Compensation 46
3.5 Semantically Tailored Parametric Detections 47
3.5.1 Lidar Point Selection and Static Parallax Compensation 47
3.5.2 Lidar Point Weighting 50
3.5.3 Initial Distance Estimation and Inlier Selection . . . 51
3.5.4 Semantically Tailored Parametric Models 54
3.5.5 Parameter Estimation 55
3.5.6 Measurement Deduplication 59
3.6 Highly Accurate HD Mapping 62
3.6.1 Acausal Processing 63
3.6.2 Data Association 63
3.6.3 Robust Map Element Estimation 71
3.7 Metrics for Hyperparameter Optimization 73
3.7.1 Detection Rendering Instance IoU 74
3.7.2 Map Rendering Instance IoU 76

3.7.3 Localization Quality Estimation 78
3.7.4 Inferior Metrics 78
3.8 Hyperparameter Optimization 78
3.8.1 Dataset 80
3.8.2 Detection Hyperparameters 80
3.8.3 Map Hyperparameters 81
3.8.4 Localization Hyperparameters 82
3.9 Evaluation 82
3.9.1 Metrics 82
3.9.2 Parametric Detections 84
3.9.3 Mapping 95
3.10 Limitations 103
3.11 Conclusion and Outlook 104

4 Verifiably Optimal Probabilistic Data Association and Localization 107
4.1 Foundations and Related Work 112
4.1.1 Data Association 113
4.1.2 Registration of Oriented Objects 123
4.1.3 Highly Accurate Localization 124
4.2 Probabilistic Correspondence Graphs 126
4.2.1 Problem Definition and Goal 127
4.2.2 Inlier/Outlier Process 128
4.2.3 Probabilistic Correspondence Space 129
4.2.4 Optimal Assignments in Correspondence Space . . 134
4.2.5 On Optimality in Correspondence Space 138
4.2.6 Constructing a PCG 142
4.2.7 Optimal Association as Maximum Weight Cliques . . 143
4.2.8 Fast Maximum Weighted Clique Retrieval 146
4.2.9 Efficient PCG Construction and Approximations . . 149
4.3 Probabilistic Compatibility Distributions 150
4.3.1 Squared Euclidean Distance Differences 150
4.3.2 Euclidean Distance Differences 151
4.3.3 Joint Euclidean Angular Distance Differences . . . 152

4.4 Interpretation . . . 155
4.4.1 Pose Estimation . . . 155
4.4.2 Residual Space Evaluation . . . 157
4.4.3 Ambiguity-Awareness by Marginalization . . . 157
4.4.4 Marginal Distribution of the Ego Pose . . . 158
4.4.5 Self-Assessment . . . 160
4.5 Evaluation . . . 161
4.5.1 Simulation Results . . . 161
4.5.2 Point Cloud Registration . . . 168
4.5.3 Localization for Autonomous Vehicles . . . 170
4.6 Limitations and Discussion . . . 186
4.7 Conclusion and Outlook . . . 187

5 Ternary Evidential HD Map Verification . . . 189
5.1 Related Work . . . 191
5.1.1 Map Change Detection . . . 192
5.1.2 Map Verification . . . 194
5.1.3 Other Works . . . 195
5.2 Ternary Evidential Verification . . . 195
5.2.1 Concept . . . 196
5.2.2 Lidar Visibility via Ray Casting in Range Images . . . 198
5.2.3 Change Classifier . . . 199
5.2.4 Verification by Marginalized Association . . . 201
5.2.5 Evidential Combination . . . 202
5.3 Evaluation . . . 203
5.3.1 Spatially Consistent Change Simulation . . . 204
5.3.2 Quantitative Results . . . 205
5.3.3 Qualitative Results . . . 209
5.4 Verification Beyond the Physical Layer . . . 213
5.5 Limitations . . . 214
5.6 Conclusion . . . 215

6 Conclusion and Outlook . . . 217
6.1 Conclusion . . . 217
6.2 Outlook . . . 219

Bibliography **221**

List of Figures **279**

List of Tables **283**

Acronyms **285**

Appendix

A Inferior Metrics **291**

B Sequences and Scenarios **293**

C Additional Evaluation Results **299**
C.1 Parametric Detection Precision 299

D Binarization of Correspondence Graphs **303**

E Analytical Derivation of SEDD **305**
E.1 Distribution of Squared Distances 306
E.2 Gamma Difference Distributions 307
E.3 Correlation under Gaussian Noise 309
E.4 Distribution of *Squared* EDDs $\Delta\delta^2_{ijkl}$ 310
E.5 Alternative Distributions 311

F Hyperparameter Optimization for Data Association **313**
F.1 Point Cloud Registration 313
F.2 Data Association in HAD Maps 314

Notation

This chapter introduces the notation and symbols which are used in this thesis. In cases where a symbol has more than one meaning, the context (or a specific statement) resolves the ambiguity.

Symbols

$:=$	definition
$\equiv$	constancy, *e.g.* over time or samples
$\sim$	sample from a distribution
$\propto$	proportional to
$\propto\!\!\!\!\sim$	approximately proportional to
$\simeq$	homeomorphic to
$\rightarrow$	association of measurements to map elements
$\|\cdot\|$	Euclidean norm
$\|\cdot\|_{\Sigma}$	covariance weighted norm
$\measuredangle$	angular or minimal rotational difference

Numbers, Spaces and Indexing

$\mathbb{R}$	real numbers
$\mathbb{N}_0$	natural numbers including zero (non-negative integers)
$\mathrm{SE}(3)$	special Euclidean group in $\mathbb{R}^3$

i, j, k, l	indexing for *e.g.* objects, measurements, points
$\boldsymbol{I}$	index set

Sensor Data and Semantics

Variables related to (preprocessed) sensor data and semantic classes are denoted in Fraktur script.

$\mathfrak{l} \in \mathfrak{L}$	lidar point
$\mathfrak{i} \in \mathfrak{I}$	image
$\mathfrak{p} \in \mathfrak{P}$	pixel
$\mathfrak{d} \in \mathfrak{D}$	semantic instance detection
$\mathfrak{b} \in \mathfrak{B}$	bounding box
$\mathfrak{m} \in \mathfrak{M}$	instance mask
$\mathfrak{c} \in \mathfrak{C}$	semantic class
$\mathfrak{o} \in [0, 1]$	confidence (of a detected instance)

General Objects

$a, b \in \mathbb{R}$	scalars
$x, y, z \in \mathbb{R}$	coordinates in $\mathbb{R}^3$
$d \in \mathbb{R}$	Euclidean distance (*e.g.* of a lidar point)
$\boldsymbol{t} \in \mathbb{R}^3$	translation vector
$\boldsymbol{R} \in \mathbb{R}^{3\times3}$	rotation matrix
$T \in \boldsymbol{T} \subset \mathrm{SE}(3)$	isometric transformation or pose
τ	thresholds, usually $\tau \in \mathbb{R}$ or $\tau \in \mathbb{N}_0$
$\mathfrak{w} \in \mathbb{R}$	weight
$\zeta \in \mathbb{R}$	scale
$e \in \mathbb{R}, \boldsymbol{e} \in \mathbb{R}^n$	(measurement) error

$\xi \in \mathbb{R}$	any parameter/coordinate of a parametric representation or an $\mathrm{SE}(3)$ pose
$\varphi \in \mathbb{R}$	yaw angle
$q \in \mathbb{R}$	quotient or share, usually $\sum_i q_i = 1$
$u, v \in \mathcal{V}$	vertices in a graph
$s, t \in \mathcal{V}$	sink and source vertices
$(u,v) \in \mathcal{E}$	edge in a graph
c	cost
$\mathcal{T} \in \boldsymbol{\mathcal{T}}$	track
$\beta \in \mathbb{R}$	scalar prefactor
Ψ	normalization term
J	cost function
ρ	robust loss function

Parametric Detections and Landmarks

$\boldsymbol{d} \in \mathcal{D}$	semantic parametric detection (in global map/world frame)
$\boldsymbol{d}' \in \mathcal{D}'$	—"— in the sensor frame it has been measured
$\boldsymbol{d}^j \in \mathcal{D}^j$	—"— in sensor frame at time j
$\boldsymbol{\ell} \in \mathcal{M}$	semantic map element (landmark)
$\boldsymbol{c} \in \mathbb{R}^3$	center point
$l,w,h \in \mathbb{R}$	length, width, height
$\boldsymbol{o} \in \mathbb{R}^3$	orientation vector
$\phi \in \mathbb{R}$	orientation angle
$\boldsymbol{n} \in \mathbb{R}^3$	normal vector

Stochastic Models

$p \in [0, 1]$	probability of an event
$f \in [0, \infty)$	(possibly unnormalized) density, also called likelihood
$F \in [0, 1]$	cumulative density function (cdf)
$\ell \in \mathbb{R}$	log likelihood
σ	standard deviation or scalar noise magnitude
$\boldsymbol{\Sigma}$	covariance matrix
$\boldsymbol{W}$	noise process
$\mathcal{N}$	Gaussian normal distribution

Probabilistic Correspondence Graph

$\boldsymbol{x} \in \boldsymbol{X}, \boldsymbol{y} \in \boldsymbol{Y}$	data points with at least a position parameter in $\mathbb{R}^n$
$\theta \in \Theta$	assignment between points of two sets or to the outlier symbol $\emptyset$
$\cdot^+, \cdot_+$	inliers
$\cdot^-, \cdot_-$	outliers
$\cdot^*$	optimally estimated result
$\cdot^\star$	true value (*e.g.* ground truth)
$u, v \in \mathcal{V}$	vertices in a graph
$(u,v) \in \mathcal{E}$	edge in a graph
$\mathcal{G}$	(probabilistic correspondence) graph
$\mathcal{C}$	correspondences between two point sets
Δ	transformation invariant measurement (TIM)
$\delta \in \mathbb{R}$	Euclidean distance (between two points)
$\Delta\delta \in \mathbb{R}$	Euclidean distance difference
$\Delta\delta^2 \in \mathbb{R}$	squared Euclidean distance difference

$\Delta\delta^\circ \in \mathbb{R}^2$	joint Euclidean angular distance difference

Evidence Theory

$\mathbf{\Omega}$	frame of discernment
$\omega \in \mathbf{\Omega}$	possibility
$m : 2^{\mathbf{\Omega}} \to [0, 1]$	mass function

Procedures

`DT`	distance transform
`R`	rendering procedure

Binary Prefixes

KiB	kibibyte	2^{10} Bytes $= 1.024\,\mathrm{kB}$
MiB	mebibyte	2^{20} Bytes $\approx 1.049\,\mathrm{MB}$
GiB	gibibyte	2^{30} Bytes $\approx 1.074\,\mathrm{GB}$
TiB	tebibyte	2^{40} Bytes $\approx 1.100\,\mathrm{TB}$

For the benefit of clearer notation, notation in the appendices may deviate from the main part.

1 Maps as Static Images of a Dynamic World

When children learn to ride a bike, they often rely on training wheels until they can safely balance themselves. Similarly, most automated vehicles rely on detailed maps until even the most complex interpretation of the static environment is solved reliably at runtime. While parents and educational scientists argue whether training wheels are more likely to help or harm children, Tesla's push to supposedly discard such maps entirely results in a similar discourse regarding the use of detailed maps for autonomous driving. Like training wheels, maps can offer a false sense of safety that can have dire consequences when relying on them in case of failure, *i.e.* when being outdated.

Unfortunately, the author of this thesis is not very optimistic about the near-term breakthrough of artificial general intelligence (AGI) on board vehicles, which would render detailed maps redundant. This necessitates ways to overcome the potentially fatal failure cases of outdated maps. Hence, this thesis presents an approach to verify the map ahead of the vehicle that can then be truly trusted.

1.1 Motivation

Automated driving offers the potential to enable private transport regardless of the passengers' abilities or formal qualifications. However, to drive as comfortably and safely as the most responsible human drivers, an automated driving system needs to perceive and interpret information about the environment at high resolution and range with an extraordinary level of reliability.

Obviously, dynamic information about the environment, like the position, motion, and intention of other traffic participants, can only be perceived during runtime. This challenge for the perception modules of an automated driving system has been recognized early, resulting in various datasets, benchmarks, and challenges. In contrast, the remaining part of the world has long been assumed static, supposedly allowing to record, process, and interpret information about it offline. The result is then stored or provided in form of so-called maps.

Maps not only make it possible to infer one's own pose with respect to a common reference frame, providing so-called ego localization. In fact, the author of this thesis holds the opinion that localization should rather be viewed as enabler that allows using all other kinds of information contained in a map, turning the map into a powerful virtual sensor.

Compared to real sensors, maps offer a number of advantages. First, as maps are stored within the autonomous system or can be retrieved via broadband cellular networks, they are virtually unlimited in range and resolution. They are also independent from weather conditions, making maps arbitrarily robust. Finally, maps can contain information that has been created using data from a whole fleet, using long-running offline processing methods or even human assistance.

While maps offer all these advantages over real sensors, they come with a catch: As long as the still incredibly challenging mapping process cannot be done online, *i.e.* in real time on board a vehicle, maps will always be a somewhat static image of a continuously changing world.

1.2 Goals

As it is not expectable that maps will be inferred online even in the midterm future, one possible consequence could be to discard maps entirely. This would result in either the abandoning of current safety standards or the unavailability of autonomous driving functions. The more challenging, but also more attractive alternative is to verify if the map is still up-to-date and exploit the benefits of map information as long as it is safe to do so. This motivates this thesis

which presents the first comprehensive on-board solution to verify that map information is up-to-date and can be used safely.

Naively detecting changes and making map information available as soon as no change has been detected quickly turns out to be a dead end. In the real world, at least some parts of the map will be occluded at all times. If this constitutes a change, the map will never be usable. In the other case, when all changed parts are occluded, the map would be fully verified mistakenly. Hence, any safe and feasible solution requires a **verification that handles occlusion and verifies only visible parts of the map**.

Additionally, verifying the map as a whole does not scale to maps as they are required for automated driving, so-called highly automated driving (HAD) maps: The more details a map contains about the local environment, the likelier it is that *any* thing has changed since the most recent map update. Therefore, the second goal of this work is to **resolve verification results to individual map elements** that have been confirmed and are safe to use. This allows using the up-to-date parts of the map even in the presence of commonly occurring changes, vastly reducing requirements on map update latency ("map freshness") and increasing availability of high quality map content.

Comparing map content and detections from sensor data on the level of individual physical elements requires the vehicle to detect these elements in the first place. This makes the detection range a major limiting factor for the availability of verified maps. Hence, one needs to **detect map elements at high range**. To react comfortably, in urban environments, the detection range should be at least $50\,\mathrm{m}$.

The second elementary step for comparing map and sensor data is the association of detected and stored map elements. In the course of this work, this turned out to be a difficult chicken-and-egg problem: When the map has changed, it becomes hard to safely localize the ego vehicle within the map. At the same time, if the ego pose is unknown, it is hard to associate map elements to verify them. Since the whole verification hinges on the resolution of this key challenge, **a robust, yet verifiably safe solution of the data association problem** is required.

When map information is confirmed using data from sensors, each with limited range and partially obstructed field of view, the absence of detected errors does

not guarantee that a map element is still up-to-date. At the same time, no autonomous driving functions will use *e.g.* a traffic sign that is known to be missing. In contrast, many functions can assume that a traffic sign that is occluded by a truck is still valid. To resolve this, the goal is to **recognize changes and verification independently**: map elements can be verified, changed or unseen.

Finally, using on-board detections from a single vehicle, verifying abstract layers of a map, *e.g.* traffic rules or intersection topologies, is very hard. As behavior decision and trajectory planning modules depend on this abstract content, the last goal is to **transfer verification results from physical elements to abstract map content**.

1.3 Contributions and Outline

To accomplish the aforementioned goals, this thesis proposes a comprehensive framework for the continuous verification of HAD maps. An overview of the system is depicted in Figure 1.1.

Chapter 2 presents one of the first surveys that investigates how the mapped world changes qualitatively and quantitatively and how these changes affect the validity of maps. Its insights are crucial to design and implement a system for map verification that covers all relevant changes and is applicable to real automated vehicles.

Based on the necessity to detect map elements in large distance with high precision, Chapter 3 describes how this can be achieved by combining semantic detections from camera images with lidar point clouds. The proposed parametric representations, which are tailored to each semantic class, turn out to be an excellent balance between expressiveness and robustness of estimation. Parametric detections can also be tracked and combined effectively to fully automatically create a map which matches human annotation quality. The lack of suitable ground truth inhibits both supervised learning methods or a corresponding evaluation. Instead, Chapter 3 proposes novel weakly and self-supervised metrics for hyperparameter optimization and evaluation of the approach.

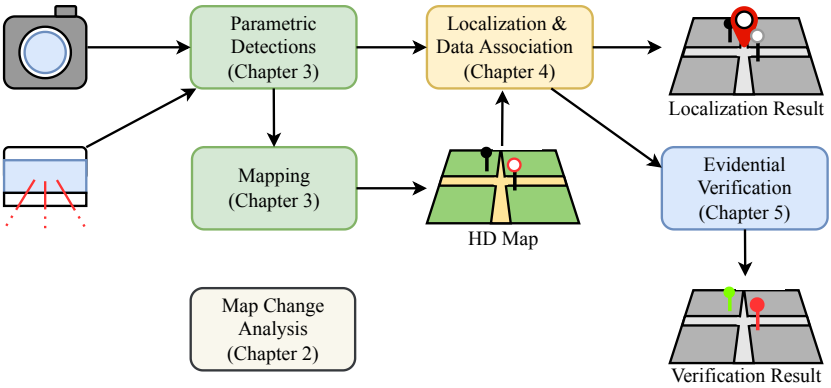


Figure 1.1: Overview of the building blocks of the proposed map verification framework. Individual chapters are color coded.

With detections and a map at hand, in Chapter 4, a solution for the core challenge of map verification is proposed: the reliable association of detections to map elements even in changed environments. It not only solves the probabilistic data association problem efficiently and globally optimally using only a single frame of measurements. It also detects ambiguities and, thus, avoids uncertain data associations. Indirectly, the proposed data association enables a safe yet highly accurate localization that works in real time. And, when applied to other association problems like point cloud registration, it is the new state of the art.

The highly reliable association results facilitate verification by marginalization over association hypothesis. In Chapter 5, this is used in a ternary aggregation scheme that tracks verification and change beliefs independently using evidence theory. Changes, *i.e.* the absence of map elements, can be actively detected using fast lidar ray casting in ordered range images. It also describes how belief can be propagated from physical to more abstract layers using a well-designed map format.

In summary, this makes it possible for the first time to verify large parts of the map far in advance of the vehicle using only on-board sensors. This enables

the integration of HAD maps not only as unrealiable prior, but as dependable virtual sensor with outstanding range, resolution, and powerful information.

2 HAD Maps, Map Changes and the Obsolescence of Map Information

From the perspective of this thesis, maps contain the seemingly static parts of the world which are relevant for automated vehicles. The question how these parts evolve over time and how this affects these maps has not been researched widely. In fact, not all changes in the real world automatically render a map outdated for all applications. For instance, a guardrail that has not been moved, but was only made safer by increasing the number of mounting poles is unchanged for any regular behavior or planning algorithm. Only functions that use this very detail, like a radar-based localization, are affected. Hence, in order to design map-based automated driving functions for the real world, one first needs to investigate how the real world evolves and which effects this has on the validity of map information.

Contributions

This chapter proposes **sound definitions for the terms *HAD maps* and *map changes***, which are used very inconsistently in literature. Based on the definition of map changes, **one of the first structured investigations on changes of the mapped world** is presented and explored both **qualitatively and quantitatively**. The quantitative part of this chapter focuses on German highways. Qualitatively, the findings are transferred from highways to analogous changes in urban settings, allowing to gather comprehensive knowledge on map changes across domains. This knowledge base not only serves as guideline to cover all possible changes when designing a localization or map verification system. It is also relevant when developing any map-based automated driving or assistance function.

Previous Publications

The main contributions of this chapter have been published previously [PSH+18, PPJ+18]. For this thesis, a previously presented map change analysis [PSH+18] is used to extract meaningful findings to design and develop a localization and map verification approach. Lanelet2 [PPJ+18] is used as example for an HAD map that is particularly well-suited to be verified.

2.1 Related Work

This chapter focuses on the basic concepts of HAD maps and their changes. Methods for map creation and the detection of map changes are discussed in Chapter 3 and Chapter 5, respectively.

2.1.1 HD Maps

The invention of high definition (HD) maps is often connected with the autonomous Bertha Benz drive in 2013 [BZS14, ZBS+14, Her18]. In contrast to previous digital maps, also referred to as standard definition (SD) maps, HD maps are supposed to contain the necessary data to safely and comfortably automate driving functions. These data need to fulfill a required level of (at least local) accuracy not only in 2D, but in 3D [LWZ20, EFH+23].

The map information is typically structured into multiple so-called layers as depicted in Figure 2.1. While SD maps contain only a coarse network on the road level, HD maps contain lane level information, usually including the precise lane course, lane boundaries and connectivity. In addition, automated functions require semantic information like traffic rules or relations between signs and lanes. Modern extensions of this map model comprise layers for behavior priors and dynamic events such as accidents [EFH+23].

Finally, to enable the use of HD maps with the required accuracy, sometimes a geo-referenced localization layer is added and referenced to the remaining layers. While early localization layers were often specific to one sensor model or at least

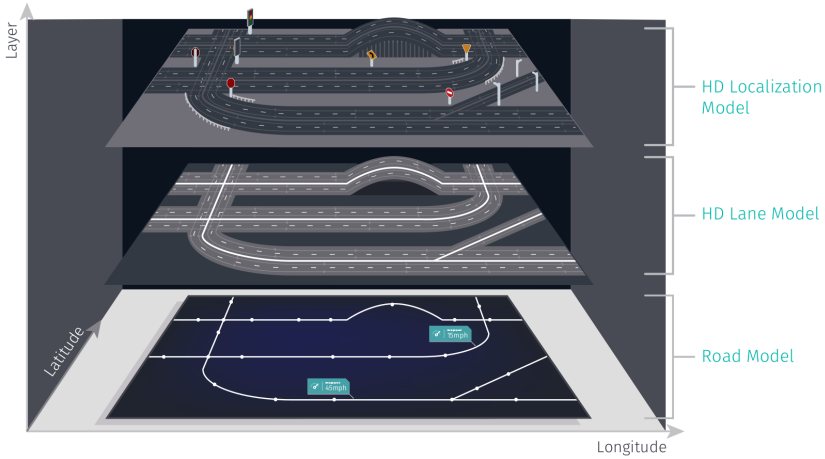


Figure 2.1: The three layers of an HD map according to map provider HERE Technologies [HER17]. While it is a prototypical depiction of HD map layers, the author disagrees with the depiction in the fact that traffic lights, road signs or guardrails are only localization features. Instead, as proposed in the Lanelet2 map framework, one should also regard them as essential semantic map elements that induce significant constraints on the behavior of automated vehicles (*cf*. Figure 2.2). Image source: [HER17].

one sensing modality, recent localization approaches attempt to use semantic map elements already contained in other layers [PPP+20]. This makes localization independent from a specific sensor and avoids referencing issues between layers.

Due to the sheer breadth of research around HD maps, the author would like to highlight the perspective of Plachetka et al. [PMF+20] on HD maps. Coining the concept of *map dependability*, they discuss a wider context of outdated maps, including errors during map creation and inconsistencies, which is worth reading. For more related works, the reader is referred to the two most common map formats [DSG10, PPJ+18], two general surveys on HD maps [LWZ20, EFH+23], an HD mapping review [WGK21], and a survey on map-based localization [CPA23]. The fact that three of the surveys assess a high degree of inconsistency around the term *HD map* [LWZ20, CPA23, EFH+23] motivates the definition of the term *HAD map* in Section 2.2.

2.1.2 Map Changes

Many works simply assume that map elements vanish or appear [JKS18], considering them as point-like landmarks in the world. In two surveys [BGB+22, EFH+23], map changes are discussed, but both focus on change detection or map update methods while map changes themselves are only described superficially. Wang and Kuhn [Wan19] discuss map changes from a functional safety perspective, but also avoid a clear definition.

One early sound definition for automotive application was given by Raaijmakers [Raa17], stating that the map dictates a valid range of values. For instance, roundabouts specify circles with a certain radius and position. A map element is then defined to be changed if and only if no measured element in the world lies within this valid range of values.

Plachetka et al. [PMF+20] also noted the lack of a clear definition. While they indeed define interesting and meaningful map deviation metrics, they hide the question about the ground truth of a change behind a not further determined data association. In a later publication, Plachetka et al. [PSF+23] give an exemplary answer using class-specific overlap criteria for the association.

The map provider HERE offers map changes via an application programming interface (API) and used to visualize them on a now seemingly broken web interface [HMC]. However, the author is aware of neither a corresponding definition for a map change, any related aggregated statistics nor qualitative insights. A rather coarse overview of changes in a collaboration with HERE is presented by Jomrich [Jom20].

Regarding the map change analysis presented in the course of this chapter, the author knows only one work with comparably detailed results, which is by Plachetka et al. [PMF+20]. Coining the term “map deviation”, they compared data over $127\,\mathrm{km}$ of urban road using a map from 2017 and measurement drives from 2019. Being surprised themselves, they did not notice a single persistent change, but only temporary modifications on $8.8\,\%$ of the road length.

2.2 Definitions

In order to frame this work and properly distinguish it from previous works, first, two essential terms are defined: *highly automated driving (HAD) maps* and changes in the world that render such a map outdated, called *map changes*.

2.2.1 HAD Maps

While the terms *map* and *HD map* have already been used in this thesis, they lack a proper definition. This is not only relevant for laymen, but the research community is missing proper definitions as well [LWZ20, CPA23, EFH+23]. The fact that a wide range of scientific publications use the term *HD map* in all kinds of meanings combined with various available products by map suppliers motivates a proper definition to work with.

Years of research, not only on mapping and localization, but on a comprehensive software stack for automated driving at the Institute of Measurement and Control Systems (MRT) have lead to the proposal of a unified holistic map that fulfills the needs of the full extent of automated driving functions. This map format and framework, called Lanelet2 [PPJ+18], has since been widely adopted in the field of autonomous driving research.

To contrast with the vaguely use term *HD map*, the corresponding publication also proposed the term *HAD map*, which serves as foundation of the following definition targeting the requirements of automation levels 3 and 4 of the SAE J3016 standard [SAE21]. For linguistic variety, within this thesis, the term *semantic HD map* will be used synonymously.

Definition 2.1: Highly Automated Driving (HAD) Map

An highly automated driving (HAD) map is a collection of information that fulfills the following conditions:

1 **Unified content:** An HAD map is a unified representation of information shared across applications / automated driving functions, including but not

restricted to routing, behavior generation, object prediction, trajectory planning, special maneuvers, scene understanding, algorithm validation, sensor simulation, localization, and map verification. Any kind of information that meets the conditions specified below required by any application is contained in the HAD map to ensure consistency.

2 **Spatial accuracy and resolution:** The information is available at a resolution and local accuracy to safely associate measurements to map content[a]. If information is split into multiple layers, they are referenced to each other with similar accuracy. This allows using map elements consistently across layers. Global accuracy is subordinate and only needs to be accurate enough to facilitate the use of GNSS localization methods as spatial prior[b]. The same resolution and accuracy is required across all three spatial dimensions and possibly even extent or orientation parameters.

3 **Spatial and temporal consistency:** Within this spatial accuracy, the information is valid for a time span that comprises the typical interval between two separate drives. This differentiates an HAD map from *e.g.* occupancy grid maps. The two drives are not required to happen with the same vehicle, allowing to include road works, blockages or accidents.

4 **Semantics:** The information contained in the map has a semantic meaning and logical interconnections. In particular, this excludes abstract sensor-specific feature layers as used for localization.

5 **Physical grounding:** If possible[c], abstract information needs to be grounded in physical elements to obtain a replicable and verifiable map. While the direct annotation of abstract information can be a valid, *e.g.* for faster processing, their provenance needs to be comprehensible from existent physical map elements. This is key not only for map verification, but also to correctly fuse sensor data with map information.

[a] Less than 10 cm with current technology.
[b] Less than 1 m with current technology.
[c] The only valid exception known to the author are typically driven, but unmarked "virtual" lane boundaries within intersections.

One example of such an HAD map is the open source map format and framework Lanelet2 [PPJ+18] to which the author contributed significantly. As depicted in

Figure 2.2, Lanelet2 is defined bottom-up, starting with physical elements, such as road boundaries, road markings, traffic signs, and traffic lights. These are then connected with so-called relations to form lane sections, the eponymous lanelets, which are in turn consolidated to lanes and/or connected with traffic signs to induce traffic rules. Traffic lights as well as stop or yield signs are semantically linked to the respective stop line. Successive lanelets share identical end and start points while neighbors emerge by sharing a common lane boundary. Together, they form a topological routing graph which constitutes the most abstract layer of map content. This pervasive semantic connection from physical elements up to all derived map content is not an unremarkable detail, but a crucial design feature of Lanelet2. And, as shown in Chapter 5, it actually enables the verifiability of higher level map content in the first place.

2.2.2 Map Change

The second necessary definition is that of a *map change*. While the map itself does not change, but the mapped world is changing, in this thesis this term is used as abbreviation for *map-relevant changes*. The often used term *road*

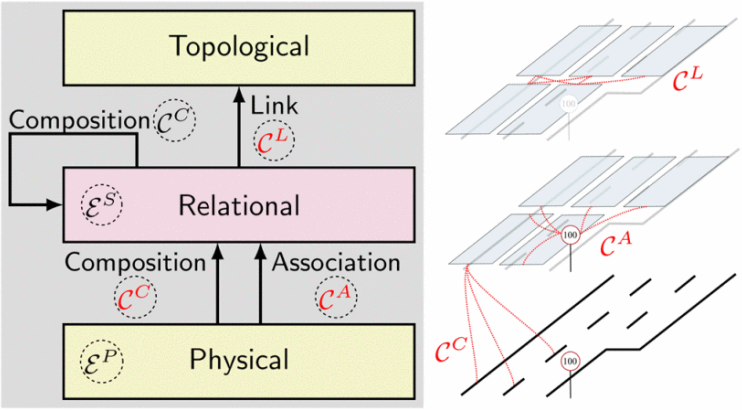


Figure 2.2: The three layers of a Lanelet2 map with the physical and relational map elements, $\mathcal{E}^P$ and $\mathcal{E}^S$, respectively. To put them into relation, Plachetka et al. [PMF+20] formally specified compositions $\mathcal{C}^C$, associations $\mathcal{C}^A$, and links $\mathcal{C}^L$. Image source: [PMF+20].

change, also used by the author in previous publications, is misleading since HAD maps contain much more than just roads and road elements.

At talks and conferences, the definition of a map change often raised questions and caused intensive discussions since most people supposedly have an intuitive understanding of map changes. The common idea is that any humanly perceivable change would make a map outdated. However, for some changes, this is not the case for some or even all automated driving functions.

For this work and especially its investigation of map changes, a precise definition is required. It needs to take into account the requirements of current and future automated driving functions. At the same time, instead of relying on the ephemeral technical state of the art, it uses human interpretations as reference. This human interpretation of the mapped world is called "annotation", assuming a trained human map annotator would build a map of the above definition using arbitrary sensor data.

As proposed by Jo et al. [JKS18], this thesis avoids the idea of "moved" map elements, but only considers their creation, persistence, and removal. This is not only simpler from a formal point of view. Also in practice, in most cases, map elements are not moved, but outdated road markings, poles, traffic signs, and traffic lights are removed and new map elements are put in the desired new place.

Definition 2.2: Map Change

A physical map element ℓ in a map $\mathcal{M}$ is outdated if and only if a map $\mathcal{M}'$ that was annotated using up-to-date sensor data and sufficient visibility, *i.e.* ℓ is not occluded, would contain no element ℓ' that is similar with statistical significance, *i.e.* excluding typical annotation noise:

$$\ell \text{ outdated} \Leftrightarrow \nexists\, \ell' \in \mathcal{M}' : \ell' \approx \ell \tag{2.1}$$

If even one map element is outdated according to this definition, the map is considered to have changed and a map change is defined to have been occurred.

This definition offers two major advantages. First, it builds upon human annotations, which are prohibitively expensive, but still the gold standard of mapping. This allows decoupling the definition from a concrete technical method. Implicitly, this assumes that machine learning as well as any other methods asymptotically approach human performance which serves as fallback and quality control – at the moment and for the foreseeable future. One method to test statistical similarity w.r.t. human annotation noise would be the two one sided tests (TOST) procedure [Sch87].

Second, defining a change in the (annotated) element space implicitly excludes any changes that can be perceived by humans or even technical systems, but do not affect the map content. For instance, a traffic sign that has been replaced with an identical but new sign that is mounted almost identically should not constitute a relevant change – at least unless the age of signs is stored in the map.

For the scope of this thesis, the definition of changes is limited to physical map elements. It also explicitly excludes new elements since the addition and incorporation of additional map elements is a research topic on its own that has barely been explored.

Due to the physical grounding of abstract map content required in Definition 2.1, the absence of physical changes induces the absence of abstract changes. This holds in particular if no new map elements have appeared. Based on this idea, Chapter 5 presents how verification results can be propagated from the physical to abstract layers. One exception which would contradict the concept of physical grounding are breaking changes of traffic rules. However, due to the necessary reeducation of human drivers, those are extremely rare to non-existent.

For an idea how specific changes of the abstract layers could be defined and measured, the reader is referred to Plachetka et al. [PMF+20]. Unless a comprehensive and detailed empirical map change analysis for abstract layers is available, the author refrains from proposing a definition other than this existing one.

2.3 Quantitative Map Change Analysis

With this definition of a map change one can perform a quantitative analysis of how they occur in the real world. The analysis presented in this chapter consists of three steps: data collection, spatial referencing of the data before and after possible changes, and the actual detection of changes. This section shows how accurately georeferenced high resolution aerial imagery facilitates the first two steps. Changes can be extracted by comparing superimposed imagery from different years.

2.3.1 Aerial Imagery as Data Source

The use of aerial imagery as a data source for map changes may seem surprising at first since the Institute of Measurement and Control Systems (MRT) maintains several vehicles equipped with the latest research sensor technology. The idea of using measurement vehicles to record data for change analysis was actually tried on several highway sections before and after construction or maintenance works. However, it was quickly found that comprehensive road surface changes and other extensive road works would alter any features that could be used to reference drives before and after the change. Thus, feature-based referencing is very challenging or even impossible not only in fully changed parts, but especially in the interesting transitions between unchanged and changed environments. Referencing using an RTK-GNSS/INS[1] unit was attempted but found to be too inaccurate to resolve small changes with certainty.

In contrast to measurement drives, aerial imagery also covers areas next to the highway, which allows pixel-accurate referencing of aerial images even across years or complete reconstructions, as illustrated in Figure 2.3. This suggests the use of aerial imagery for map change detection.

However, easy referencing is not the only advantage of aerial imagery over data from measurement vehicles. Aerial imagery also enables change detection in

[1] Real-time kinematic (RTK) - global navigation satellite system (GNSS) / inertial navigation system (INS)

retrospect. While it is possible to request planned changes from the responsible institutions, construction measures typically take months or even years. Not only might this be longer than the typical duration of a PhD; setting up and maintaining a comparable sensor and software setup before and after the change is not compatible with the other ongoing research at the institute. Aerial imagery, on the other hand, is widely available from archives for even more than a decade. Imagery has consistently high resolution and, for the most parts, is referenced pixel-accurately. This advances the temporal scope far into the past.

A ground sample distance (GSD), *i.e.* the size of each pixel side, of around $5\,\mathrm{cm}$ puts the resolution limits of aerial imagery into the same order of magnitude as data recorded from measurement vehicles. Hence, aerial images are a valid substitute for onboard sensor data that has been consistently recorded over multiple years and aligned with highest accuracy.

2.3.2 Map Change Detection

To detect map changes, aerial imagery, acquired from the cites of Karlsruhe and Erlangen, was used. While Karlsruhe requires financial compensation, the city of Erlangen is among the few leading cities that make high resolution imagery publicly available across multiple years, thus, enabling large-scale research.

Student assistants and the author extracted map data and changes using open source software for annotation [JOSM] and provisioning aerial imagery [GeoServer]. Road markings and guardrails were selected as map features since they are clearly visible in bird's eye view (BEV). The extraction of those features was done separately from a detection of changes, which directly used the imagery. This offers two data streams whose consistency can be verified.

The detected changes follow Definition 2.2, but spatially contiguous changes were consolidated to a single change with a length of change attribute. For instance, if eight successive markings have changes over $144\,\mathrm{m}$, this is a single change. Knowing the total length of the road and the number of distinct features, one can understand the relative share of changes.

Figure 2.3: Exemplary superimposition of aerial imagery from 2015 (bottom left) and 2017 (top right). While the highway has been changed entirely and even was widened, the unchanged parts next to the highway still allow pixel-accurate geo-referencing. Aerial Imagery: © Stadt Karlsruhe | Liegenschaftsamt

Finally, changes were separated by semantic features and extent. The category *guardrail(s) changed* means that for the changed part only one or the guardrails on both sides have been changed, but no marking. Conversely, *marking(s) changed* indicates that only one or more markings have been changed, but no guardrail. *Both changed* describes that a mix of guardrails and markings have been changed, with some elements remaining unaltered. This is interesting since if *e.g.* the markings that form the right lane are unchanged, an autonomous vehicle could – although probably with degraded velocity and enhanced caution – still use the right lane with map support. Finally, *full reconstruction* means that all features have changed. This is the usual case when the whole pavement was removed and renewed.

(a) Map changes on A3 and A73 around Erlangen

(b) Map changes on A5 near Karlsruhe

Figure 2.4: Illustration of the Observed Map Changes. The changes are colored as follows: white: no change, blue: guardrail(s) changed, yellow: marking(s) changed, orange: both changed, red: full reconstruction.
Aerial Imagery: © Esri, DigitalGlobe, GeoEye, Earthstar Geographics, and the GIS User Community

2.3.3 Spatial and Temporal Extent

Spatially, the analysis investigates six sections of three German highways covering all lanes of each section. The sections of A3 and A73 around Erlangen as well as the sections of A5 near Karlsruhe are depicted in Figure 2.4.

In total, the sections have a length of $79.5\,\mathrm{km}$. With two to three lanes, they are representative examples of German highways.

Regarding the temporal extent, aerial imagery from 2011 and 2016 were compared for the highways A3 and A73 around Erlangen. The exact date of the images is unknown. For the A5 near Karlsruhe, the aerial images have been taken in 2015 and 2017 at the end of the foliage season, *i.e.* late winter.

2.3.4 Results

The detected changes by absolute numbers are presented in Table 2.1a. Assuming two and five full years, respectively, one can also calculate relative change rates as presented in Table 2.1b. Figure 2.4 illustrates the detected changes visually.

In total, most changes affect markings, but this is not consistent across all highways. Hence, with the presented dataset, change frequency cannot be predicted by the semantic type of map elements.

Considering relative changes, the annual change rate of 16% is surprisingly close to the 15% published by the mapping company TomTom [Tom18].

Table 2.1: Length and share of map changes between the years 2011 and 2015 (A3, A73) as well as 2015 and 2017 (A5), respectively.

Result \ Highway	**A3**	**A73**	**A5**	Σ
No change	25.6	13.7	7.6	46.9
Guardrail(s) changed	2.6	0.8	0.6	4.0
Marking(s) changed	1.6	2.9	7.1	11.6
Both changed	0.4	2.4	5.8	8.6
Full reconstruction	2.4	-	5.9	8.3
Total changed length	7.0	6.1	19.4	32.5
Total length	32.6	19.8	27.0	79.4

(a) Results in absolute numbers (km).

Result \ Highway	**A3**	**A73**	**A5**	∅	**∅ p.a.**
No change	79	69	28	59	
Guardrail(s) changed	8	4	2	5	1.2
Marking(s) changed	5	15	26	15	5.6
Both changed	1	12	22	11	4.4
Full reconstruction	7	-	22	10	4.3
Total changed length	22	31	72	41	15.5
Change per year	4	6	36	16	

(b) Results in relative numbers (%). The right column contains the mean across all sections weighted by temporal distance assuming two and five full years, respectively.

Possibly more interesting than the average is that the distribution ranges from 4% to 36% annual changes. One can also divide the changes into partial and complete changes since for partial changes, the unchanged parts could be used at least for degraded functionality. About 11% of highways undergo partial changes in a year while 4% of the road length are renewed completely.

2.4 Qualitative Map Change Analysis

The quantitative change analysis provides a foundation to parameterize map-based assistance or automated driving functions such as the map verification itself. However, it does not offer a deeper understanding of the reasons and patterns that are behind these bare figures.

The goal of the qualitative map change analysis is to characterize the most relevant causes for map changes as this not only allows understanding the processes that render a map outdated. It also enables a discussion about expected failure cases for maps as virtual sensors when considered from a functional safety perspective. While the quantitative analysis was limited to highways, now also changes of urban or rural environments are included.

2.4.1 Causes of Map Changes

This investigation identified four main reasons for map changes that need to be taken into account when designing map-based functions.

Design Changes

Especially in urban and rural environments, most changes are due to design changes. This means that new roads are constructed or existing roads are moved or redesigned. While most design changes are due to changes in the world surrounding the roads, such changes often come along with safety-relevant changes like better pedestrian crossings or newly integrated bicycle lanes.

A special case of design changes is to replace intersections with roundabouts. This has already been noticed and led to a number of previous works that specialized in these very changes [ZBI12b, RB14, RB15, Raa17].

Safety-relevant Changes

Especially on highways most changes do not alter the course of the road, but intend to improve the safety of the traffic participants. For instance, due to stricter norms, old guardrails are replaced with newer ones with better restraint efficiency. For the same reasons, metal guardrails are replaced with Jersey barriers made from concrete. These changes are not distributed uniformly, but occur with higher frequency around bridges or similar places that necessitate an increased level of safety.

This category also comprises the replacement of static traffic signs with dynamic traffic control systems that not only improve traffic flow, but can also increase safety by lowering the speed limit in denser traffic. Like new roundabouts, this particular change has been covered in a specialized publication [NGZ09].

Deterioration, Weathering, Accidents, and Effects of Severe Weather

While design and safety-relevant changes are desired and intentional, the world is also changing unintentionally. This is the case for the deterioration due use, especially by heavy traffic, accidents, and severe weather. Accidents and meteorological events happen suddenly and surprisingly. At the same time, effects of both are usually spatially very limited.

On the contrary, deterioration and weathering is happening over the years, but over a much larger spatial extent. However, over time, the deteriorating quality of markings or readability of signs can be detected and taken into account by either robust systems or a possible automated feedback to the administrative bodies responsible for their maintenance.

Maintenance Works

The undesirability of the previous category of reasons why maps change directly leads to their counterpart. Maintenance work repaints markings and replaces old signs or dented guardrails. Some maintenance work is limited to very few meters of damaged guardrail and can be tolerated easily by the robustness of almost any real-world driving function. Full reconstructions of the pavement, on the contrary, render all map content for a lengthy stretch of the road outdated.

As maintenance only happens after a certain degree of deterioration, tracking the quality of map elements could also be used to predict maintenance work.

A very important insight is that ongoing road works are the easy case. Due to warning signs, colored markings, as well as red and white beacons, they are easy to detect and methods to incorporate them into the road geometry have already been proposed early on [DRS+15].

The actual challenge are unnoticed maintenance works, *e.g.* happening over night, during road closures or in little used roads. When the road works are done, all visual warning signs might have been removed already. Hence, the first vehicle driving on such modified roads has no clue, but still needs to deal with map changes of unknown scale. In fact, this very difficulty was a major motivation for the onboard verification approach proposed in this thesis.

2.4.2 Correlations

Next to the causes themselves, the correlations between them have not yet been considered. As changes do not happen uniformly at random, but due to logical reasons, one can observe several correlations or patterns.

Spatio-semantic Correlations

For instance, almost never only a single marking is repainted, but the change of a single dash means that the successive dash has also been changed with

high probability. In urban scenarios, often all lanes are redesigned or all traffic lights in an intersection are replaced at once.

While this spatio-semantic correlation pattern seems to be apparent, it breaks with the assumption of uniformly distributed and independent change events that might be convenient for mathematical models. Instead, it is suggested to explicitly model these correlations when verifying maps and exploit them, as done in a series of previously proposed map change detection approaches [PSH+20a, PSH+20b, PSH+21].

Transitions

Another pattern can be observed at transitions between unchanged and changed areas. Probably to make transition zones safer and more comfortable, they are intentionally made smooth. This means that changes are often smaller close to unchanged areas and increase in magnitude over distance. Hence, few altered elements could serve as indicator for potentially more changes.

However, while human drivers benefit from this effort, filtering-based localization approaches run the risk of slowly drifting or degrading in performance. Especially when some parts are still unchanged, this can pose an issue for localization integrity and monitoring.

2.5 Insights, Interpretation and Conclusion

The presented map change analysis is, together with its previous publication [PSH+18], one of the first of its kind. It describes the processes that render a map outdated both quantitatively and qualitatively. The average annual change rate of about 16% is made up to two thirds by partial changes while one third are complete reconstructions. This underlines the importance of map verification procedures that can resolve map changes to the individual changed elements. Compared to naive approaches that make the whole map unavailable as soon as a change is detected, they have the potential to increase (partial) map availability by up to 200%.

In the future, the growing collection of fleet data will allow to extend the scope of similar investigations with comparatively negligible effort. However, compared to aerial imagery, their referencing is a non-trivial issue and only time will tell how many insights from private fleet data will actually be published.

In the qualitative analysis, four major reasons why maps become outdated have been identified. It also characterized the correlation patterns that are known and should be incorporated into real-world systems or pose severe challenges. Additionally, the prediction, mitigation or feedback of some of the expectable map changes has been described.

As a result, this chapter allows to improve the design and parametrization of map-based driver assistance and automated driving functions far beyond this work.

3 Detection and Mapping of Semantically Tailored Parametric Landmarks

Sensing and measuring the current state of the environment is a fundamental task of autonomous vehicles. While the necessity to perceive dynamic objects is obvious, detecting the static parts of the world is equally important for two reasons. First, GNSS based localization systems are neither accurate nor reliable enough for highly accurate ego localization. Landmarks have to be matched between map and sensed environment to achieve the necessary accuracy and reliability, especially in GNSS-impaired surroundings like cities.

In addition, as pointed out in the previous chapter, maps cannot be assumed static. For any verification of map elements or change detection, an autonomous vehicle needs to perceive the current state of the environment. In order to use the map ahead to drive comfortably and safely, in particular the map elements ahead of the vehicle need to be perceived. The larger the detection range, the earlier map elements can be verified, and the more comfortably and safely the automated vehicle can drive.

While the use of fleet data can mitigate finite sensor and detection range, this work is limited to onboard sensors. This allows implementing the proposed verification framework in next-generation series vehicles and fully use it even before fleet data is common and densely available enough to replace on-board detections.

The approach proposed in this thesis uses camera images with their high spatial resolution and color information to generate initial detections in the image domain using a deep neural network (DNN). Instead of estimating depth using one or multiple camera frames, depth measurements from a lidar sensor are

projected into the camera image. Lidar's time of flight (ToF) or frequency modulated continuous wave (FMCW) measurement principles enable distance measurements with a range-independent error. Since detections should be as uncorrelated as possible, the approach does not make use of temporally aggregated data and estimates detections using only a single time frame at once.

Finally, to demonstrate the localization and verification methods proposed in the remaining chapters of this thesis, one needs to obtain an exemplary HD map with sufficient accuracy.

Contributions

To turn the sensor data into meaningful and highly accurate detections, it proposed to use parametric representations. The key innovation and a major contribution of this chapter is to **choose a suitable parametric representation *tailored* based on the semantic class** that is predicted by the DNN. In contrast to previous, more generic representations this enables modeling the characteristics of each category of map elements so that the parameters, like position, orientation, and size, can be determined with **very high accuracy even when only a sensor data from a single time frame is used**. This vastly facilitates data association during both mapping and localization.

One focus when deriving parametric measurements is an **elaborate fusion of lidar points with camera detections**. This includes obvious, but often neglected steps like taking into account beam divergence and compensating static and motion parallax effects. Additionally, multiple ways how to weight points based on instance masks are compared and it is shown how to select inlier points that then allow the estimation of class-specific parameters.

In order to aggregate individual detections to a map, this thesis proposes to **leverage state-of-the-art object tracking algorithms to associate the detections over time**. The map can then be computed by robustly averaging all associated measurements. This allows aggregating the detections fully automatically to create an exemplary HD map that has similar annotation quality to parametric annotations by humans. Hence, the resulting map not only enables safe

localization and map verification as described in the remainder of this thesis. Without using it further in this work, the proposed technique also **enables to create a map in ground truth quality for machine learning purposes** with minimal human intervention.

Finally, this chapter proposes a family of **novel metrics for parametric detections, the estimated map, and ego localization** using semantic instances as pseudo-labels. It allows comparing the various proposed building blocks and **enables fully automatic hyperparameter tuning in a weakly/self-supervised manner**, *i.e.* without requiring manually annotated ground truth, which is particularly expensive and tedious for parametric map elements.

All building blocks of this chapter are illustrated in Figure 3.1.

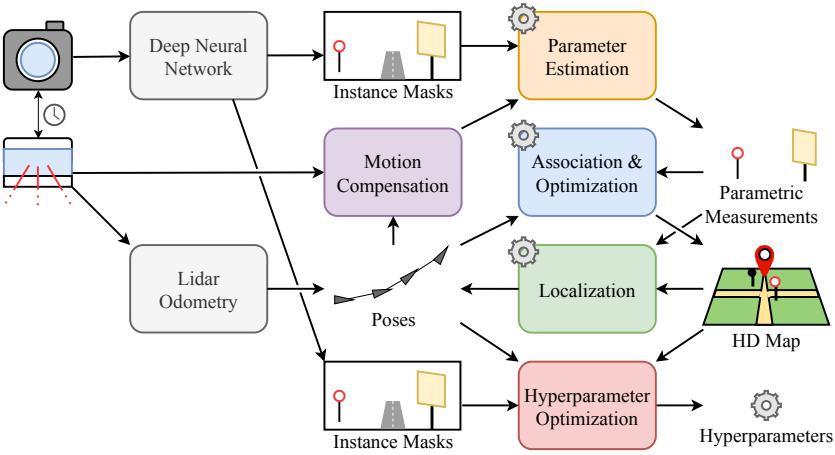


Figure 3.1: Overview of the building blocks of the proposed parametric mapping module. Gray blocks are existing methods that are used for this work. Colored blocks are contributions of this thesis or previous publications by the author.

Previous Publications

The main ideas for parametric detections and corresponding mapping have been developed or first implemented during the student theses of Benjamin Schmidt [Sch20] and Yu Fang [Fan21], both supervised by the author.

Core concepts of this chapter have previously been published in [PSS21, PS22], but the author since refined the detection procedure significantly. This includes changing the temporal data association during mapping to a more advanced multi-stage graph-based method from object tracking and adding the weakly/self-supervised metrics and hyperparameter optimization.

3.1 Foundations and Notation

Since most readers are assumed to be familiar with the fundamental concepts of this work, the reader is referred to the following works which contain the relevant fundamentals. Projective geometry, such as camera models and homogeneous coordinates, are described in detail by Beyerer et al. [BPF12]. A broader overview of the field, including the principles of 3D reconstruction, is given by Hartley and Zisserman [HZ04]. The reader is also referred to their appendix for quaternions and singular value decomposition (SVD), including as method for principal component analysis (PCA). The idea of the simultaneous localization and mapping (SLAM) problem is described by Thrun et al. [TBF05].

Rigid body motions described by isometric transformations T form a Lie group and only locally behave like "ordinary" Euclidean space. The foundations and implications of this fact are covered by Selig [Sel05].

In this work, the application of a transformation T as isometric operator is denoted by $\boldsymbol{x} = T\boldsymbol{y}$, acting as multiplication on homogeneous coordinates. The composition of transformations is a standard operation within the $\mathrm{SE}(3)$ Lie group and denoted by $T = T_1 \oplus T_2$.

As final notational remark, detections $\boldsymbol{d}$ and map elements $\boldsymbol{\ell}$ are assumed to be elements from a joint space that unifies an $\mathrm{SE}(3)$ pose, two real-valued extent

parameters, and a semantic class. To simplify the notation, for mathematical operations on $\boldsymbol{d}$ or $\boldsymbol{\ell}$, the semantic class is neglected.

3.2 Related Work

While the proposed approach might be transferable to other domains like indoor robotics, the focus of this work lies on sensing and creating semantic HD maps as they are used for automated driving.

3.2.1 Map Perception

If annotated ground truth is available and performance is paramount, today machine learning (ML) approaches are basically the only viable option. Unfortunately, as explained below, there are no suitable datasets with lidar and/or camera information available that contain more than one class of elements of an HD map in 3D space.

Most modern map perception approaches are not specialized to one specific class. Instead, related methods, both handcrafted and learned, are distinguished by sensing modality. These are – with few exceptions – camera, lidar or a fusion of both.

Datasets

Early datasets with semantic annotations of the static world have been recorded using mobile laser scanning systems, *e.g.* [RDG18]. Unfortunately, as neither camera nor suitable, *i.e.* single automotive/onboard, lidar scans are available, they are unsuitable for learning online perception of the map ahead. Additionally, they are very limited in spatial extent having lengths of at most $2\,\mathrm{km}$. Also, they usually annotate semantic classes per lidar point instead of more abstract object representations.

This is similar to the task of 3D semantic segmentation, for which there are a number of datasets with onboard sensor data, sufficient spatial scope, like

SemanticKITTI [BGM+19] or Panoptic NuScenes [FMH+22]. Some of them even contain suitable semantic classes, like the Waymo Open Dataset [Way22] or KITTI-360 [LXG23]. However, by annotating each point instead of parametric objects, they leave the non-trivial problem of deriving meaningful parametric representations from object point clouds.

This is solved in datasets, such as KITTI-360 [LXG23], which offer semantic object-level annotations in 3D. KITTI-360 was seriously considered for this work, but discarded due to the outdated lidar sensors which were not promising with regard to their limited efficient detection range.

Using a very similar, oriented object representation as the approach proposed in this thesis and poles, signs, and traffic lights as map elements, Plachetka et al. [PSF+22] recently published 3DHD CityScenes, a HD map element dataset including aggregated lidar point cloud tiles. While recorded using a mobile laser scanning system, they provide virtual scans that are supposed to simulate data from a multi-layer lidar. Unfortunately, this otherwise very suitable dataset has two drawbacks. Not only are camera images missing which facilitate detections in large distances. It is dubious that the virtual scans with their uniform point density allow to generalize to single scans from multi-layer automotive lidars whose point density strongly decreases with distance. Being released so recently and still having this significant domain gap eliminated the dataset for this work.

A whole other category are datasets which annotate objects in 2D images, either as bounding boxes or with pixelwise instance masks. Early publications, like COCO [LMB+14] or Cityscapes [COR+16], focused on a very wide, but sparse range of categories or mostly dynamic objects. More recently, semantically comprehensive datasets with a (partial) focus on map elements, like Mapillary Vistas [NOB+17], have been published. Also, 2D annotations of elements like traffic lights have been added to existing datasets [Jan22, WLL+23]. Regarding datasets focusing on individual categories of map elements, the reader is referred to [FMK+18, FGS+18] for traffic lights and [LLC+19, JGB+20] for road signs.

One eventual goal of this work is the evaluation on data that contains map changes, *i.e.* recorded several months or even years apart. Unfortunately, camera and lidar datasets with sufficient temporal scope, like the Oxford RobotCar

dataset [MPL+17] or Boreas [BYW+23], contain no maps or map annotations for training while those datasets with maps or suitable annotations have no sufficient temporal scope. Training on a public dataset with map annotations and deploying on a multi-season dataset would lead to a domain gap that is assumed to be no better than using data recorded with measurement vehicles as they are available at the MRT.

In the course of a student thesis supervised by the author [Feh21] and a joint publication [PFL+22], this domain gap was partially closed by combining Mapillary Vistas as diverse 2D dataset with object-level depth estimation trained using lidar measurements from KITTI as target domain. However, extending the results beyond coarse depth, as required to perceive parametric map elements, is not trivial.

Finally, annotating ground truth with all parameters in sensor data is possible, but extremely time-consuming. For a small sample which is only sufficient for evaluation, but not training, this option was tried as published in [PSS21]. In Section 3.9 this result serves as baseline.

To summarize, there is a lack of suitable datasets that prevents training a deeply learned approach that fuses camera and lidar data. Instead, this thesis proposes a (late) fusion method that combines 2D object detections from a DNN trained on a well-generalizing dataset with raw lidar data. However, in the future, the maps that result from this thesis could serve as pseudo-ground truth for training a fusion DNN. Since the goal of this chapter is to show how a suitable map can be created and perceived with very high accuracy, it was deemed that implementing such an additional method was beyond the scope of this thesis.

Camera-based Map Perception

Sensing semantic map content using cameras has been done since the early days of HD maps, firstly with the goal of localization. The key challenge for camera-based perception is how to obtain 3D map elements from 2D measurements. Road markings are probably the most common elements and can be mapped using so-called inverse perspective mapping [SKF13, ZBS+14]. Like other

works reviewed in this section, this thesis uses poles, traffic lights, and signs as exemplary map elements since they have the advantage of being visible from farther away.

Poles only rarely have a regulatory meaning, but are commonly used as landmarks for localization. Thus, they are missing a clear definition of their semantic class. In contrast, road markings, traffic signs and traffic lights have a clearly defined multitude of semantic classes. This leads to the differentiation between the detection of their existence, usually including a location, and the fine-grained extraction of their semantic class(es). The proposed approach definitely profits from semantic diversity in order to properly associate map elements correctly, *i.e.* more specific sign classes or traffic light types are beneficial as long as they can be distinguished correctly. However, the focus of this work lies on the detection of signs and traffic lights, taking their semantic classes for granted.

Since traffic lights are static in position and shape, but their light state is dynamic, they have been an early focus topic for camera-based perception. For surveys of traditional, mostly handcrafted methods for their detection, the reader is referred to [JPM+16, FGS+18]. Modern approaches use DNNs to detect traffic lights in camera images [WWZ16, BNB17, BSD18, MD18, Jan22, PLB+23]. In their implementation, they are similar to generic object detection approaches, but differ in solving both detection and classification of the traffic light state in a single step, *e.g.* using hierarchical classes, or a joint, data-driven or learned pipeline.

To map traffic lights in 3D space, many approaches use frame-to-frame association combined with triangulation [FU11, LAD+11]. One way to obtain initial estimates of the distance is by assuming knowledge of the size of lights [FU11] or their height [DCS12] as specified in norms, laws or guidelines. Another common method uses multiple cameras, typically in the form of stereo vision [FMD17, MFD17, FMK+18]. They all share the problem of an error that grows quadratically with the true distance. Still, previous works were able to demonstrate that traffic lights can serve as landmarks to improve vehicle localization [WHJ+19].

Traffic sign detection is a known problem from advanced driver assistance systems (ADAS). As for traffic lights, state-of-the-art approaches use deeply learned

methods, typically inspired by 2D object detection frameworks. For an overview of the general field, the reader is referred to two surveys [LLC+19, JGB+20].

Just like traffic lights, roads signs from an a priori map can be used to improve vehicle localization [LNT10, WRW15]. One of the first approaches that mapped road signs in 3D proposed to use a particle filter to estimate the 3D position [MKM08]. Using classical feature matching and multi-view estimation, another early approach was able to report an average position error of $0.25\,\mathrm{m}$ [TZV09]. Other approaches that not only estimate the 3D position, but also the size and orientation of traffic signs from multiple views achieved errors in the range of centimeters and single-digit degrees [SPV13]. The author is not aware of any method using only camera images that can estimate the 3D position, and potentially even the size and orientation, of road signs with such precision using a single or very few images – especially at high range.

Extracting poles from stereo camera images using depth edges or detected lines was proposed for the goal of localization [SGR16, BN18], using particle filters to solve the simultaneous localization and mapping (SLAM) problem. Unfortunately, they do not report the accuracy of their methods.

Lidar-based Map Perception

While lidars clearly beat cameras in terms of depth perception, their limited resolution and lack of spectral range/color makes it significantly harder to distinguish the diverse semantic classes of traffic lights and traffic signs. Hence, most pure lidar approaches cover only the detection, but not the exact recognition of such semantically rich map elements.

When lidars are used, two vastly different kinds of sensors can be meant. Mobile mapping systems use one or more single-layer laser scanners with exceptional angular resolution, effective range, accuracy, and total number of returns. Their scans are aggregated using INSs/GNSSs and processed offline, yielding a vast, contiguous point cloud with very high density.

In contrast, automotive lidar sensors typically use multiple layers or similar patterns that cover the vehicle's environment multiple times per second [RCG22].

While being worse in most specification parameters, they enable to perceive the environment multiple times per second, especially many meters ahead of the vehicle. This distinction leads to fundamentally different approaches which cannot always be used on data from the respective other domain. A general survey of lidar-based map perception methods has been published by Gargoum and El-Basyouny [GE17].

For the detection of traffic lights, the author is only aware of two approaches. Both use a DNN trained on a suitably labeled lidar dataset that also contains poles and road signs, either without [PSF+22] or with map prior [PSF+23]. Since the dataset contains equidistantly spaced points from a mobile lidar mapping system, its accuracy does not depend on the detection range. For both approaches Plachetka et al. report errors of single-digit centimeters in position and size. The error of the orientation estimation diverges: $4.4°$ with and $14°$ without map prior. While a suitably trained DNN is used on very dense lidar point clouds, especially the orientation estimation results are still impressive.

Compared to traffic lights, road signs are easier to detect in lidar data. By exploiting their retro-reflective property, several approaches [GRA+11, VYF+13, RDC+16, GES+17, GEM+19] detect them in lidar point clouds using the measured intensity. While the mapping is obsolete using mobile mapping systems, Vu et al. [VYF+13] proposed to use gating in 2D centroid coordinates and orientation for data association in a stationary frame. Evaluating only few signs, they report position errors below or around $10\,\mathrm{cm}$. Using a DNN to detect and localize traffic signs in a single (virtual) point cloud cannot only achieve similar centroid position errors in 3D. It also allows the orientation and size to be estimated with errors in the single digit degrees/centimeters [PSF+22, PSF+23].

The area where multi-layer lidar approaches are predominant is the detection of poles and similar elongated objects. Brenner [Bre09] proposed to extract poles from mobile mapping system point clouds using a multi-layer hollow cylinder model, assuming an accuracy of $10\,\mathrm{cm}$. One of the first works using automotive multi-layer lidars uses jumps in the depth within one scan line to detect poles and match them with an a priori map [SB14]. Other approaches try to find a measure for the “poleness” of points that can then be clustered [TFC+14]. Sefati et al. [SDS+17] transferred the idea by Brenner, *i.e.* stacked cylinders

or circles, to multi-layer lidar sensors. Yet another approach is to detect poles from occupancy maps [SBV+19].

Modern approaches use deep learning to detect poles in real time from range images [DCS+23]. The DNN-based approaches proposed by Plachetka et al. [PSF+22, PSF+23] also detect poles. In their more recent approach, using single, virtual lidar scans, they report errors of $5\,\mathrm{cm}$ for the 3D base point and $2.7\,\mathrm{cm}$ for the diameter. The 3D position error is expected to be larger in real scans from multi-layer lidar sensors, but it is a good baseline for 2D position errors. Note that they omit to estimate the height, which turned out to have a large estimation error using the approach proposed in this thesis.

Camera-Lidar Fusion

Approaches that combine camera and lidar information can benefit from the dense, colorful image information in order to understand the semantics while lidars provide range-independent depth measurements that form the perfect complement. While some approaches use different kinds of map elements from lidar or camera [SDS+17], this thesis focuses on methods that fuse camera and lidar data in order to improve detection over either of the individual modalities.

In an early work, Vu et al. [VYF+13] detect traffic signs candidates in lidar using their retro-reflective property. These candidates are then projected in the camera image and classified using template voting. Other approaches obtain lidar points in a region of interest (ROI) via ego localization. The points are then colorized using camera images and finally classified using a support vector machine (SVM) [ZD14].

Combining the fusion of camera and lidar data with early deep learning methods has first been proposed for moving object detection [SSO06]. With the availability of datasets, DNNs that process combined lidar and camera data are the state of the art.

One of the first works that proposed to use a DNN to detect traffic signs in camera images used the detected bounding boxes to crop regions from an aggregated mobile mapping system point cloud [YWW+19]. Possatti et

al. [PGC+19] accumulate and cluster lidar points whose projections lie inside detection bounding boxes from a DNN across multiple frames in order to obtain sufficiently dense point clouds for 3D position estimation for traffic lights. In the same year, Naujoks et al. [NBW19] combined semantic multi-class 2D detections from camera images with clustered 3D points from lidar.

The author is not aware of previous work, *i.e.* prior to the previous publication [PSS21], that achieved comparable landmark estimates from single frame measurements by combining multi-class semantic detections from a DNN used on camera images with depth perception from lidar data.

3.2.2 Map Element Representations

Next to the detection, another important question is how to represent map elements in a map. A suitable representation needs to balance two goals. On the one hand, it should be compact to be stored or transmitted efficiently. Moreover, fewer parameters are easier to estimate accurately when only limited data, *i.e.* one or few frames, are available. On the other hand, the representation has to be sufficiently rich in terms of semantic classes, but also in terms of geometry. For instance, knowing the dimensions or shape of a traffic sign can significantly help to resolve ambiguities during data association.

While sensor specific representations, *e.g.* image feature descriptors or aggregated point clouds, have been proposed, they are ignored in this work since semantic HD maps are conceived to be sensor agnostic. This category also comprises sensor specific learned object representations [DDS+17, DCD+20, SWD20].

Initial approaches for traffic lights, signs, or poles used only a position in 2D or 3D. Focusing on traffic signs, two early approaches used oriented planes in lidar data [VYF+13], optionally storing the shape [SPV13]. Being semantically agnostic, a widely adapted solution for localization proposed to use edge and planar points from lidar [ZS14]. Kümmerle et al. [KSP+19] demonstrated the use of a similar representation for poles and building facades. Recently, such lines and planes have been described as Grassmannian subspaces, yielding

mathematically sound association metrics [LH22]. Plachetka et al. [PSF+22, PSF+23] showed that the parameters of cylinder and plane representations can be regressed by DNNs.

Another research direction avoids to find specific shapes that represent each semantic category, but instead try to devise generic geometric representations that can approximate multiple object categories. Assuming predefined 3D models, Salas-Moreno et al. [SNS+13] are often credited with proposing object-oriented SLAM. Early approaches combining this idea with DNNs for object detection used centroids [MLP+16] or dense point cloud representations [SPL+17]. More recent approaches model objects of various semantics and shapes using ellipsoids [OLF+19], (dual) quadrics [RCD18, NMS19], superquadrics [TNS+21], or cuboids [YS19] as compact and geometrically sound representations. For an overview, the reader is referred to a recent review with a focus on representations [RDT+21]. The generality of such geometric representations does not require exact knowledge of a semantic class or a specific geometric model for each class. At the same time, in HD maps, the semantics of map elements has wider implications for *e.g.* the driving behavior. Hence, it is deemed acceptable to drop this advantage in favor of better estimates of the elements' positions and shape parameters.

3.2.3 Data Association

Next to object perception and representation, the temporal association of detected elements is an issue to create high-quality HD maps. In a typical SLAM setting, two association parts need to be distinguished [CCC+16]: short-term and long-term data association. The latter is often referred to as "loop closure" and respective techniques are discussed in Section 4.1. For this work, loop closures are neglected and merely local consistency is deemed acceptable. Instead, the focus lies on creating HD maps with human-like precision fully automatically. The integration of the proposed approach as front end in a SLAM framework is considered a solved problem that requires significant engineering work without benefits for the actual contributions.

In a previous publication [PSS21], it was proven that highly accurate maps can be generated given poses and using frame-to-frame associations with naive track management. More sophisticated approaches proposed multiple hypothesis [WE18, HK19] and random finite set methods [MVA+11, DRD15, FGS+17], stochastic processes [NBW19], non-parametric techniques [MLP+16, DFL19] or suitable factor graph formulations [SP13, PP18] to tackle clutter and missing detections.

In contrast to these methods, this work proposes to adapt a graph-based method from object tracking that is exact and optimal [Wan19]. Unlike filtering approaches, it does not assume the Markov property. In contrast to complex factor graph methods, it is known to converge and do so quickly even for conservative assumptions on the death of tracks. For object tracking in images it has furthermore shown state-of-the-art performance. The availability of an open source implementation as well as the possibility to obtain a straightforward parametrization are additional benefits that enabled a fast and certain way to achieve an outlier robust data association. To the best of my knowledge, this work is the first to use graph-based object tracking methods to solve the data association problem for mapping or SLAM.

3.3 Sensors, Synchronization and Calibration

To the best of my knowledge, camera and lidar are the only two widely available sensor modalities with sufficient semantics and resolution in far distance at the moment. The applicability of the approach is demonstrated on two internal datasets recorded with partially different sensor setups.

While this work uses research vehicles and expensive scientific sensors, the part of the setup used for this work is comparable to what has been announced for upcoming series models in terms of both camera resolution [Sha22] and lidar specifications [Ran23], *i.e.* range, field of view and point density. Combined with its modest compute requirements, this makes the proposed approach viable for implementation in both individual customer cars and commercial robotaxis.

3.3.1 2020 “Bertha” Sensor Setup

In 2020, a Flir Blackfly S global shutter color camera was mounted inside the research vehicle behind the windshield. Having $8.9\,\mathrm{Mpx}$ resolution with a roughly $170°$ wide fisheye optics, this results in about $24\,\mathrm{px}/°$ angular resolution which the author considers the relevant number. Cropping the vertically middle part around the horizon leaves $6\,\mathrm{Mpx}$ that are actually processed.

The Velodyne VLS-128 Alpha Prime lidar, mounted on the roof, has about $240\,\mathrm{m}$ effective range and measures around 1810×128 points over $360°\times40°$ when spinning at $10\,\mathrm{Hz}$. This corresponds to a horizontal angular resolution of about $5\,\mathrm{pts}/°$. The vertical angular resolution is variable and highest around the horizon at about $10\,\mathrm{pts}/°$.

The dataset recording with this setup was done by colleagues of the author, Frank Bieder and Haohao Hu.

3.3.2 2023 “Joy” Sensor Setup

In order to close the domain gap between commonly used, publicly available datasets [CBL+20, CLS+19, WQA+21] and the research vehicle used for future research at the MRT, the author designed a new sensor setup consisting of six surround “ring” cameras as well as both a stereo camera and an high dynamic range (HDR) camera to the front, combined with the same Velodyne VLS-128 Alpha Prime lidar sensor used in the 2020 setup.

For this work, only images from the front ring camera, a global shutter Lucid Vision Labs Atlas ATL-071S camera with $7.1\,\mathrm{Mpx}$, were used. Using a lens with $88°$ wide field of view (FoV), it has an angular resolution of about $36\,\mathrm{px}/°$. Both sensor setups are depicted in Figure 3.2.

3.3.3 Synchronization and Calibration

Camera and lidar sensors are synchronized in time using a custom printed circuit board (PCB) which allows locking the lidar's rotational phase such that it passes the center of the image when the camera is triggered.

For intrinsic and extrinsic camera calibration, the frameworks by Strauß et al. [SZB14, Str15] as well as Beck and Stiller [BS18, Bec21] were used. The lidar sensor only required to be calibrated extrinsically, which was done using a spherical target and the method developed by Kümmerle et al. [KKL18, KK19, KK20, Küm20].

3.3.4 Sensor Requirements and Generalization

Perhaps the most important requirement is not inherent in the sensors themselves, but in their synchronization, phase locking, and particularly the extremely accurate calibration as even sub-degree angular errors inhibit sensor fusion already at medium range.

It is supposed that coarsely the same range and angular resolution of both sensors is required to reproduce the results achieved in this work. In contrast,

(a) 2020 sensor setup on the research vehicle "Bertha" with cameras visually enhanced (photo by courtesy of Frank Bieder)

(b) 2023 sensor setup on the research vehicle "Joy" (photo by courtesy of Amadeus Bramspiepe, KIT)

Figure 3.2: Depiction of both sensor setups used to test and evaluate this work.

the horizontal fields of view can vary as long as about 90° to the front of the vehicle are covered by both sensors. For all crucial steps, generalization is preferred over perfect performance and, hence, the author is confident in generalization to different sensor setups.

3.4 Preprocessing

In order to derive semantic parametric detections, one first needs to processes both camera and lidar sensor data. For camera images, a DNN that performs panoptic segmentation [PBC+19], *i.e.* both pixelwise semantic labeling and detection of object instances consisting of boxes and masks, is applied. Lidar point clouds are first used to compute pose using a lidar odometry approach. The poses are then used to compensate the ego vehicle's motion.

3.4.1 Semantic Detections

In this work, the presence of a map element is derived using camera images only. In real systems, a deeply learned sensor fusion approach that directly predicts parametric detections from camera images and lidar points is expected to outperform the proposed approach, but designing such a neural network would go beyond the scope of this thesis, especially due to the lack of a suitable dataset. Instead, this work will show that detections from camera images can be sufficient.

Deep Neural Network and Training Dataset

This work employs Seamseg [PBC+19], a DNN that was readily available and trained on the Mapillary Vistas dataset [NOB+17]. In fact, the author believes that most modern mask-based DNNs would be sufficient for this task as long as masks and not only bounding boxes are predicted. Comparing a network based on YOLO [RF18], *i.e.* using only bounding boxes, for a similar task in a student thesis [Feh21] showed that masks are crucial to capture the relevant lidar points for depth estimation in the later steps.

Figure 3.3: Example detections of the DNN used for extracting instance masks on a typical camera image of the 2020 sensor setup. All masks are combined as clear filter while all other pixels are faded out.

In the author's view, the training dataset is much more relevant than the actual DNN. While Cityscapes [COR+16], Argoverse [CLS+19], NuScenes [CBL+20], and many other popular datasets are captured using a single sensor setup or a fleet of almost identically equipped vehicles, Mapillary Vistas uses a wide range of sensors ranging from smartphones over dashcams to professional cameras. This enables a vastly better generalization to the proprietary sensor setup used at MRT and for this work in particular.

Detections, Bounding Boxes, Masks, and Confidences

For each image $\mathfrak{i} \in \mathfrak{I}$, the output of the DNN is a set of semantic object detections $(\mathfrak{b}, \mathfrak{m}, \mathfrak{c}, \mathfrak{o}) := \mathfrak{d} \in \mathfrak{D}$, often referred to as *instances*, each consisting of a bounding box $\mathfrak{b} \in \mathfrak{B}$, a segmentation mask $\mathfrak{m} \in \mathfrak{M}$, a most probable semantic class $\mathfrak{c} \in \mathfrak{C}$ together with a detection confidence, also referred to as "objectness", $\mathfrak{o} \in [0,1]$.

Ordering and Filtering

However, not all detected instances are equal. Hence, the instances are processed in decreasing order of confidence $\mathfrak{o}$ and discarding less confident detections if their mask overlaps with an already processed instance's mask by more than a class-specific threshold $q^{\mathfrak{c}}_{\text{overlap}}$.

As detections can only be estimated properly when the potential map elements are fully visible, instances that are within a certain distance to the image border are discarded. Additionally, as proposed by a student thesis [Sch20], instances whose mask $\mathfrak{m}$ fills the corresponding bounding box $\mathfrak{b}$ by less than a threshold $q^{\mathfrak{c}}_{\text{fill}}$ are discarded as well. This allows rejecting detections that are very unsuitable for the representation, in particular poles with horizontal or curved arms.

3.4.2 Lidar Odometry

The proposed approach requires 6D poses for two reasons. Firstly, the effect of the ego motion on the continuous rotation of the lidar's sensor head needs to be compensated. Secondly, poses are required to associate the measurements during mapping. Using only the localization and data association method proposed in Chapter 4 for frame-to-frame association would be conceivable in theory. However, being designed for optimality and verifiability, it cannot keep up with the availability of the method proposed in this chapter, *i.e.* using lidar odometry poses with a graph-based association method from object tracking, especially when initial detections are poor or semantic landmarks are sparse.

To compute 6D poses from lidar, this work uses KISS-ICP [VGM+23], a state-of-the-art lidar odometry that performs well on multiple datasets using the same set of parameters. Like for the DNN, again prefer a method that can generalize well to a proprietary sensor setup over highly tuned methods with optimal performance on the respective training set, but vastly inferior generalization ability.

KISS-ICP follows the general ideas of iterative closest point (ICP) methods, but succeeds in implementing them robustly enough to be close to more complex

state-of-the-art lidar odometry methods on a wide range of datasets and sensor setups. It handles motion distorted point clouds by compensating motion internally by extrapolating the motion estimated from the previous frames.

3.4.3 Lidar Motion Compensation

To exploit the strengths of both sensor modalities, lidar point clouds, which are recorded using a continuously rotating and recording lidar sensor, need to be fused with camera images, which are captured using a global shutter camera. This fusion suffers from motion parallax as illustrated in Figure 3.4.

Hence, the motion of the ego vehicle is compensated using the poses computed via a lidar odometry. Since the internal motion compensation of KISS-ICP is worse than a subsequent correction using the all odometry poses, for mapping, the ego motion is interpolated between three poses. As the focus lies on static

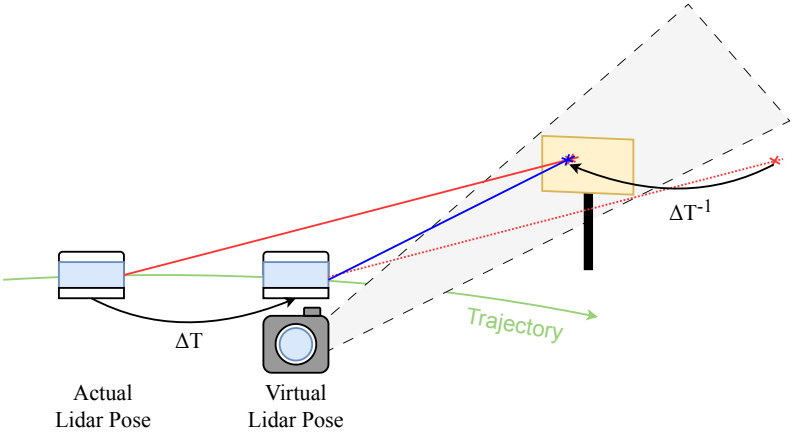


Figure 3.4: Illustration of the motion parallax effect. As the lidar sensor is continuously rotating while recording, a point measured at a vehicle pose with offset ΔT to the pose at which the camera image was captured might falsely seem to lie next to the map element when motion compensation is neglected. This is depicted by the true, solid red ray that is interpreted as dashed red ray. By compensating the relative motion from the actual to the virtual pose at the time the camera image has been taken at, for static objects, the true point can be recovered completely (blue ray).

map elements, one can assume that all relevant lidar measurements are static. Hence, the motion can be compensated by inverting the ego motion relative to the pose at which the camera took the image corresponding to a full lidar scan. So all points are corrected as if they were measured from the same pose as the camera image. To conserve the ordering of the range image, no-returns are shifted accordingly.

3.5 Semantically Tailored Parametric Detections

For each frame of measurement, one can now assume to have object detections $(\mathfrak{b}, \mathfrak{m}, \mathfrak{c}, \mathfrak{o}) = \mathfrak{d} \in \mathfrak{D}$, comprising of bounding box $\mathfrak{b}$ and instance mask $\mathfrak{m}$ in the image, confidence $\mathfrak{o}$, and semantic class $\mathfrak{c}$, as well as a motion compensated lidar point cloud $\mathfrak{L}$. To estimate the parameters of the respective map elements, first, for each instance $\mathfrak{d}$, the relevant lidar points in $\mathfrak{L}$ need to be determined.

3.5.1 Lidar Point Selection and Static Parallax Compensation

As the input data can now be assumed to share a common pose T, one can project all lidar measurements into the image using the camera's intrinsics and an external calibration. Instead of finding the points for each mask individually, it is proposed to project all lidar points into the image at once and cache their pixel coordinates in a spatially sorted order. This allows selecting the relevant points for each detected object in the image using the corresponding mask very efficiently, *i.e.* two orders of magnitude faster than a naive approach.

Still, the projected lidar points have two open issues. First, due to the non-identical optical center, even after motion compensation there will be a significant parallax effect that needs to be compensated. This is illustrated in Figure 3.5 as well as Figure 3.7. Second, not all points have equal likelihood to lie on the true map element that is to be detected.

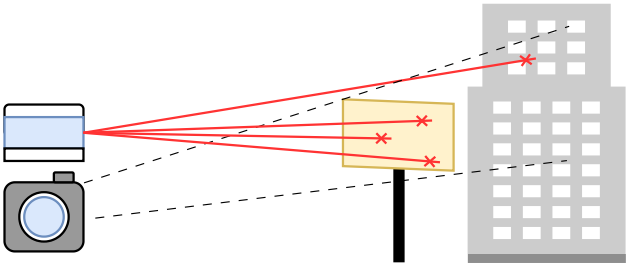


Figure 3.5: Illustration of the static parallax effect. Due to the non-identical optical centers, lidar points can lie behind the map element, but their projection lies within the corresponding detection in image space.

Image Domain Parallax Compensation based on Beam Divergence

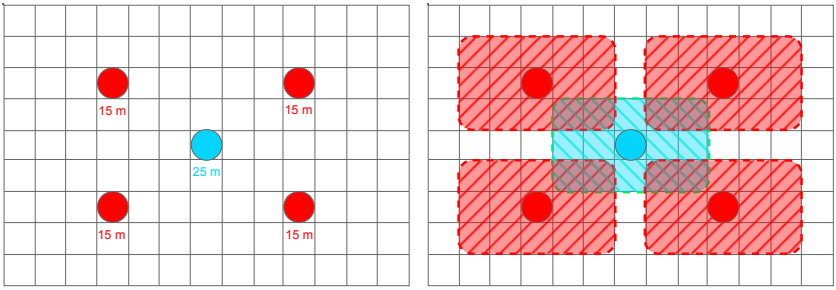
To compensate the parallax effect remaining after compensating the ego motion, for each pixel, the closest lidar point is determined, taking into account the sensor's beam divergence. More specifically, for each beam, using an R-tree [Gut84] the pixels that are within the beam, knowing its divergence specified by the lidar manufacturer, can be tracked efficiently.

If multiple returns hit the same pixel, an overlap due to parallax can be detected and only the closest lidar point is kept. As illustrated in Figure 3.6, this allows mitigating static parallax errors at a much denser level than simple point projections. Its real-world effect is depicted in Figure 3.7.

While no explicit evaluation was performed, during hyperparameter optimization, parallax compensation based on beam divergence was significantly better than tracking only the pixels hit by the beam center.

Retro-Reflective Traffic Signs

For safety reasons, most traffic signs have a so-called retro-reflective surface which reflects most incoming light back into direction of origin. In lidar measurements, this is indicated by an intensity value range larger than the range used for diffuse reflections. Inspired by previous works [VYF+13, RDC+16,



(a) No overlap in the image when considering only ray center pixels (circles).

(b) Overlap detected based on beam divergence (hatched colored areas).

Figure 3.6: Illustration of two strategies to detect parallax errors in the image domain. As each lidar ray center only hits a single pixel, in high resolution images, this rarely leads to multiple hits per pixel, which could be used to reject more distant points (blue). By taking beam divergence into account, most parts of the image are more or less densely covered with lidar hits. This can be exploited to reject most hits with parallax error (blue) as they now overlap with closer hits (red areas). A real-world example is depicted in Figure 3.7.

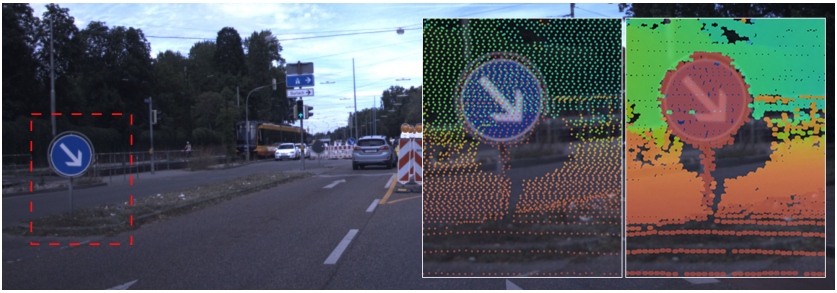


Figure 3.7: A real-world example of parallax error and its compensation. From the perspective of the camera, the traffic sign contains by both (correct) closer lidar points, depicted in red-ish color, and background reflections, depicted in green tones.
A naive reprojection approach would lead to about as many background as foreground points, making distance estimation difficult (left crop). In contrast, if lidar's beam convergence is taken into account, almost all background points can be discarded due to overlap with closer points (right crop).
In the left crop, pixels hit by the ray center are colorized by lidar distance. In the right crop, all pixels hit by the diverged beam are colorized by distance, using the same scale. For both crops, more distant lidar points are rejected during parallax compensation.

GES+17, GEM+19], this allows filtering lidar points on traffic signs by requiring retro-reflective intensity measurements.

3.5.2 Lidar Point Weighting

One now could assume that the remaining lidar points are a good approximation of the distance of the pixel they are projected to and, hence, should contribute to the parameter estimation. However, the probability that this is the case is unlikely to be uniform. Hence, next to uniform weighting, two more sophisticated weighting schemes are proposed. Depending on the semantic class, during hyperparameter optimization, different approaches turned out to be best. All methods are illustrated in Figure 3.8.

The first idea is to weight points relative to their distance to the bounding box center $\mathfrak{p}_c$ with smaller distances inducing higher weight. This works well for detections that are about as wide as high and could trivially be adapted well to pure bounding box detections without a mask. The weight $\mathtt{w}_{\mathfrak{l}_i}$ of a lidar point $\mathfrak{l}_i$ with corresponding pixel $\mathfrak{p}_i$ can be described by

$$\mathtt{w}_{\mathfrak{l}_i} = \left(1 - \frac{\|\mathfrak{p}_i - \mathfrak{p}_c\|_2}{\max_{\mathfrak{p}_j \in \mathtt{m}} \|\mathfrak{p}_j - \mathfrak{p}_c\|_2}\right). \tag{3.1}$$

However, for elongated detections, like poles, this weighting scheme is not well-suited. Instead, the distance to the mask contour can be used as weight with larger distance inducing larger weight. This can be computed efficiently by applying a distance transform (DT) on the mask $\mathtt{m}$.

$$\mathtt{w}_{\mathfrak{l}_i} = \mathtt{DT}_{\mathtt{m}}(\mathfrak{p}_i) \tag{3.2}$$

At the image borders, the masks are zero padded for DT correctness. Optionally, the weight is saturated using a configurable hyperparameter. An adaption to bounding boxes is conceivable using the bounding box as contour.

To filter out detections with no or insufficient depth information, semantic instances are required to have a minimum weight sum over all lidar point weights. Only instances with sufficient point weight are processed further.

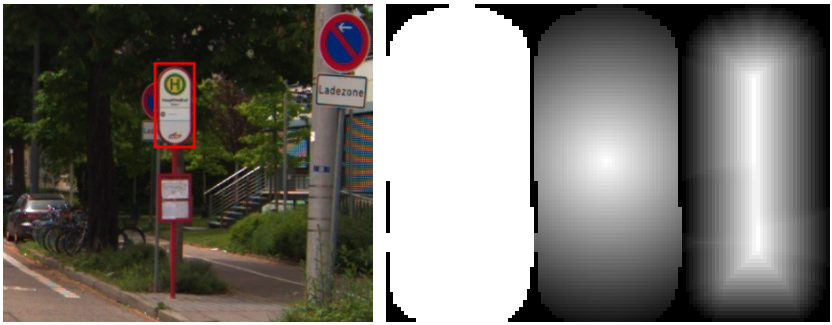


Figure 3.8: Depiction of the proposed schemes to weight lidar points. On the left, a crop of the original image with the detected sign is shown. On the right, the instance mask $\mathfrak{m}$ with uniform, distance to center point and distance to contour weighting is depicted (from left to right). Brighter color indicates higher weight.

3.5.3 Initial Distance Estimation and Inlier Selection

The last step of the camera lidar fusion chain is to provide a good initial estimate of the map element's distance. This initial distance helps to select suitable inlier lidar points and will be crucial to correctly estimate the actual element parameters in the next section.

As visualized in Figure 3.9, despite the previous parallax compensation and point selection, still not all remaining lidar points lie on the true map element. Hence, two methods are proposed to compute an initial distance estimate, density-based spacial clustering of applications with noise (DBSCAN) and a weighted median.

DBSCAN

The method that was already explored in a student thesis [Sch20] and previously proposed in [PSS21] is to perform a density-based spacial clustering of applications with noise (DBSCAN) in the distance domain.

The main idea is to find clusters that have a certain density, defined by a minimal number of $n_{\min}$ neighbors within a radius ϵ. With only two hyperparameters DBSCAN is one of the most popular clustering methods. For a comprehensive

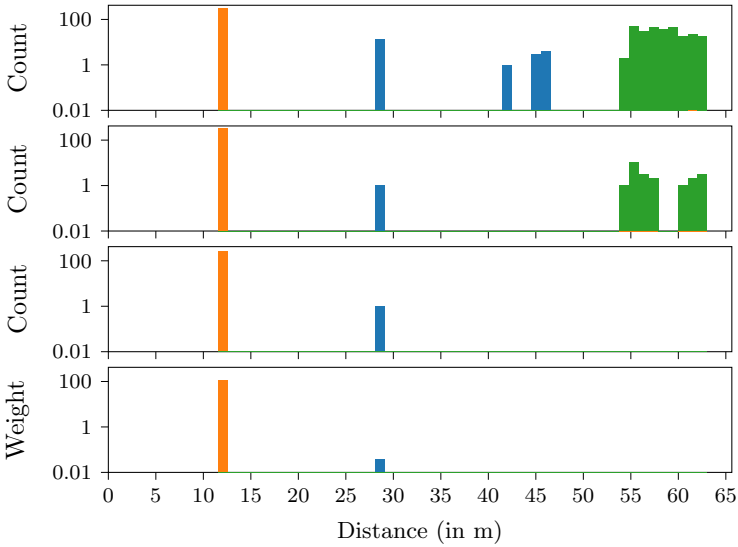


Figure 3.9: Colored histograms of lidar points over distance for the traffic sign depicted in Figure 3.7. Colors indicate hits on the traffic sign of interest (orange), returns on the background (green) and points in between (blue).
The first row shows the histogram of all lidar points within the mask, irrespective of intensity, using ray center parallax compensation. In the second row, beam convergence has been used for parallax compensation. The third row shows the effect of additionally filtering by intensity. Finally, the last row shows results after distance to contour weighting. Note that the ordinate is scaled logarithmically.

explanation of the algorithm and its development, the reader is referred to the original publication [EKS+96] and a modern review [SSE+17].

For this work, a custom C++ implementation is used. It supports weights by modifying the density definition to a minimal weight sum $\mathtt{w}^{\mathtt{c}}_{\min}$ taken over all neighbors within radius ϵ to form a valid cluster. It precomputes neighborhoods, but does not use spatial data structures that do not pay off for the relatively small amount of points. The typical runtime around $1\,\mu\mathrm{s}$ for a single point, growing asymptotically with $\mathcal{O}(n \log n)$ after the potentially parallelizable neighborhood computation.

The initial distance is then derived as the weighted mean over all points in the cluster $\mathcal{C}_{\max}$ with largest weight:

$$d_{\text{init}} = \frac{1}{\sum_{\mathfrak{l}_i \in \mathcal{C}_{\max}} \mathtt{w}_{\mathfrak{l}_i}} \sum_{\mathfrak{l}_i \in \mathcal{C}_{\max}} \mathtt{w}_{\mathfrak{l}_i} \|\mathfrak{l}_i\|. \tag{3.3}$$

Weighted Median

An even faster, but possibly inferior method is to take the weighted median of the lidar points' distances. It can make use of the previously introduced weights $\mathtt{w}_{\mathfrak{l}_i}$ and be computed using a fast state-of-the-art method [RA12].

For the mean weight $\mathtt{w}_0 = \frac{1}{2}\sum_{\mathfrak{l}_i} \mathtt{w}_{\mathfrak{l}_i}$, it finds the weighted median lidar point with index $k = \arg\min_k \sum_{i=0}^{k} \mathtt{w}_{\mathfrak{l}_{N-i}} \geq \mathtt{w}_0$. Hence, the initial distance is defined as the distance of the weighted median point $d_{\text{init}} = \|\mathfrak{l}_k\|$.

Its typical runtime is below $1\,\mu\text{s}$ even for hundreds of points scaling linearly on average [RA12]. While faster methods might be available for larger amounts of data by exploiting parallelization, for the comparatively small data size that usually fits in the CPU's L1 cache no speedup is expected.

Inlier Selection

The estimated initial distance allows "inlier" lidar points to be selected from those reprojected into the mask. Only those points are then used to actually estimate the parameters of the parametric detections. Inlier points are selected in the distance domain, *i.e.* relative to the initial distance, depending on the inlier distance threshold $\tau^{\mathfrak{c}}_{\text{distance}}$:

$$\mathfrak{L}^{\mathfrak{d}_j}_{\text{inl}} = \{\mathfrak{l}_i \in \mathfrak{L} \mid |\|\mathfrak{l}_i\| - d_{\text{init}}| < \tau^{\mathfrak{c}}_{\text{distance}}\}. \tag{3.4}$$

If no cluster of sufficient density, *i.e.* total weight, is found using DBSCAN or the inlier set is smaller than a threshold number $n^{\text{inl},\mathfrak{c}}_{\min}$, the potential detection is discarded.

3.5.4 Semantically Tailored Parametric Models

One key innovation in this chapter is to select a suitable parametric representation based on the semantic class of the detected object. In contrast to many previous works, this work does not attempt to find one general representation that can approximate objects of any semantic class, but is limited in representation accuracy. Instead, the representation is *tailored* to the semantic class by choosing the minimal amount of parameters that is necessary to obtain a representation that is suitable for both the data association in Chapter 4 and the visibility analysis in Chapter 5.

For this work, with traffic signs, traffic lights, and poles, three prototypical semantic classes have been chosen that depict the majority of non-road objects in urban HD maps. Hence, their omnipresence is sufficient to prove the validity of the concepts which will be proposed in the following chapters and build upon the map elements introduced now. The core idea is transferable to other semantic classes using either the same or conceptionally similar representations that are adapted to the concrete requirements of the respective class.

All classes are represented by relatively simple geometric bodies: Traffic signs are simplified as rectangles that are oriented around the up axis. Traffic lights and poles are modeled by cylinders. While traffic lights are assumed to be upright, poles are modeled with an orientation.

This relatively simple modeling choice already allows estimating the parameters with very high accuracy, but also obtain models that have enough fidelity to later be used to verify that each map element is visible. Further modifications, such as the shape of traffic signs, are conceivable and would certainly improve the method. However, when using the same parametric measurements during localization and verification, it is assumed to have only a single frame of measurements and, given this limitation, such additional parameters are extremely hard to estimate accurately. Hence, this is left as an open issue.

3.5.5 Parameter Estimation

As all three semantic classes follow different parametric models, the estimation of parameters is different as well. Regardless of the semantic class, each detection $\boldsymbol{d}$ consists of a 6D pose $(\boldsymbol{c}, \boldsymbol{o}) \in \mathrm{SE}(3)$, two extent parameters $w,h \in \mathbb{R}$, and a semantic class $\mathfrak{c} \in \mathfrak{C}$. The pose is represented by a center point $\boldsymbol{c} = (x_c,y_c,z_c)$ and an orientation $\boldsymbol{o}$, which is stored as quaternion, but compared using the object's major axis. One can imagine the pose as transformation between object and world coordinate system. For the sake of a clearer notation, the semantic class $\mathfrak{c}$ is neglected for mathematical operations on $\boldsymbol{d}$.

Poles

As approaches to estimate the orientation of a pole in the image domain turned out to be unreliable, a pole's orientation is estimated using an (unweighted) principal component analysis (PCA) on the inlier lidar points $\mathfrak{L}_{\mathrm{inl}}^{\mathfrak{d}_j}$ of the corresponding detection $\mathfrak{d}_j$. The pole axis or orientation vector $\boldsymbol{o}$ is then determined by the first principal component, *i.e.* the largest eigenvector of the covariance matrix, normalized to unit length. More robust, but slower PCA approaches, *e.g.* using Grassmann averages [HFE+16], were tried, but the initial point filtering in distance domain renders them redundant. Extremely skewed poles are rejected by a threshold on the angular magnitude of $\boldsymbol{o}$.

In contrast to the orientation, due to vertically limited lidar coverage, the center point $\boldsymbol{c}$ is extracted using the corresponding instance mask $\mathfrak{m}$. First, viewing rays to the middle pixels $\mathfrak{p}_{\mathrm{top}},\mathfrak{p}_{\mathrm{bottom}}$ of the uppermost and lowermost row are determined. Next, the points $\boldsymbol{x}_{\mathrm{top}},\boldsymbol{x}_{\mathrm{bottom}}$ on the axis going through the lidar points' centroid in the direction of the pole's orientation $\boldsymbol{o}$ that are closest to each of the respective viewing rays are calculated. The center point is then the middle of both points:

$$\boldsymbol{c} = {}^1\!/_2(\boldsymbol{x}_{\mathrm{top}} + \boldsymbol{x}_{\mathrm{bottom}}). \tag{3.5}$$

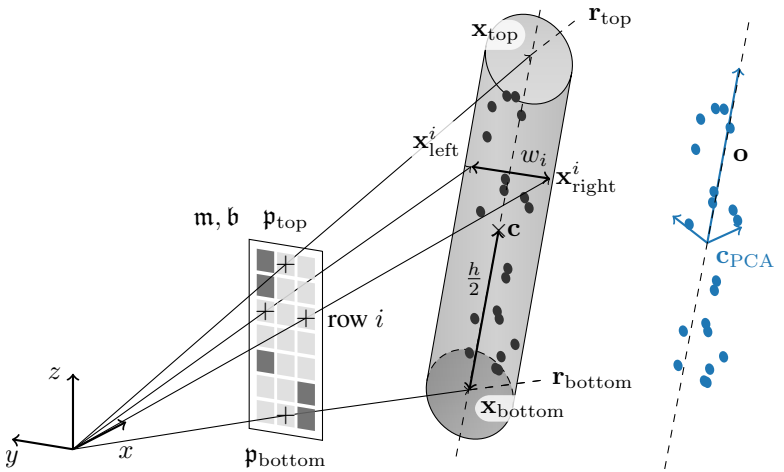


Figure 3.10: Illustration of the estimation of the parameters that represent a pole as a cylinder with center $\boldsymbol{c}$, width w, height h and an orientation vector $\boldsymbol{o}$.
While the real instance mask $\mathfrak{m}$, lidar points, and the model are depicted in black and gray, the abstract PCA coordinate frame is depicted in blue.

The height h of a pole can then be determined by their distance

$$h = \|\boldsymbol{x}_{\text{top}} - \boldsymbol{x}_{\text{bottom}}\|. \tag{3.6}$$

Determining the width of a pole in the image is not trivial as it might change significantly depending on the visibility of the (whole) pole. The fact that the instance masks often include the pole's arm(s) only make matters worse. At the same time, at large distances, the number and density of lidar points is not sufficient to properly determine the pole width or radius.

As robust solution, it is proposed to iterate over each non-zero row of the instance mask $\mathfrak{m}$ and measure the 2D BEV distance of the viewing rays to the leftmost and rightmost mask pixels at the initially estimated distance d_{init}. The pole's width w is then determined as the median width over all rows. All steps of the parameter estimation are depicted in Figure 3.10.

Traffic Lights

Neither the DNN nor the resolution of the lidar sensor allow to resolve the orientation of the traffic light. Instead, it is assumed that the major axis of traffic lights is always oriented upwards in the sensor frame. Admittedly, this assumption is valid neither globally nor for very dynamic driving scenarios, but it is deemed acceptable for this work.

Due to the often rectangular shape which is matched well by the bounding box, the center $\boldsymbol{c}$ is determined from the viewing rays to the central top and bottom pixels of the bounding box $\mathfrak{b}$. Sampling them at the initially estimated distance, d_{init}, this yields the points $\boldsymbol{x}_{\text{top}}$ and $\boldsymbol{x}_{\text{bottom}}$ which can again be averaged:

$$\boldsymbol{c} = {}^1\!/_2(\boldsymbol{x}_{\text{top}} + \boldsymbol{x}_{\text{bottom}}). \tag{3.7}$$

The height h of a traffic light can be determined from

$$\boldsymbol{x}_{\text{top}} = \begin{pmatrix} x_{\text{top}} \\ y_{\text{top}} \\ z_{\text{top}} \end{pmatrix}, \quad \boldsymbol{x}_{\text{bottom}} = \begin{pmatrix} x_{\text{bottom}} \\ y_{\text{bottom}} \\ z_{\text{bottom}} \end{pmatrix} \tag{3.8}$$

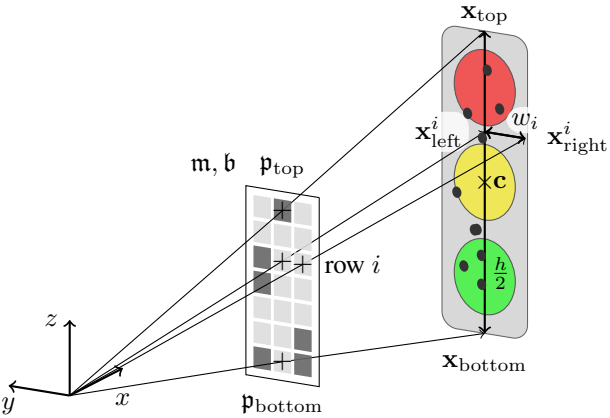


Figure 3.11: Estimation of a traffic light's parameters. Traffic lights are represented as upright cylinders, depicted by its cross section, with center point $\boldsymbol{c}$, width w and height h.

using the upright orientation assumption:

$$h = {}^1\!/_2(z_{\text{top}} - z_{\text{bottom}}). \tag{3.9}$$

Finally, the width w of a traffic light is determined analogously as for poles. This robust method helps particularly when viewing a traffic light from the side which induces significant variation in the observed width.

Traffic Signs

As for poles, for traffic signs first the orientation is estimated, assuming only a rotation around the up axis. Similar as for traffic lights, this limitation is admitted but deemed acceptable for this work. A PCA in the xy domain of the inlier lidar points $\mathfrak{L}_{\text{inl}}^{\mathfrak{d}_j}$ determines the normal vector $\boldsymbol{n}$ as object's major axis, which is chosen to be oriented towards the sensor origin. This allows the orientation to be modeled properly since the DNN only detects the front side of traffic signs while the back sides have a separate semantic class. For deriving an HD map, the orientation could also be used to attribute the induced traffic rules to the relevant lanes.

The center $\boldsymbol{c}$ is determined by intersecting the viewing ray through the center pixel $\mathfrak{p}_{\text{center}}$ of the bounding box $\mathfrak{b}$ with the sign plane. Similarly, width w and height h can be determined by casting viewing rays through the left/right and top/bottom pixels, intersecting them with the sign plane and measuring the respective distance of the intersection points $\boldsymbol{x}_{\text{left}}$,$\boldsymbol{x}_{\text{right}}$ and $\boldsymbol{x}_{\text{top}}$,$\boldsymbol{x}_{\text{bottom}}$.

Like for poles, a hyperparameter is used to filter out signs are observed from a really flat angle as already small deviations in the size of $\mathfrak{b}$ can introduce significant errors in the estimated width and height. But, in contrast to poles, for signs this was rejected during hyperparameter optimization seemingly being inferior to no filtering.

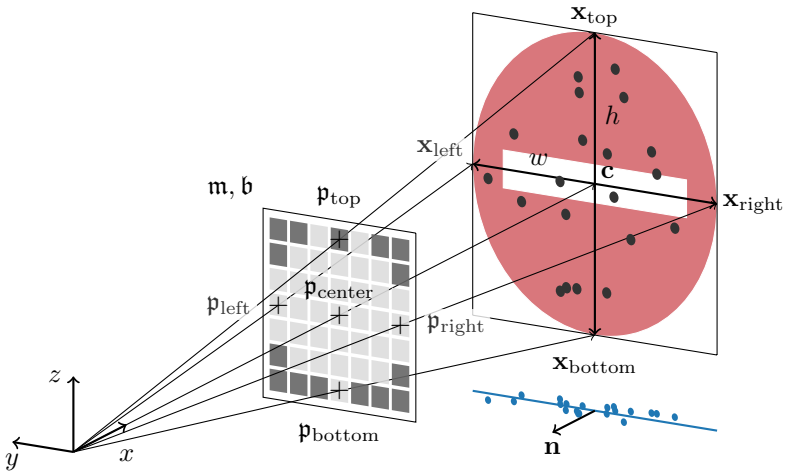


Figure 3.12: Estimation of the parameters representing a traffic sign. Traffic signs are represented using a rectangle which is upright, *i.e.* only rotated around the up axis, and hence can be described by a center point $\boldsymbol{c}$, width w, height h and a normal vector $\boldsymbol{n}$.
The instance mask $\mathfrak{m}$, lidar points, and the model are depicted in black, gray and red. The BEV projected points and the corresponding first axis of the abstract PCA are colored blue.

3.5.6 Measurement Deduplication

As depicted in Figure 3.13, the single measurement frames may contain duplicate instance detections from the DNN. Hence, next to filtering based on mask overlap, as explained in Section 3.4.1, this section examines the possibility to remove duplicates in the estimated parameter space. Duplicates can be detected based on either a heuristic or a probabilistic space using DBSCAN clustering. For simplicity, semantic confusion is ignored and only measurements within each semantic class are deduplicated.

Heuristic Duplicate Detection

A simple heuristic for duplicate detection is to use the parameter subspace that has proven to be stable enough even across spurious detections. For

poles, this is the 2D center in BEV, (x_c,y_c). When detecting traffic lights, the height is consistent as well, allowing to use the 3D center $\boldsymbol{c} = (x_c,y_c,z_c)$ for clustering. Finally, traffic signs exhibit a rather unique and consistently measurable orientation, allowing to use the combination $(\boldsymbol{c}, \boldsymbol{o})$ for clustering. While the Euclidean distance is an obvious choice for pole and traffic light center points, for the combined spatial-angular clustering space, the pseudo-metric

$$\|\boldsymbol{c}_i - \boldsymbol{c}_j\| + \measuredangle(\boldsymbol{o}_i, \boldsymbol{o}_j) \tag{3.10}$$

is used to determine the distance between two measurements $\boldsymbol{d}_i, \boldsymbol{d}_j$. Here, $\measuredangle(\cdot,\cdot)$ denotes the minimal rotation difference.

Probabilistic Duplicate Detection

A fully probabilistic approach involves more parameters, but less arguable hand-crafting. It uses the Mahalanobis distance

$$d_{\text{Mahal}}(\boldsymbol{d}_i,\boldsymbol{d}_j) = \sqrt{(\boldsymbol{d}_i - \boldsymbol{d}_j)^T \boldsymbol{\Sigma}_{\mathfrak{c}_i}^{-1} (\boldsymbol{d}_i - \boldsymbol{d}_j)} \tag{3.11}$$

which is based on the class dependent covariance matrix $\boldsymbol{\Sigma}_{\mathfrak{c}_i}$ whose entries are seen as hyperparameters. To limit the number of hyperparameters, it is assumed that the parameters are uncorrelated, *i.e.* $\boldsymbol{\Sigma}_{\mathfrak{c}_i}$ is a diagonal matrix. Again, semantic confusion is ignored, hence, $\mathfrak{c}_i = \mathfrak{c}_j$.

As image resolution and lidar point density both decrease quadratically over distance, one could expect less certain measurements at larger range. In order to compensate any such potential distance dependency of the covariance, a sigmoidally inspired scaling term $\zeta_{\mathfrak{c}_i}$ is introduced. It depends on the Euclidean distance $\delta_i = \|\boldsymbol{c}_i\|$ between measurement and sensor origin:

$$\zeta_{\mathfrak{c}_i}(\delta_i) = \frac{1 + \exp\big(a_{\mathfrak{c}_j} b_{\mathfrak{c}_j}\big)}{1 + \exp\big(a_{\mathfrak{c}_j}(\delta_i + b_{\mathfrak{c}_j})\big)}. \tag{3.12}$$

(a) First detection mask **(b)** Second detection mask **(c)** Overlay of both masks

Figure 3.13: Example of duplicate DNN detections. Especially poles with signs mounted on them are detected multiple times, often with strongly varying masks. Instead of finding duplicates only in the image domain, they are also detected and merged in the parameter space.

This term can then be used to scale the inverse covariance matrix

$$\boldsymbol{\Sigma}^{-1}_{\mathfrak{c}_i\text{scaled}} = \zeta_{\mathfrak{c}_i}(\delta_i)\boldsymbol{\Sigma}^{-1}_{\mathfrak{c}_i}. \tag{3.13}$$

Examples of the scaling function $\zeta_{\mathfrak{c}_i}(\delta_i)$ for various values of $a_{\mathfrak{c}_i}$ and $b_{\mathfrak{c}_i}$ are depicted in Figure 3.14.

While this scaling makes the Mahalanobis distance asymmetric in the two measurements $\boldsymbol{d}_i, \boldsymbol{d}_j$, this can be neglected as any relevant, *i.e.* sufficiently small, distance compares measurements of approximately similar Euclidean distance to the sensor origin $\delta_i \approx \delta_j$. It is assumed that this renders the Mahalanobis distance approximately symmetric for the relevant range of values.

Indirectly, hyperparameter optimization allows observing any possible distance dependency of the covariance scale. In Figure 3.14, one can see that especially with the 2023 sensor setup there is no or only a negligible distance dependency.

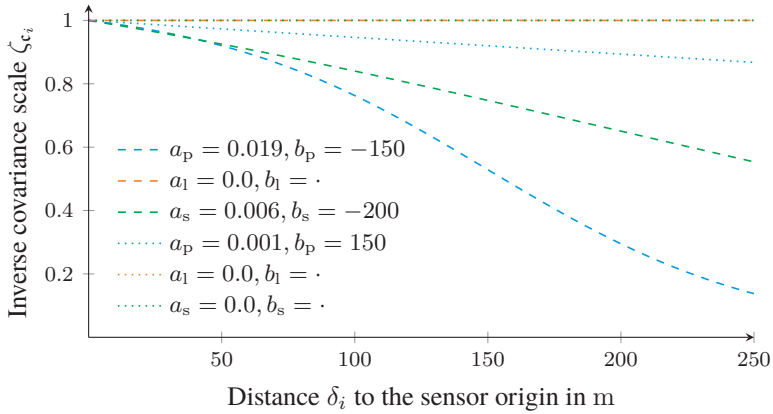


Figure 3.14: Exemplary parametrizations of the scaling function ζ plotted over detection range. A value of $\zeta = 1$ implies not scaling, *i.e.* a covariance that is independent of the detection's distance. The dashed line depicts the automatically optimized values for the 2020 sensor setup for **p**oles, traffic **l**ights, and traffic **s**igns. Dotted are the respective values for the 2023 sensor setup. For $a_{\mathfrak{c}_i} = 0$, the value of $b_{\mathfrak{c}_i}$ does not matter, and all curves with this value are overlapping in the plot.

Duplicate Aggregation

Cohesive duplicate measurements in one time frame are aggregated into a single measurement using the same robust averaging technique that will be introduced in Section 3.6.3 for aggregating measurements across multiple time frames during mapping.

3.6 Highly Accurate HD Mapping

The parametric measurements described so far enable localization and data association for map verification. However, this section shows that they are also an excellent foundation to build a highly accurate HD map in two simple steps: data association over time and optimization of the resulting map elements. The mapping is illustrated in Figure 3.15.

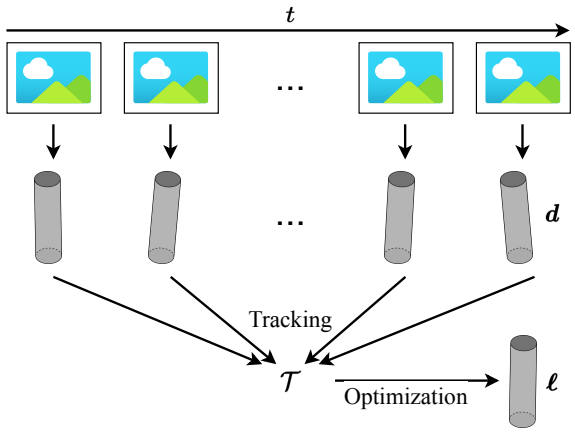


Figure 3.15: Illustration of the mapping procedure. Parametric detections $\boldsymbol{d}$ are extracted individually and tracked over time in parameter space. Map elements $\boldsymbol{\ell}$ can then be estimated for each track $\mathcal{T}$ via robust non-linear averaging.

3.6.1 Acausal Processing

Before discussing these major points, we need to address a seemingly minor detail that was discovered during a student thesis supervised by the author [Sch20] and actually has great impact on mapping performance: the acausal processing of parametric measurements as visualized in Figure 3.16. When associating and aggregating detections reversed in time, *i.e.* as if driving backwards, map elements are first seen when they are largest in the camera image and have the best lidar coverage. This allows initializing them accurately, hence, significantly improving data association via linear assignment, landmark optimization, and localization in the previously created map.

3.6.2 Data Association

The goal of the data association is to determine tracks $\mathcal{T}_k \ni \{\ldots,\boldsymbol{d}_i,\ldots\} \subset \mathcal{D}$ that consolidate all parametric measurements $\boldsymbol{d}_i$ belonging to one putative map element. Using poses from odometry, the detections are transformed into a map coordinate system to facilitate data association. This can also be done using

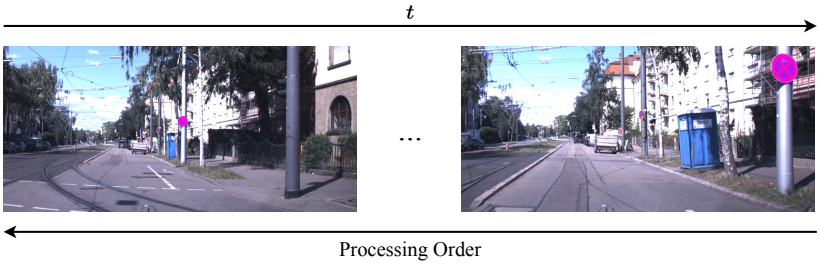


Figure 3.16: Illustration of the acausal processing idea. Typically, map elements are closer and, hence, better to estimate later in time. Thus, processing data backwards in time yields more stable initial estimates and may improve localization in the already estimated map.

the transformation estimated by the novel localization approach proposed in Chapter 4, utilizing the already estimated map in a SLAM fashion[1].

To solve the association problem, in this work, two methods are proposed that both exploit the parameter space of the measurements.

Linear Assignment

In a student thesis [Sch20] and previous publication [PSS21], the Hungarian algorithm [Kuh55, Mun57, JV87] was proposed to solve a linear assignment problem in parameter space. Its input is a cost matrix that is based on the assumption that measurements are distributed normally around the true map element, *i.e.*

$$\boldsymbol{d}_i \sim \mathcal{N}(\boldsymbol{\ell}_k, \boldsymbol{\Sigma}_\mathbf{c}), \tag{3.14}$$

yielding the comparison distribution

$$\boldsymbol{d}_i - \boldsymbol{d}_j \sim \mathcal{N}(\mathbf{0}, \sqrt{2}\boldsymbol{\Sigma}_\mathbf{c}). \tag{3.15}$$

[1] In fact, hyperparameter optimization showed that it is optimal to use the localization approach with very optimistic parametrization, but bounded by the lidar odometry. *I.e.* the localization result is used as long as it is close enough to the odometry pose, otherwise the odometry is used.

Note that $\boldsymbol{\Sigma}_{\mathfrak{c}}$ is the covariance *in the full parameter space* of the semantic class $\mathfrak{c} \in \mathfrak{C}$. For simplicity, semantic confusion is neglected and association is only allowed within the same semantic class. In the experiments, $\boldsymbol{\Sigma}_{\mathfrak{c}}$ is assumed to be diagonal. The variances are estimated empirically.

The solution of the linear assignment problem are associations between measurements and map elements as well as matches of both with a dummy entry used for gating. New tracks are formed from measurements that are not associated to any map element, *i.e*. the dummy entry. Parametric detections that are associated to a map element extend its track.

Despite its simplicity, this already yielded astonishing results. In particular, in contrast to visual tracking in the image space, it could robustly track traffic lights that changed their state.

Minimal-update Successive Shortest Path (muSSP)

When scaling the approach, the combination of false positive and false negative detections either requires manual track handling or leads to duplicate/erroneous map elements. Hence, this thesis proposes an improvement over the framewise linear assignment by adapting muSSP [Wan19], a state-of-the-art graph-based method from object tracking. Together with deeply learned data association [XZC+19] and multi-hypothesis filtering [RVV+14, GWG+18], graph-based methods [ZLN08, LGU15, Wan19] form the state of the art in object tracking.

The effort of training deep learning approaches and the lack of readily available filtering approaches excluded the first two categories. In contrast, muSSP [Wan19] was available as open source code and can be parameterized empirically based on the work by Zhang et al. [ZLN08].

Concept

The basic idea is, as illustrated in Figure 3.17, to form a graph that connects detections according to putative associations as well as to a common source

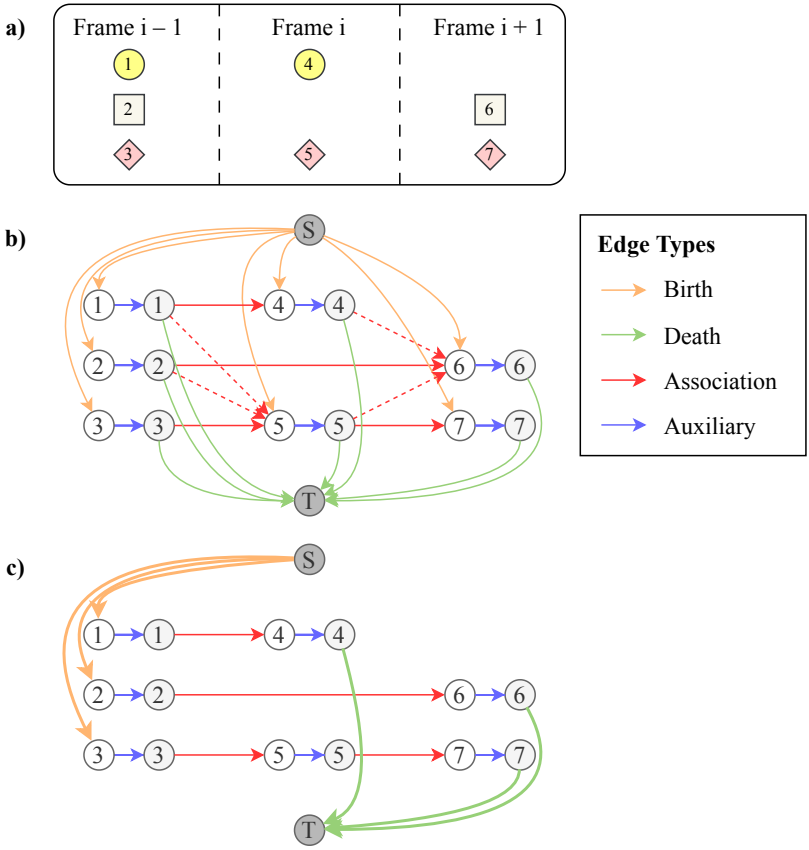


Figure 3.17: Illustration of graph-based data association for mapping across the three frames $i-1$ to $i+1$. In a) three example frames with two or three detections are depicted. In b) these detections are then incorporated in a graph using two vertices each as well as a source vertex S and a sink vertex T. True association edges are solid while false association possibilities are dashed. Tracks are found by computing cost-optimal paths from S to T across birth, detection, association and death edges, representing the respective events for each track. Exemplary optimal paths are depicted in c).

and sink vertex to model birth and death of a respective track. In addition, an auxiliary edge within each edge is required. This results in four kinds of edges of unit capacity that can each be assigned costs c.

The costs are chosen such that birth and death of a track are relatively costly compared to connections between similar detections. However, all of them are positive and only the auxiliary edges have negative costs for including a detection. This cost formulation is probabilistically motivated [ZLN08] and influences the minimal cost paths in two ways. First, it successfully suppresses spurious detections that appear too few times as any too short path has positive costs. In addition, up to a certain point, connecting similar detections in a track is cheaper than creating new tracks.

Formally, edges from source s and to sink t have birth and death cost according to the negative log likelihood of a track appearing or disappearing before/after detection $\boldsymbol{d}_i$, both estimated as class dependent empirical average.

$$c_{si} = -\log(p_{\text{birth}}(\boldsymbol{d}_i)) \equiv -\log(\overline{p_{\text{birth},\mathfrak{c}}}) \tag{3.16}$$

$$c_{it} = -\log(p_{\text{death}}(\boldsymbol{d}_i)) \equiv -\log(\overline{p_{\text{death},\mathfrak{c}}}) \tag{3.17}$$

Detections $\boldsymbol{d}_i$ and $\boldsymbol{d}_j$ are connected within a gating radius in both parameter space and time, *i.e.* frames may be skipped up to a window. Edges connecting $\boldsymbol{d}_i$ and $\boldsymbol{d}_j$ have the negative log association likelihood, *e.g.* modeled by assuming zero-mean Gaussian noise around the true map element (cf. Equations (3.14) and (3.15)), as costs

$$c_{ij} = -\log(p(\boldsymbol{d}_i \mid \boldsymbol{d}_j)) = -\log\Big(f_{\mathcal{N}}\Big(\boldsymbol{d}_i - \boldsymbol{d}_j, \boldsymbol{0}, \sqrt{2} \cdot \boldsymbol{\Sigma}_{\mathfrak{c}}\Big)\Big). \tag{3.18}$$

Finally, detections are modeled by an internal auxiliary edge that models the relative log likelihood of being a valid detection instead of a false alarm. The standard cost term for the auxiliary edge [ZLN08] is

$$c_{ii} = \log\left(\frac{\overline{p_{\text{clutter},\mathfrak{c}}}}{1 - \overline{p_{\text{clutter},\mathfrak{c}}}}\right). \tag{3.19}$$

It can be combined with the detection confidence ("objectness") $\mathfrak{o}_i$ output by the DNN, yielding the cost term

$$c_{ii} = -\log\left(\frac{1-(1-\overline{p_{\text{clutter},\mathfrak{c}}})\mathfrak{o}_i}{(1-\overline{p_{\text{clutter},\mathfrak{c}}})\mathfrak{o}_i}\right). \tag{3.20}$$

Note that the confidence values are not calibrated, although this is recommended for stochastic accuracy [GPS+17, PSC+18, KD19, KKS+20, KHK+22, Küp23]. For a formal and very detailed explanation of the cost terms, including the empirical derivation of the involved likelihoods, the reader is referred to [ZLN08].

The solver, muSSP, now determines the successive shortest (most cost-effective) path problem by iteratively solving the according minimal cost flow problem exactly. The resulting cost-optimal paths then directly correspond to the most likely tracks $\mathcal{T}_k$ which each associate measurements to form a future map element $\boldsymbol{\ell}_k$.

Implementation Details

To accelerate the association, all possible associations and association probabilities are pre-computed. The association problem is constrained to a parametric-temporal vicinity via a hybrid gating which filters by a maximum number of skipped frames as well as a minimal association probability.

As for the Hungarian algorithm, semantic confusion is neglected and only the diagonal entries of the covariance matrix $\boldsymbol{\Sigma}_{\mathfrak{c}}$ are estimated empirically. Again, this assumes that measurements are, in parameter space, normally distributed with zero mean around the true map element. To include the possibility of slightly increasing covariance over distance, the same inverse covariance scaling introduced in Equations (3.12) and (3.13) for probabilistic measurement deduplication was used here as well.

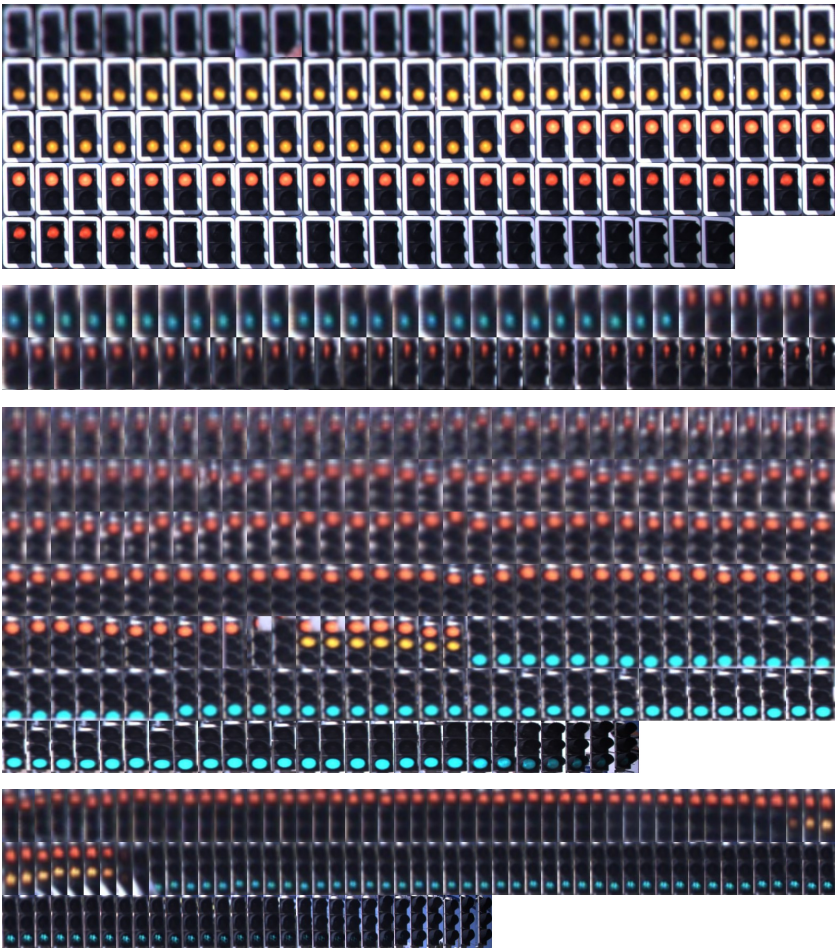


Figure 3.18: Qualitative example results of the data association for traffic lights. Each of the four blocks shows the image crops belonging to the detections of one track, *i.e.* one map element. For this visualization, all crops are scaled to the same size within each track. For many visual tracking techniques, the abrupt change in appearance when the traffic light's state switches would be an issue. In contrast, the proposed tracking in semantic parameter space elegantly circumvents this issue and allows robust tracking despite significant changes in appearance.

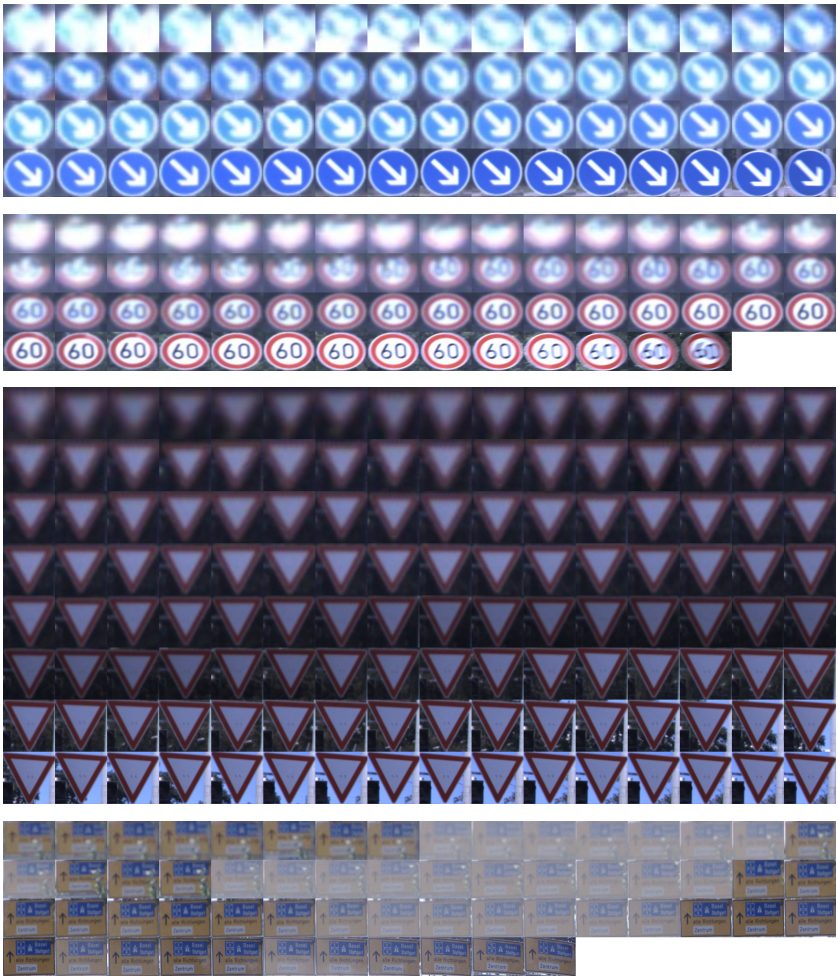


Figure 3.19: Qualitative example results of the data association for traffic signs. Each of the four blocks shows the image crops belonging to the detections of one track, *i.e.* one map element. For this visualization, all crops are scaled to the same size within each track. Tracking is successful over more than a hundred frames and across significant variance in appearance due to lighting, variable exposure, occlusions, changing background, and motion blur. The proposed tracking method is furthermore agnostic of the shape and size, *e.g.* the last sign is several meters high and wide.

Results

Results of the tracking are illustrated in Figures 3.18 and 3.19. Poles can be tracked equally well, but a visualization is omitted as poles are missing visual features that make their tracking graspable for the human eye. The robust tracks enable estimating map elements simply by robust averaging, which will be presented in the next section.

3.6.3 Robust Map Element Estimation

The final remaining question is how to determine the parameters of map element $\boldsymbol{\ell}_k$ from each track $\mathcal{T}_k$. Due to the high accuracy of the parametric measurements $\boldsymbol{d}_i$, like explored in a student thesis [Sch20] and previously published [PSS21], it is proposed to simply perform a robust weighted averaging.

The robust weighted average can be formulated as robustified non-linear optimization problem. More formally, the following problem is solved using Ceres [AMT20] as solver

$$\boldsymbol{\ell}_k = \underset{\boldsymbol{\ell}_k}{\arg\min} \sum_{\boldsymbol{d}_i \in \mathcal{T}_k} \mathtt{w}_i(\delta_i)\, \rho_{\mathtt{c}_i}\left(\|\boldsymbol{\ell}_k - \boldsymbol{d}_i\|_2^2\right). \tag{3.21}$$

Although the individual parameters could be estimated independently, one can use the prior knowledge that *e.g.* a bad depth estimate will impair the estimated width and height. Hence, estimating all parameters jointly weighting each residual with one common robust loss function $\rho_{\mathtt{c}_i}(\cdot)$ neglects outliers w.r.t. to one parameter for the estimation of all parameters.

In Equation (3.21), $\rho_{\mathtt{c}_i}(\cdot)$ is either the Cauchy or Tukey's biweight loss function [BGP19] with their respective scale parameter depending on the semantic class $\mathtt{c}_i$.

For the initialization of $\boldsymbol{\ell}_k$, the hyperparameter optimization can choose between the first seen measurement, the latest measurement or the weighted median of all measurements. While the former two choices are trivial, the weighted geometric median of center points is implemented using a weighted variant of

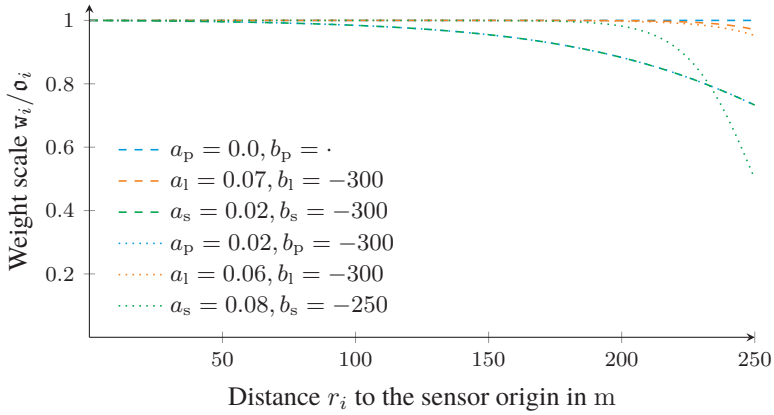


Figure 3.20: Exemplary parametrizations of the weight scaling function plotted over detection range in color. The dashed lines depict the automatically optimized values for the 2020 sensor setup for **p**oles, traffic **l**ights, and traffic **s**igns. Dotted are the respective values for the 2023 sensor setup. For $a_{\mathtt{c}_i} = 0$, the value of $b_{\mathtt{c}_i}$ does not matter.

Weiszfeld's algorithm [Wei37]. The weighted median of orientations can be computed via quaternion representation [MCC+07].

As additional, but qualitatively minor improvement it is proposed to scale weights $\mathtt{w}_i$, again using a sigmoidally inspired weight scaling function with parameters $a_{\mathtt{c}_i}, b_{\mathtt{c}_i}$ that re-weights the detection's confidence $\mathtt{o}_i$:

$$\mathtt{w}_i(\delta_i) = \frac{1 + \exp(a_{\mathtt{c}_i} b_{\mathtt{c}_i})}{1 + \exp(a_{\mathtt{c}_i}(\delta_i + b_{\mathtt{c}_i}))} \mathtt{o}_i. \tag{3.22}$$

Like for the inverse covariance scaling, the input parameter δ_i is the distance of the detection to the sensor origin. During hyperparameter optimization, for different classes different parameter combinations turned out to be optimal. The corresponding examples are illustrated in Figure 3.20.

It is interesting to note that the hyperparameter optimization preferred weighting without any significant distance dependency, at least for all relevant distances. This observation is similar to the rejection of covariance scaling in Figure 3.14. While no direct proof, it allows the assumption that the quality of detections

does not vary significantly over detection range. This simplifies the assumptions of the data association approach presented in Chapter 4.

3.7 Metrics for Weakly / Self-Supervised Hyperparameter Optimization

The lack of suitable datasets and the huge effort to annotate a parametric HD map with all parameters not only denies using standard deep learning approaches for parametric detections. It also makes hyperparameter optimization and evaluation a challenge. As a remedy, a novel metric for parametric detections, resulting HD maps, and localization therein is proposed. It enables both automatic hyperparameter tuning and a quantitative evaluation *without* needing any ground truth. While this option was not explored, one could as well imagine it to be a suitable loss to train a DNN to detect map elements.

As for machine learning in general, there are two ways to evaluate a predicted result when lacking annotated ground truth. First, one can take pseudo labels, *i.e*. noisy and imperfect predictions, that stem from a pretrained DNN such as the ones used in this work to create parametric detections. This follows the machine learning paradigm of weak supervision and exploits that the DNN predictions are independent from the subsequent detection and mapping processes (though not vice versa).

Another way to obtain a supervision signal is to exploit natural principles, such as optical projections and the persistence of static objects in space over time. This is referred to as self-supervised learning.

By combining both ideas, one can obtain a new family of metrics, called *Rendering Instance IoU*. It renders objects that are known to be static into an artificial camera image which can then be compared with the DNN pseudo labels.

Similar ideas have previously been proposed under the terms *analysis-by-synthesis* or *render-and-compare*, *e.g*. to train 3D object detection DNNs. There are various ways to construct suitable losses. When only the shape is known, object coordinates can be used [ZKB+20]. Knowing shape and semantic class,

like proposed in this work, allows rendering a semantic mask that can be compared. Finally, when not only the shape, but also texture can be estimated, a photometric loss can be computed [CLG+19, BKM+20].

3.7.1 Detection Rendering Instance IoU

More formally, for each image $\mathfrak{i}$, given corresponding predicted instance masks $\mathfrak{M}_\mathfrak{i}$ from a DNN such as Seamseg [PBC+19], one can transform parametric detections $\boldsymbol{d} \in \mathcal{D}$ into the sensor coordinate frame, $\boldsymbol{d}' = T^{-1}\boldsymbol{d}$. Assuming known intrinsic camera parameters, they can then be rendered into the image $\mathfrak{i}$ using the rendering process $\mathtt{R}(\boldsymbol{d}', \mathfrak{i})$.

This makes it possible to calculate the Rendering Instance IoU, referred to as RIIoU, of a predicted instance mask $\mathfrak{m}$ and a detection $\boldsymbol{d}'$:

$$\mathrm{RIIoU}\big(\boldsymbol{d}', \mathfrak{i}, \mathfrak{m}\big) = \frac{\big|\mathtt{R}(\boldsymbol{d}', \mathfrak{i}) \cap \mathfrak{m}\big|}{\big|\mathtt{R}(\boldsymbol{d}', \mathfrak{i}) \cup \mathfrak{m}\big|}. \tag{3.23}$$

One can then derive the *mean* Rendering Instance IoU, mRIIoU, over all detections $\mathcal{D}'$ and instance masks $\mathfrak{M}_\mathfrak{i}$ for an image $\mathfrak{i}$:

$$\mathrm{mRIIoU}(\mathcal{D}', \mathfrak{i}, \mathfrak{M}_\mathfrak{i}) = \frac{1}{\Psi} \sum_{\substack{\boldsymbol{d}' \in \mathcal{D}' \\ \mathfrak{m} \in \mathfrak{M}_\mathfrak{i}}} \mathbf{1}_{\boldsymbol{d}'}(\mathfrak{m}) \frac{\big|\mathtt{R}(\boldsymbol{d}', \mathfrak{i}) \cap \mathfrak{m}\big|}{\big|\mathtt{R}(\boldsymbol{d}', \mathfrak{i}) \cup \mathfrak{m}\big|}. \tag{3.24}$$

Its computation involves solving a linear assignment problem [Kuh55] between rendered detections and masks such that the mean RIIoU is maximized. The indicator function is $\mathbf{1}_{\boldsymbol{d}'}(\mathfrak{m}) = 1$ if and only if $\boldsymbol{d}'$ and $\mathfrak{m}$ are matched in the optimal assignment. The normalization term

$$\Psi = |\mathcal{D}'| + |\mathfrak{M}_\mathfrak{i}| - \sum_{\substack{\boldsymbol{d}' \in \mathcal{D}' \\ \mathfrak{m} \in \mathfrak{M}_\mathfrak{i}}} \mathbf{1}_{\boldsymbol{d}'}(\mathfrak{m}) \tag{3.25}$$

ensures that the mRIIoU is normalized to the unit interval: It is zero for no overlap at all and one if every detection and mask is matched pixel-perfectly.

Note that, in contrast to the generic pixel level mean intersection over union (mIoU) used for semantic segmentation, the metric exploits knowledge about individual instances and possibly matching detections. The intersection over union (IoU), operating on a pixel level, was chosen over instance-specific metrics like average precision (AP) and average recall (AR) as to gain better, pixel accurate resolution. However, comparing mRIIoU and analogue AP/AR metrics both formally and empirically remains an open research question.

The mRIIoU can then be interpreted as a noisy approximation of the likelihood of the detections given the image's corresponding instance masks[1]

$$f_{\mathcal{D}|\mathfrak{M}_\mathfrak{i}}(\mathcal{D} \mid \mathfrak{M}_\mathfrak{i}) \propto \mathrm{mRIIoU}(\mathcal{D}', \mathfrak{i}, \mathfrak{M}_\mathfrak{i}). \tag{3.26}$$

A drawback when using detections and instance masks from the same pose is that the depth and, hence, 3D position and size of the parametric detections cannot be observed properly. Thus, it is proposed to use instance masks from the close vicinity of the pose that the detections stem from. To render parametric detections into another image one can use the poses available by lidar odometry or methods proposed in the remainder of this thesis. Transforming between time frames involves two poses, T_i and T_j:

$$\boldsymbol{d}_i^j = T_j^{-1}\boldsymbol{d}_i = T_j^{-1}T_i\boldsymbol{d}_i^i. \tag{3.27}$$

Due to persistence of static objects and the negligible changes in occlusions, this makes the estimated depth, 3D position, and size observable and turns the mRIIoU into a valid metric for evaluating all parameters of the proposed detection method. For detections from time frame i evaluated using image and masks from time frame j this metric is called *Detection Mean Rendering Instance IoU* $\mathrm{mRIIoU}\big(\mathcal{D}_i^j, \mathfrak{i}_j, \mathfrak{M}_{\mathfrak{i}_j}\big)$:

$$f_{\mathcal{D}_i|\mathfrak{M}_{\mathfrak{i}_j},T_i,T_j}\big(\mathcal{D}_i \mid \mathfrak{M}_{\mathfrak{i}_j}, T_i, T_j\big) \propto \mathrm{mRIIoU}\Big(\mathcal{D}_i^j, \mathfrak{i}_j, \mathfrak{M}_{\mathfrak{i}_j}\Big). \tag{3.28}$$

[1] Note that, although the pose T is formally involved here, it cancels out when evaluating the detections $\boldsymbol{d}' \in \mathcal{D}'$ in the sensor frame in which they have been measured.

Empirically, $j = i \pm 3$ was found to be a good trade-off between occlusions and depth/size observability.

An example is illustrated in Figure 3.21.

3.7.2 Map Rendering Instance IoU

Analogously to detections from a single time frame, one can render the optimized map $\mathcal{M}$ into the images $\mathfrak{I}$ recorded within the mapped area. Again, this assumes poses $\boldsymbol{T}$ which can stem from either lidar odometry or the localization method proposed in Chapter 4. To speed up computation times, the rendered map elements are pruned to the possibly visible part given the camera's field of view and a conservatively approximated maximum detection range. Like for detections, for each image $\mathfrak{i} \in \mathfrak{I}$, an optimal linear assignment problem between instance masks and map elements is solved to maximize the mean RIIoU.

This allows us to define the *Map Mean Rendering Instance IoU* metric to approximate the likelihood of the map given the images $\mathfrak{I}$ with corresponding instance masks

$$f_{\mathcal{M}|\mathfrak{M}_\mathfrak{I},\boldsymbol{T}}(\mathcal{M} \mid \mathfrak{M}_\mathfrak{I}, \boldsymbol{T}) \propto \frac{1}{|\mathfrak{I}|} \sum_{\mathfrak{i}_i \in \mathfrak{I}} \mathrm{mRIIoU}\big(T_i^{-1}\mathcal{M}, \mathfrak{i}_i, \mathfrak{M}_{\mathfrak{i}_i}\big). \tag{3.29}$$

The metric has two issues that need to be discussed: occlusions and false positive detections. While the metric neglects occlusions, which can be severe and consistent over multiple frames, one can argue that *any* viable mapping result will suffer from the same occlusions. Regarding false positive detections, due to the nature of the proposed detection and mapping framework, *i.e.* being based on detected instances from a DNN, one may object that false positives which exactly match occluded map elements are so rare that they can be neglected.

It was also tried to use the visibility check that is proposed in Chapter 5 to predict occlusions. However, as already small pose errors severely harm true positive map elements that are falsely predicted occluded, it did more harm than good.

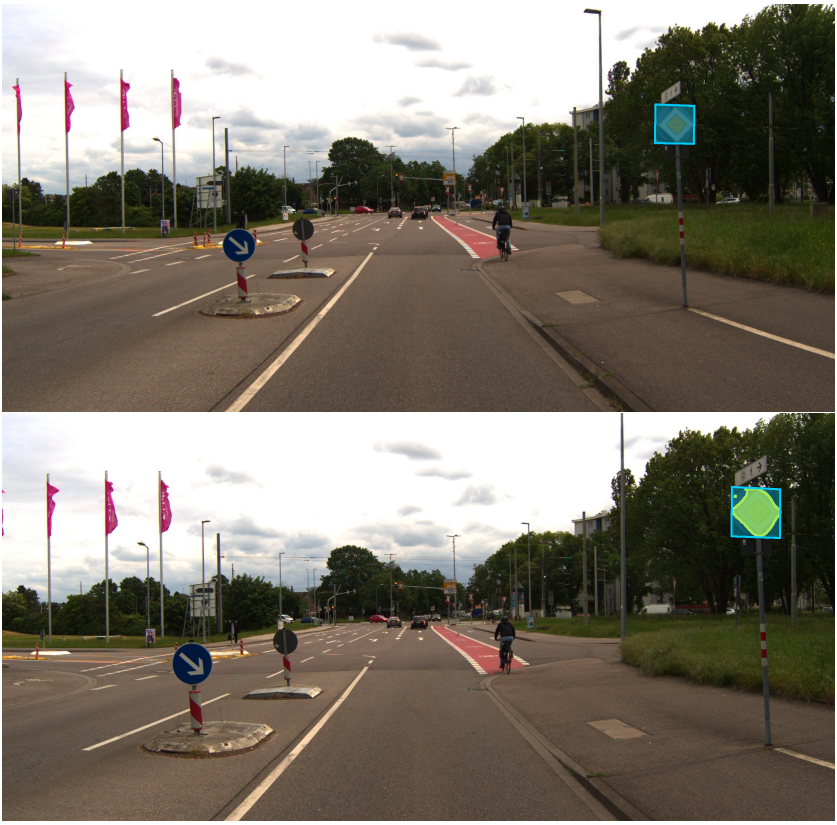


Figure 3.21: Example of the Rendering Instance IoU (RIIoU) metric for parametric detections. In the upper half, we see the parametric detection of a traffic sign rendered as blue quadrilateral $\mathtt{R}\big(\boldsymbol{d}_i^i, \mathtt{i}_i\big)$ depicted over the original image $\mathtt{i}_i$ it was detected in.
In the lower half, *the same* detection is rendered into the evaluation image $\mathtt{i}_{i+3}$ of three frames later, depicted in the background. The green overlap is the instance mask, originally drawn in yellow, yielded from the DNN by processing the evaluation image $\mathtt{i}_{i+3}$, with largest IoU with the rendering $\mathtt{R}\big(\boldsymbol{d}_i^{i+3}, \mathtt{i}_{i+3}\big)$. The IoU between blue rendering and yellow mask is maximal if and only if the center in 3D, the size, and the orientation are estimated correctly.
As the shape of many map elements is challenging to predict from great distance using only a single frame, this work omits estimating them. Hence, the RIIoU is only close to $100\,\%$ for map elements whose shape perfectly fits the representation, *e.g.* rectangular signs or straight poles with constant width. In the example, it is only $58\,\%$ although the detection is close to optimal.

3.7.3 Localization Quality Estimation

By reformulating, one can also use the map mRIIoU to approximate the likelihood of a pose or localization result T_i given a pre-computed map $\mathcal{M}$ and the image $\mathfrak{i}_i$ with corresponding instance masks $\mathfrak{M}_{\mathfrak{i}_i}$ that both correspond to pose T_i:

$$f_{T_i|\mathcal{M},\mathfrak{M}_{\mathfrak{i}_i}}(T_i \mid \mathcal{M}, \mathfrak{M}_{\mathfrak{i}_i}) \propto \mathrm{mRIIoU}\big(T_i^{-1}\mathcal{M}, \mathfrak{i}_i, \mathfrak{M}_{\mathfrak{i}_i}\big). \tag{3.30}$$

The extension to multiple poses $T \in \boldsymbol{T}$ is straightforward.

3.7.4 Inferior Metrics

During the development of this work, two further self-supervised metrics were tried, but both turned out to be inferior to the proposed RIIoU metric. They are discussed in Appendix A.

3.8 Hyperparameter Optimization

Over the time of development of the detection and mapping framework, a number of hyperparameters accumulated. While some parts such as the data association described in Section 3.6.2 could be parameterized empirically, the various options of this work required a general way to optimize hyperparameters anyway. At the same time, the availability of a hyperparameter optimization framework enables the inclusion of small improvements or testing of ideas relatively easily since any additional hyperparameters can simply be found through optimization.

Hence, in order to enable the many small ideas and to minimize the influence of manual tuning, one can employ sequential model-based algorithm configuration (SMAC) [LEF+22], a state-of-the-art hyperparameter optimization framework. Together with the previously proposed metric, it enables an automatically optimized detection and mapping framework. Like additional features, the localization presented in the next chapter can be added into the optimization problem.

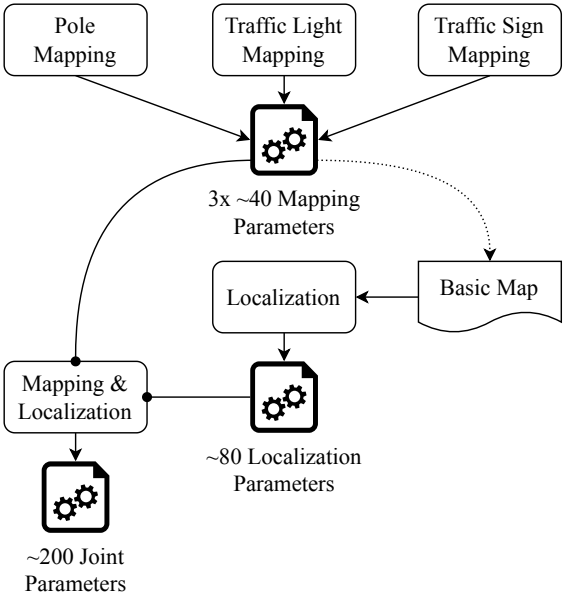


Figure 3.22: Illustration of the distributed hyperparameter optimization scheme. First, the mapping hyperparameters are optimized for each semantic class separately using lidar odometry poses. Next, the localization hyperparameters are optimized using a basic map which has been created using the previously optimized parameters (combined for all classes) and poses from lidar odometry. Finally, all around 200 hyperparameters for joint mapping and localization are optimized using the previously explored hyperparameter distributions as priors (indicated by round arrow ends). The exact numbers of each parameter set vary slightly by task, experiment, and configuration.

The employed hyperparameter optimization method, SMAC (as of version SMAC3 v2.0), not only supports the latest features from automated machine learning, like multiple objectives and multi-fidelity, *i.e.* the ability to quickly condense the best hyperparameters on gradually larger training sets. It also comprises some of the state-of-the-art approaches, including Hyperband [LJD+18] and BOHB [FKH18]. The even more advanced DEHB [AMH21] was tried as well, but discarded due to usability issues and unclear advantages for this use case.

While SMAC makes use of sophisticated surrogate models and acquisition functions, it is still advantageous to simply minimize the parameter space.

Hence, this work proposes a distributed hyperparameter optimization scheme that explores the parameters for mapping the involved semantic classes as well as the localization separately; possibly in parallel for each semantic class.

Using the method of Hvarfner et al. [HSS+22], SMAC can incorporate expert knowledge, empirical parametrizations or the results of previous stages as priors. To do the latter, for each hyperparameter with sufficient, *i.e.* ≥ 3, distinct samples in the best decile of results, one can fit a normal distribution centered at the best parameter value as prior. Categorical hyperparameters can be assigned a weighted discrete prior. The weight is relative in $[0, 1]$ to their performance in the top decile of results, but each option has at least a weight of 0.1.

Without evaluating the speedup enabled through the distributed optimization scheme empirically, it is estimated to be at least $2\times$ even *without* parallelization across classes. An illustration can be found in Figure 3.22.

3.8.1 Dataset

For each of the years 2020 and 2023, this work uses sequences along four routes recorded in and around the city of Karlsruhe, Germany, depicted in Figure 3.23. Comprehensive details, including date, time, weather, and road conditions, are provided in Tables B.1 and B.2.

To exploit the advantages of multi-fidelity, 27 typical, but rather challenging sections were selected from each of the 2020 and 2023 datasets. As typical for machine learning, the scenarios used for optimization were chosen to have no spatial overlap with the sequences used for evaluation, basically having a *training* set for hyperparameter optimization as well as a separate *validation* set. Comprehensive lists of all hyperparameter optimization scenarios with brief descriptions can be found in Tables B.3 and B.4.

3.8.2 Detection Hyperparameters

Evaluating parametric detections is the fastest of all since only a single frame is computed at each time which can be parallelized perfectly not only across

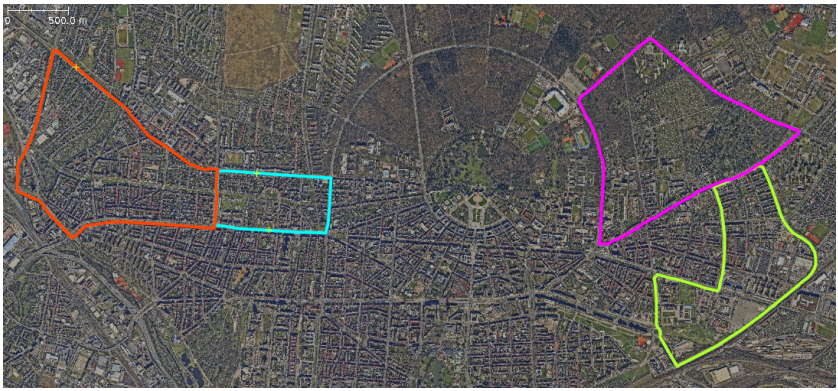


Figure 3.23: Depiction of the sequences used for hyperparameter optimization (green, cyan) as well as evaluation (red, pink). The scenarios used for multi-fidelity hyperparameter optimization are sampled from the non-overlapping parts of the sequences.
Aerial Imagery: © Stadt Karlsruhe | Liegenschaftsamt

semantic classes, but also across time. Hence, this is used to find initial values for the extraction of parametric detections from raw or pre-processed sensor data. As loss, the novel RIIoU metric for parametric detections (*cf*. Equation (3.28)) is used. Due to correlated data, only every fifth frame is computed and evaluated with the instances masks from three frames later.

Detection hyperparameter optimization runs multiple times faster than real time, even after accounting for skipped frames. Since it can be parallelized efficiently, it was found being limited mainly by loading and parsing data.

3.8.3 Map Hyperparameters

To optimize the hyperparameters required for mapping, one needs to compute the detections and map at every time frame in order to find optimal hyperparameter values for mapping at full temporal resolution. However, again due to highly correlated data, the map quality was evaluated only at every fifth frame. To avoid hyperparameter sets that suppress a semantic class entirely, a soft/smooth minimum, *e.g*. mellowmin [AL17], of the RIIoUs of all classes can be used to equalize performance while leaking information about the non-minimal classes

to the hyperparameter optimization framework. In a typical hyperparameter optimization scheme, *i.e.* multiple parallel trials on the same machine and with varying detection density over the instances, $10\,\mathrm{s}$ of data were mapped and evaluated in 10 to $50\,\mathrm{s}$.

3.8.4 Localization Hyperparameters

Assuming for now a parametrizable black box, one can as well find optimal hyperparameters for the localization approach proposed in the next chapter. As they are independent from the hyperparameters used for detections and mapping, one can run an initial optimization to obtain reasonable value ranges and prior distributions.

3.9 Evaluation

For evaluation of all contributions, *i.e.* detections, mapping, and hyperparameter optimization, one can use the proposed Rendering Instance IoU (RIIoU) metrics that has been introduced in Section 3.7.

3.9.1 Metrics

While IoU metrics allow to compare different configurations, it is difficult to infer concrete meaning for applications. Hence, two more metrics are proposed to evaluate the approach.

k-Recall

The first metric is the so-called k-recall of landmarks in a range window $(d^-, d^+]$ that measures from how far away a map element is detected how often. To illustrate this, one can imagine to drive along a sequence with landmarks appearing at the horizon. The goal is to detect all landmarks as soon as they are

visible and/or in lidar range. Ignoring occlusions, the k-recall now measures to which degree this goal has been achieved at which distance.

Specifically, it evaluates the share of landmarks in range $(d^-, d^+]$ that were already associated with at least k measurements. This involves the landmarks in range $(d^-, d^+]$ from the sensor origin $\boldsymbol{t}_i$ at frame i, $\mathcal{M}(d^-, d^+](\boldsymbol{t}_i) = \{\boldsymbol{\ell} \in \mathcal{M} : d^- < \|\boldsymbol{t}_i - \boldsymbol{c}_{\boldsymbol{\ell}}\| \leq d^+\}$, and the set of detections until frame i that were associated with map element $\boldsymbol{\ell}$ during mapping[1], $\mathcal{D}^{\boldsymbol{\ell}}_{\leq i}$. The k-recall can then be defined by

$$\mathrm{Rec}\big(k, \big(d^-, d^+\big], \mathcal{M}\big) = \sum_{i=0}^{N} \frac{1}{N|\mathcal{M}(d^-, d^+](\boldsymbol{t}_i)|} \sum_{\substack{\boldsymbol{\ell} \in \mathcal{M}\left(d^-, d^+\right](\boldsymbol{t}_i): \\ \left|\mathcal{D}^{\boldsymbol{\ell}}_{\leq i}\right| \geq k}} 1. \tag{3.31}$$

Ideally, a k-recall of 1 is achieved as soon as the landmark is visible for k frames. A useful upper bound that corresponds to this visibility for the proposed system is the number of frames in which enough, *i.e.* at least two, lidar points hit the landmark.

Measurement Errors

As second metric, one can use the statistic of the measurement error e_ξ of parameter(s) ξ. It is defined via the detections $\boldsymbol{d} \in \mathcal{D} : \boldsymbol{d} \to \boldsymbol{\ell}$, which are associated to the respective landmarks $\boldsymbol{\ell} \in \mathcal{M}$ during mapping, by

$$e_\xi(\boldsymbol{d}, \boldsymbol{\ell}) := \|\xi_{\boldsymbol{d}} - \xi_{\boldsymbol{\ell}}\|. \tag{3.32}$$

As the landmarks themselves are only estimated, they are only *conventional* true values from a metrological point of view.

In combination with the rendering instance IoU and k-recall metric, measurement errors provide a comprehensive picture of detections and map. At the same time, k-recall and measurement errors allow to evaluate the proposed RIIoU

[1] In contrast to the proposed mapping approach, this employs *causal* processing of measurements.

metric and reveal possible “cheating”, *i.e.* overfitting in the IoU domain during hyperparameter optimization without providing an actual benefit.

3.9.2 Parametric Detections

Regarding parametric detections, the claim that the proposed method achieves human-like precision at high range needs to be substantiated. As parametric ground truth annotations in 3D space are considerably more laborious than *e.g.* annotations in image space, a true ground truth is omitted. Instead, in this work, the parametric detections are evaluated using the HD map, created from the very same parametric detections, as reference.

Since this could become a self-fulfilling prophesy decoupled from reality, in the next section, the map as reference is evaluated quantitatively using the proposed RIIoU metric and qualitatively in 2D images. Additionally, the author performed visual inspection in 3D which showed that the 2D projections are not deceiving. On the contrary, it is difficult to find instances where manual annotation could improve the automatically generated parametric map elements.

Measurement Errors

In Figures 3.24 and 3.25, the measurement errors of the parametric measurements are depicted over the range, *i.e.* the distance from sensor to map element. For the center points the errors in 3D and the ground plane, $e_{\mathbf{c}}^{3D}$ and $e_{\mathbf{c}}^{xy}$, respectively, are depicted. In addition, the errors in width, e_w, height, e_h, and orientation, $e_{\boldsymbol{o}}$ are plotted.

The plot not only shows that the proposed approach can detect map elements with low noise even at large distances. It also underlines that the measurement errors are close to distance invariant. This is an important assumption of the localization and data association approach presented in Chapter 4. The root mean squared errors (RMSEs) corresponding to Figures 3.24 and 3.25 can be found in Tables C.1 and C.2.

k-Recall over Range

However, the measurement errors are only half the truth since there is a trade-off between the measurement errors and the possible k-recall over range: the mapping might ignore unsuitable detections, thus, improving measurement errors at the expense of recall. Hence, in Figure 3.26, the k-recall of the farthest detections of each map element is depicted over detection distance. This serves as lower bound for the detections themselves since the mapping may ignore but cannot create new detections. As upper bound for detections with distance invariant range measurements, one can use the farthest distance at which the map elements were visible in lidar using the visibility test introduced in Chapter 5.

Figure 3.26 shows that with the 2023 sensor setup at least $20\,\%$ of the traffic lights and signs, which are important to be verified ahead, can be detected as far as $100\,\mathrm{m}$ and $130\,\mathrm{m}$. Half of them can be detected at $77\,\mathrm{m}$ and $87\,\mathrm{m}$, respectively. Subsequent detections depend on the ego velocity, but can be expected to follow quickly.

When comparing the different sensor setups of 2020 and 2023, one can notice a significant improvement in recall at higher ranges. This is presumably due to the higher camera resolution, both in specific resolution and in optical acuity, and lower static parallax of the new sensor setup.

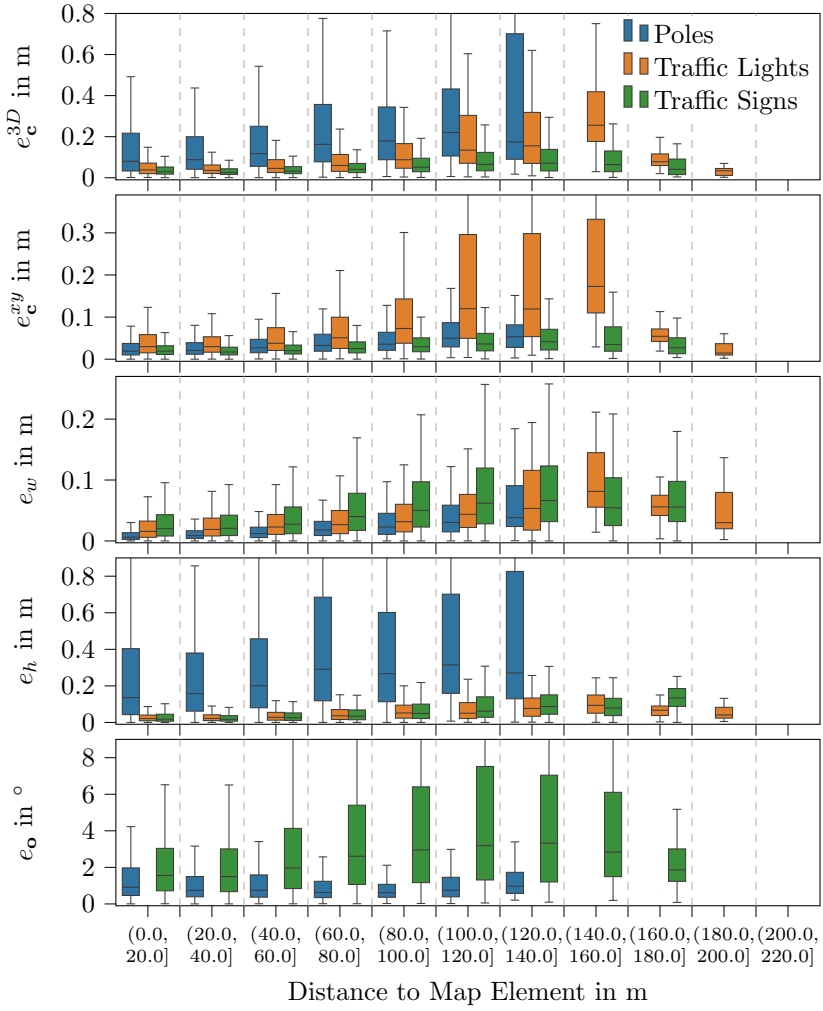


Figure 3.24: Measurement errors of the parametric detections w.r.t. the respective map element assigned during mapping, measured in the map coordinate system and using the 2020 sensor setup. Even at 120 m distance, most map elements can be measured with errors of about 10 cm. Only for poles, the height h and the z component of the center point has larger errors. The orientation error for poles is at around 1° while sign measurements deviate around 2 to 7° from the final map element.

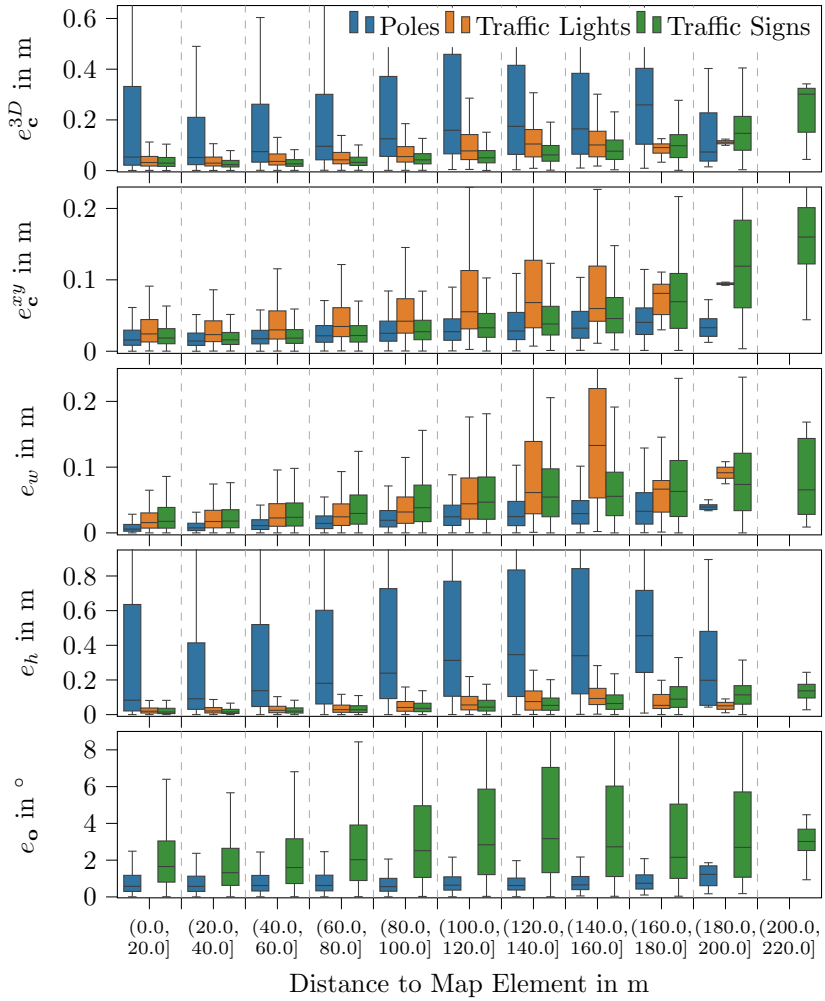
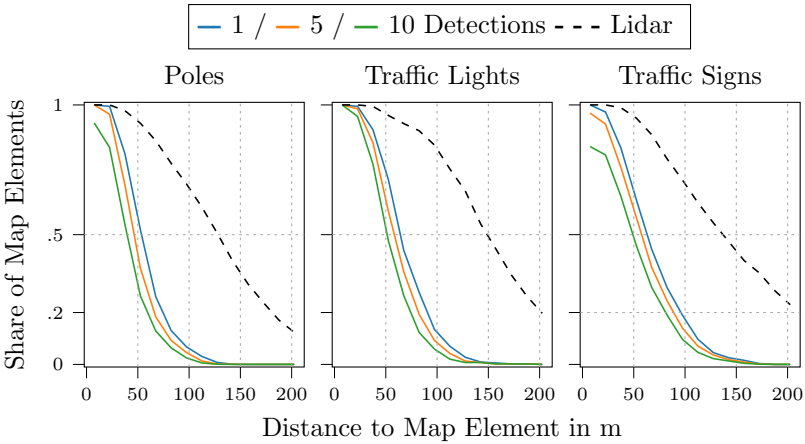
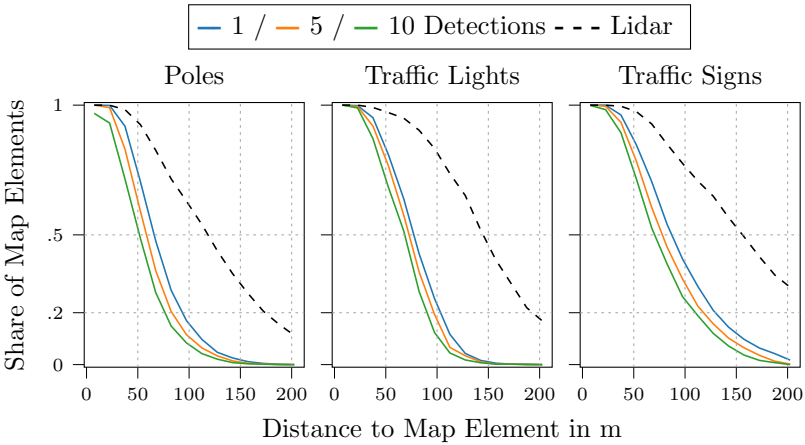


Figure 3.25: Measurement errors of the parametric detections w.r.t. the respective map element assigned during mapping, measured in the map coordinate system and using the 2023 sensor setup.
Compared to the 2020 sensor setup, detections of all classes now reach up to 180 m. Still, the errors barely increase in magnitude.



(a) Results using the 2020 sensor setup on the validation sequences *Adenauer 01* and *Moltke Big 01*.



(b) Results using the 2023 sensor setup on the validation sequences *Adenauer 01* and *Moltke Big 01*.

Figure 3.26: 1/5/10-recall of map elements over detection distance. Depicted in color is the share of map elements that were associated with at least one, five, and ten farthest detections at the respective distance.
Since during mapping some detections might not have been associated to a map element, the solid lines constitute a *lower limit* for the detection range. The dashed line indicates the farthest visibility in lidar which is a very optimistic *upper limit* for detection methods that rely on directly measured depth from lidar.

Rendering Instance IoU

Since a real ground truth is missing, in Table 3.1, quantitative results are reported using the proposed RIIoU metric as reference for future work. The average over the sequences is weighted using the respective sequence length. To make depth and size parameters observable, the instances from three frames in the future were used to evaluate the parametric detections, *i.e.* $j = 3$ in Equation (3.28).

The numbers may seem small compared to IoUs reported by state-of-the-art DNNs. However, considering the abstraction due to parametric representations, the focus on typically small objects, and the domain gap between training and evaluation data, especially the values for traffic lights and traffic signs in fact seem competitive.

When comparing the numbers from 2020 and 2023, the new sensor setup developed by the author can again show a significant advantage in RIIoU.

Table 3.1: Quantitative results using the RIIoU metric for parametric detections, reported in % and evaluated on the respective validation sequences. The metric is reported for each class, as cardinality weighted mean across classes, and as average weighted by sequence length across sequences.
TL = traffic light, TS = traffic sign.

Year	Sequence	Pole	TL	TS	∅
2020	Adenauer 01	11.9	33.0	30.7	18.6
2020	Moltke Big 01	12.7	40.7	33.9	21.5
2020	∅	12.3	37.2	32.4	20.2
2023	Adenauer 01	18.3	28.5	34.1	23.7
2023	Moltke Big 01	17.9	33.6	40.6	27.8
2023	∅	18.1	31.4	37.8	26.1

Qualitative Examples

Finally, Figures 3.27 to 3.30 show qualitative examples for the two different sensor setups from 2020 and 2023.

Figure 3.27: Qualitative examples for parametric detections using the 2020 sensor setup, inferred from a single measurement frame, *i.e.* one point cloud and one image. Detected poles are framed in blue, traffic lights in green, and traffic signs in yellow.

Figure 3.28: Qualitative examples for parametric detections using the 2020 sensor setup. Detected poles are framed in blue, traffic lights in green, and traffic signs in yellow. The signs in the lower image illustrate the accurately estimated orientation.

Figure 3.29: Qualitative examples for parametric detections, inferred from a single measurement frame, using the 2023 sensor setup.

Figure 3.30: Qualitative examples for parametric detections, inferred from a single measurement frame, using the 2023 sensor setup.

The images show that not only almost all map elements are correctly detected by the DNN and converted into parametric measurements. One can also see that even given just a single point cloud and image, not only the center points, but also the extent and orientation can be estimated with high precision. The few missing detections are caused either by absent initial detections from the DNN or poor lidar coverage. The latter holds in particular in the closest proximity of the vehicle if the target is too low or too high. Additional lidar sensors are the obvious solution, but in the close range, mono depth prediction approaches could be a very promising and more affordable replacement.

To increase the DNN performance, one could use a more modern architecture [JLC+23] pretrained on Mapillary Vistas [NOB+17] together with a, possibly iterative, finetuning on the sensor setup using *e.g.* the reprojected map produced with this approach.

Runtime Analysis

To prove the claim that the presented approach is real-time capable, the runtime to derive the parametric detections was analyzed on an AMD EPYC 7702P 64-core processor as it is used in the measurement vehicle at the MRT at the time of writing. The results, reporting average runtimes, are depicted in Figure 3.31.

Since a rather old and unoptimized DNN [PBC+19] was used as is, its performance was neither optimized nor measured. Using standard instance segmentation networks from optimized frameworks, *e.g.* a Mask R-CNN [HGD+17] from Detectron2 [WKM+19], similar detection results can be achieved in $78\,\mathrm{ms}$ even without GPU-specific optimizations. While this still makes the DNN the slowest component, the approach has been split. The slow motion and parallax compensation part can run in parallel to the DNN and builds optimized data structures which are then exploited by the sequentially running, but very fast lookup to find relevant lidar points for each instance mask.

Besides the average total runtime of $46\,\mathrm{ms}$ for all parts but the DNN, maximum quantiles of those parts with variable runtimes might be of interest. While the motion compensation has a constant processing time, parallax compensation,

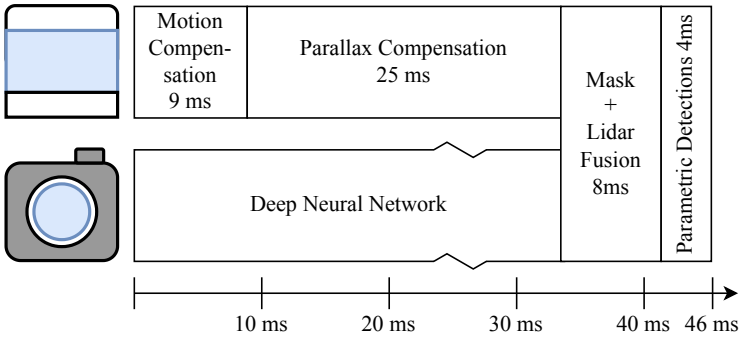


Figure 3.31: Runtime analysis of the necessary steps to derive parametric detections. As explained in the text, the DNN is neither optimized nor measured.

mask lidar fusion, and parametric detections do scale with the amount of detected map elements in the image. Their $99\,\%$ quantiles are at $37\,\mathrm{ms}$, $38\,\mathrm{ms}$, and $12\,\mathrm{ms}$, respectively. However, the last two steps could be parallelized entirely, leading to sub-millisecond average and $99\,\%$ quantile runtimes.

3.9.3 Mapping

The results of the proposed mapping approach can as well be evaluated quantitatively and qualitatively.

Rendering Instance IoU

As a ground truth is missing, Table 3.2 states quantitative results using the RIIoU metric with a maximum detection range of $150\,\mathrm{m}$ as reference for future approaches. Of course, this is now for the map instead of single shot detections.

When comparing Table 3.1 and Table 3.2, one can observe that poles benefit from mapping in the RIIoU metric while traffic lights and road signs show lower numbers. This might be explained by two facts. First, parametric detections seem to effectively exhaust the DNN's detection capacity. Hence, the missing share in detection RIIoU is due to the simplification by parametric representations

and the variation in the DNN's recall. Additionally, the map's RIIoU numbers suffer from map elements in the far distance, which are correct from a human perspective, but lower the metric as they are not yet detected by the DNN. To improve upon this, a detection probability that decreases over distance, similar to the recall plots in Figure 3.26, could be included.

Table 3.2: Quantitative results using the RIIoU metric for the HAD map, reported in % and evaluated on the respective validation sequences. The metric is reported for each class, as cardinality weighted mean across classes, and as average weighted by sequence length across sequences.
TL = traffic light, TS = traffic sign.

Year	Sequence	Pole	TL	TS	∅
2020	Adenauer 01	18.2	24.4	28.3	21.6
2020	Moltke Big 01	18.9	25.2	28.4	22.5
2020	∅	18.6	24.8	28.4	22.1
2023	Adenauer 01	24.7	27.5	32.2	27.2
2023	Moltke Big 01	22.6	29.2	35.4	27.6
2023	∅	23.5	28.5	34.0	27.4

Qualitative Examples

Figures 3.32 to 3.35 show qualitative examples. Using the same poses used to create the map, all map elements within 200 m distance are projected into the camera images.

Figure 3.32: Qualitative examples for a parametric HAD map using the 2020 sensor setup. Poles are framed in blue, traffic lights in green, and traffic signs in yellow.

Figure 3.33: Qualitative examples for a parametric HAD map using the 2020 sensor setup. Poles are framed in blue, traffic lights in green, and traffic signs in yellow.

Figure 3.34: Qualitative examples for a parametric HAD map using the 2023 sensor setup. Poles are framed in blue, traffic lights in green, and traffic signs in yellow.

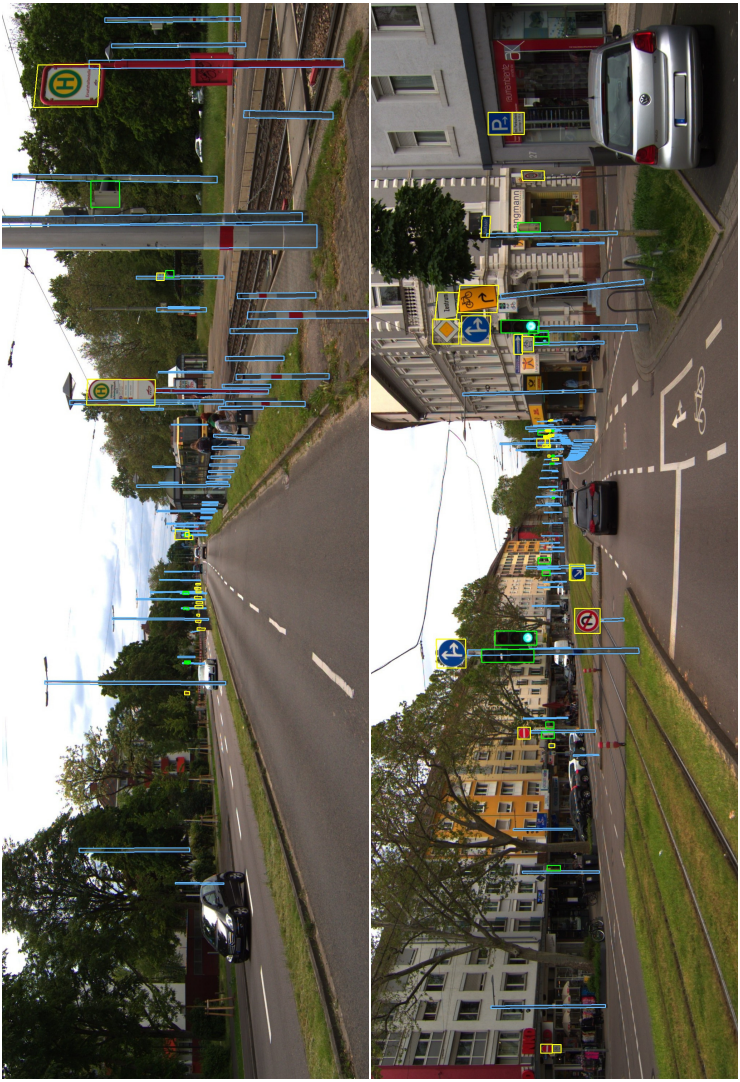


Figure 3.35: Qualitative examples for a parametric HAD map using the 2023 sensor setup. Poles are framed in blue, traffic lights in green, and traffic signs in yellow.

Accuracy against Ground Truth

Due to the lack of ground truth, one can only state precision of the detections w.r.t. map elements, but no true accuracy. However, in a previous publication [PSS21], a very similar approach was evaluated on a few sections of the *Ostring 01* sequence using manually annotated ground truth and the 2020 sensor setup. In contrast to the paper, the approach proposed in this work now allows tilted poles and uses KISS-ICP [VGM+23] as odometry source, both of which unfortunately makes the manually annotated ground truth incomparable.

However, not only the parametric representation and the source of odometry poses were improved, but also the Hungarian association was replaced with a graph-based data association and applied a hyperparameter optimization. Hence, it is suggested to view the numbers in Table 3.3 as pessimistic upper limits of the accuracy achievable with the method presented in this thesis.

When comparing the numbers with a DNN based approach that uses virtual lidar scans with equal density [PSF+23], reported in Table 3.4, one can see that the approach proposed in this thesis can achieve similar results in all parameters but pole center point z coordinate. The latter can be explained by the fact that Plachetka et al. only predict arguably much simpler base points and neglect any height estimation for poles.

While the accuracy for the approach proposed in this work is measured for mapped elements instead of individual detections, the numbers are from real lidar scans which suffer from occlusions, unregular point and beam spacing, and a point density that decreases quadratically over distance. Additionally, as Figures 3.24 and 3.25 show, the performance of the proposed approach can be expected at even more than $100\,\mathrm{m}$ distance which twice the range used by Plachetka et al.

Table 3.3: Accuracy of the optimized map against manually annotated ground truth as used in a previous publication [PSS21]. The first three columns report the RMSE, computed using the results from [PSS21] for comparison with Plachetka et al. [PSF+23]. The last six columns are stated as mean absolute error (MAE) as originally done in the previous publication [PSS21].
Due to the improvements since then, it is suggested to read the numbers as upper limits for the method proposed in this thesis. Errors of center point coordinates, width and height are reported in meters; error in orientation is reported in degrees.

	RMSE		**MAE**					
	$e_{\boldsymbol{c}}^{3D}$	$e_{\boldsymbol{c}}^{xy}$	$e_{\boldsymbol{c}}^{x}$	$e_{\boldsymbol{c}}^{y}$	$e_{\boldsymbol{c}}^{z}$	e_w	e_h	$e_{\boldsymbol{o}}$
Traffic Signs	0.15	0.13	0.09	0.07	0.03	0.03	0.06	5.6
Traffic Lights	0.17	0.16	0.11	0.08	0.03	0.04	0.03	-
Poles	0.45	0.17	0.12	0.10	0.33	0.06	0.62	-

Table 3.4: Errors reported by Plachetka et al. [PSF+23] using a DNN trained and evaluated on virtual lidar scans. Center point position errors are given as RMSE while width, height, and orientation errors are reported as MAE. Accuracy is measured against the test set of the 3DHD CityScenes dataset [PSF+22]. Errors of center point coordinates, width and height are reported in meters; error in orientation is reported in degrees.

	RMSE	**MAE**		
	$e_{\boldsymbol{c}}^{3D}$	e_w / e_d	e_h	$e_{\boldsymbol{o}}$
Traffic Signs	0.11	0.09	0.12	12.3
Traffic Lights	0.08	0.04	0.08	14.0
Poles	0.11	0.06	-	-

3.10 Limitations

Regarding the parametric detections, there are three major limitations. First, in this thesis only three parametric representations have been presented. However, many more are conceivable, *e.g.* single lane markings and manhole covers could be represented as flat rectangles on the ground. Non-confined map elements, like solid road markings or curbs, are harder to represent parametrically, but a recent approach [MSP+21, Mey23] proposed a piecewise discretization that could be used in BEV.

The second limitation is due to the DNN employed for detecting objects in the first place. It supports relatively few semantic classes and cannot distinguish *e.g.* various kinds of traffic signs. Furthermore, it currently is the major limitation of detection range. Using a dataset with higher semantic resolution with more recent high resolution approaches [CMS+22] could alleviate both issues significantly.

Lastly, the lidar coverage in the upper half of the image is an issue when detecting *e.g.* traffic lights that are mounted above the street. To increase the coverage, either additional sensors or mono depth prediction approaches are conceivable.

While the mapping works fine for most objects that are visible for at least a few frames, heavily occluded map elements are still an issue. To reduce occlusion and increase visibility, the remaining five surround "ring" cameras could be utilized as well. Extending the framework to support multiple drives, possibly crossing intersections via all arms, would perfect the map.

Another issue during mapping are signs which are composed of multiple parts. While the DNN distinguishes the parts as individual signs in the close and mid range, from far away it often detects one large sign. This could be mitigated using an appropriate, possibly hierarchical sign representation, possibly in combination with appropriate semantic classes.

To improve the Rendering Instance IoU metrics, the author proposes to incorporate uncertainties. The object detections could be weighted using calibrated detection probabilities from the DNN. For map elements, which are currently penalized for not being detecting when too far away, a detection probability

that decreases over distance is expected to yield the largest improvement. Both should improve both expressiveness as evaluation metric, *i.e.* being in line with subjective map quality, as well as efficiency as loss for hyperparameter optimization or a DNN training.

Finally, an alternative to a classical fusion approach whose hyperparameters are optimized in a data-driven way would be to train a DNN that detects parametric map elements directly in raw sensor data. The current framework offers two possibilities. First, a standard 3D/BEV object detection approach could be modified to predict the parameters of element representations. Here, the fully automated nature of the mapping framework combined with the exceptional accuracy makes the HAD map an easily scalable pseudo ground truth. The alternative is to use the novel RIIoU metric based on instances as pseudo-labels. Here, differentiable rendering [KBM+20, FSL+22] can bridge the gap between predicted parametric representations and the instance masks used for evaluation.

3.11 Conclusion and Outlook

In this chapter, an approach to detect parametric map elements using a fusion of object instances from a DNN used on camera images and lidar points was proposed. The key innovation is to exploit the knowledge of the object semantics predicted by the DNN to choose a class-specific minimal parametric element representation that can be estimated precisely while having sufficient fidelity for a unique data association. This makes it possible to achieve parametric detections with an unprecedented level of precision at detection ranges that allow to react comfortably in advance on the presence or absence of map elements. At the same time, parametric detections can be derived in real time.

The parametric detections are not only suitable for localizing in or verifying HAD maps. They are also a great foundation to automatically create them. By leveraging a state-of-the-art object tracking approach detections can be associated successfully in the parameter space over more than a hundred frames and across major appearance changes. A subsequent optimization yields parametric HAD maps with an accuracy that matches human annotations.

Not by coincidence, the parametric representations are also well-suited to render the map elements into a camera image. First, this is the foundation of a novel family of metrics, called Rendering Instance IoU, that combines the ideas of weak and self supervision. By comparing map element projections with instance masks as pseudo labels, it enables to evaluate the performance of the detection and mapping framework without the need for ground truth. To avoid manual tuning and achieve a data-driven approach, it was shown that the metrics can be combined with state-of-the-art hyperparameter optimization techniques.

Beyond the scope of this thesis, the suitability of the parametric representations for rendering enables the use of HAD maps as automatically generated pseudo ground truth. This is not only the basis for retraining the DNN that currently limits the detection range. Together with the proposed RIIoU metric for automated hyperparameter tuning, it is also the stepping stone for a self-optimizing map perception and mapping framework.

4 Verifiably Optimal Probabilistic Data Association and Localization

Correctly associating detected map elements to a map, although it might have become incomplete or contain outdated parts, is crucial to safely benefit from HAD maps, *e.g.* in terms of localization or traffic light association. But it is also *the* core issue to verify the physical map layer. However, the deployment of localization approaches in robotics or complex ADAS functions and recent research have shown that the localization problem as well as the inherently coupled data association issue are far from being solved.

Old Problem, New Challenges

Early localization approaches determined the ego pose of vehicles or robots in a 2D world, describing the it by x, y coordinates in the plane and a yaw angle around the up axis. HD maps often contain map elements not only in 2D, but also their height above ground or elevation above a reference geoid. Fully exploiting 3D data is particularly important when projective geometry is involved, like for long-range camera perception, head-up displays or learning from maps [BHS+23]. This makes it necessary to resolve the pose of the ego vehicle in six dimensions: a 3D position as well as a rotation in 3D space.

Solving 6D localization given a perfect association between detected and mapped elements is trivial [AHB87, Hor87, Ume91] as is the opposite, optimal data association given a perfect 6D localization, at least when assuming suitable measurement models [Mun57, JV87]. But, when detections are noisy, may contain clutter and possibly even the map is outdated, neither a perfect association nor a sufficient localization result can be assumed. This presents a difficult

chicken-and-egg problem that needs to be solved correctly in order to verify which map elements are up-to-date, but also to localize successfully in a potentially partially outdated map.

One possible solution is to introduce a localization layer which allows storing sensor-specific information either densely or via feature descriptors. The density and/or uniqueness of the data enables localization at high availability and with comparatively little computational effort. However, localization layers have two drawbacks. First, they always involve additional storage or transmission overhead. The data density can reach from "five kilobytes [...] per kilometer" [Ebb17] for an accuracy of "a few decimeters" [Ebb21] to many megabytes if more accurate localization is required (*cf*. Section 4.1.3). The second, possibly more severe challenge, is to reference the localization layer accurately with the relevant semantic planning information. In contrast, localizing directly in HAD maps, which are required for automated driving anyways, comes without any overhead or referencing errors.

Unfortunately, localization in a semantic HAD map comes with its own difficulties. First, as observed in Chapter 2, transitions between unchanged and changed areas are often intentionally made smooth. Hence, especially filter-based localization approaches are prone to gradual deterioration, rendering localization integrity a serious issue. One way to tackle it is to provide a self-assessment [SRD17b, Stü18], which will also be referred to as *verifiability* throughout this thesis.

Another option is to prevent deterioration due to gradually outdated environments by decorrelating individual localization results as much as possible. This can be achieved by using single shot solutions that ideally do not even require an initial 6D pose as prior. Combined with self-assessment, they can avoid deterioration entirely by withholding the localization output when gradual changes lead to ambiguities.

Ambiguities are also a particular problem of maps containing man-made structures since the ambiguity of their local constellations is a well-known problem [HSS+19, HSR+20]. Repetitive patterns of street lights, poles, guardrail mounts, or road markings can inhibit a unique localization given only local

observations. Previous publications of the author showed that verification of such repetitive environments can be achieved without resolving this ambiguity using specialized approaches [PSH+20a, PSH+20b, PSH+21]. However, to uniquely localize the ego vehicle and, hence, attribute changes and verification results to individual map elements, one needs to find a way to resolve ambiguities, if possible. If a unique solution is impossible, the data association procedure should be able to self-assess this.

A commonly used way to estimate both the optimality of a solution and its uniqueness is to use a probabilistic model. Using such a model to describe the distribution of the input data and a method that is aware of those probabilities allows obtaining the (relative) probability of the outcome. This mathematical soundness makes probabilistic methods superior to other, *e.g.* heuristic, approaches.

In summary, a desired approach needs to solve the complex data association problem i) optimally, ii) in presence of noisy detections and clutter as well as iii) map changes, iv) without requiring an initial pose. At the same time, the approach should be able to v) detect ambiguities and vi) self-assess its performance vii) using a probabilistic model, and viii) do so in real time. As will be discussed in the related work, existing solutions cover some of those requirements, but the author is not aware of any approach that fulfills them all.

Contributions

This chapter presents **a novel framework for data association** called probabilistic correspondence graph (PCG), depicted in Figure 4.1. It allows us to model and solve the data association problem probabilistically without the knowledge of an initial pose.

Novel probability distributions of transformation invariant measurements, such as differences of Euclidean distances, allow to **formulate a probabilistic maximum likelihood problem for data association problem independently from the transformation** between both data sets. It is shown how it can be rewritten into the problem of retrieving the maximum totally weighted clique in

a log likelihood-weighted correspondence graph, the eponymous probabilistic correspondence graph.

A state-of-the-art maximum weighted clique solver is adapted to support real-valued weights and output multiple hypotheses. For the first time, this enables the **retrieval of exact probabilistically optimal solutions in real time**. Retrieving not only the best, but all possible cliques within a certain log likelihood window makes it possible to self-assess solutions probabilistically and, hence, to provide **probabilistic guarantees for the uniqueness of the optimal solution**.

When used to solve the localization problem for automated vehicles, the proposed method can **provide probabilistically optimal solutions in real time using a single frame of measurements**. Via self-assessment the localization output can be consciously inhibited instead of providing wrong results.

As theoretical contribution, it is examined how transformation invariant measures work mathematically and what this implies for their probability distributions. The resulting probabilistic correspondence space is compared with commonly used probability measures when knowing the transformation between data sets, empirically showing high correlation. This thesis also presents measures to overcome differences between both spaces and necessary geometric conditions for their alignment.

Experiments show that the proposed data association approach **outperforms the state of the art on small and medium-sized data association problems** like isometric feature matching. For point cloud registration on the KITTI benchmark, **PCG outperforms all previous methods** and even surpasses that of deep learning approaches.

When localizing an automated vehicle in a sparse and compact semantic HD map using PCGs achieves an **average accuracy of about** $2\,\mathrm{cm}$ **and** $0.02°$**, which previously required significantly denser sensor-specific localization layers**. While requiring only a single frame of measurements, the proposed method outperforms previous localization approaches in sparse semantic maps which use particle filters or graph optimization. Hence, PCG represents a new state of the art for data association, especially when probabilistic outcomes,

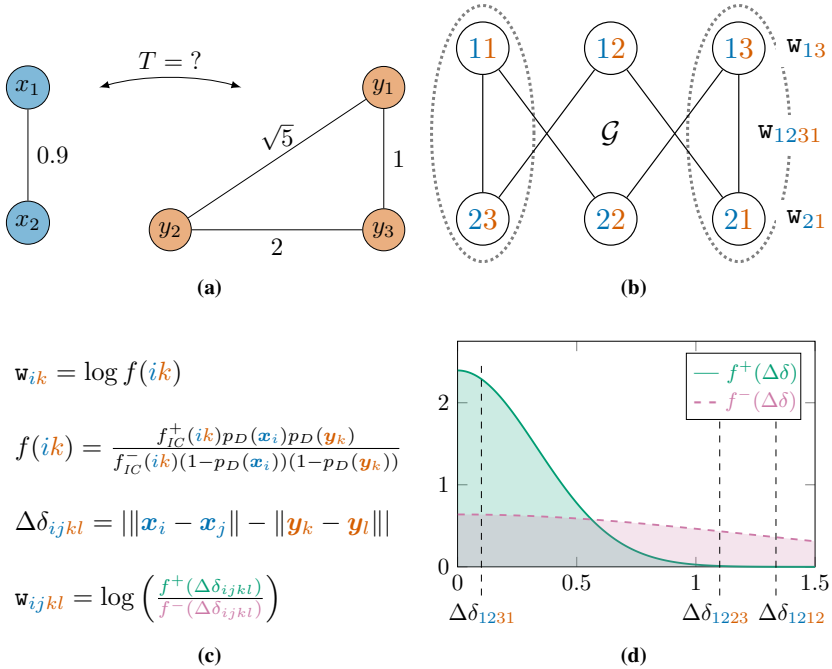


Figure 4.1: Illustration of the concept of probabilistic correspondence graphs (PCGs) using a toy example:
(a) The goal is to associate the two sets of measurements, *e.g.* the blue points $\boldsymbol{X}$ to the orange points $\boldsymbol{Y}$, without any knowledge on the transformation T between them. Edges between points are labeled with pairwise distances.
(b) Each putative correspondence (x_i, y_k) forms a vertex in a weighted graph $\mathcal{G}$, called probabilistic correspondence graph (PCG). Vertices are connected by edges if and only if the correspondences are potentially compatible and they are not mutually exclusive, *e.g.* due to requiring a one-to-one assignment.
(c) Vertices are weighted with the relative log likelihood of the correspondences, $\mathtt{w}_{ik}$, describing detection probabilities p_D and individual compatibility likelihoods f_{IC}. Edges in $\mathcal{G}$ are weighted with the relative log likelihood of the pairwise compatibility of the respective vertices, $\mathtt{w}_{ijkl}$.
(d) Pairwise compatibility can exemplary be measured by the difference of Euclidean distances $\Delta\delta_{ijkl}$ whose distribution can be approximated by a half-normal distribution for each inliers, f_+, and outliers, f_-.
The optimal assignment(s) can then be retrieved as clique(s) of maximum total weight. In this example, the two cliques encircled with dashed lines in $\mathcal{G}$ have the same likelihood. More points would be necessary to resolve the rotation ambiguity, but are omitted for the sake of clarity.

optimality guarantees, or uncorrelated localization results are important, such as in safety-critical systems.

As additional contribution, and in fact independently from the concept of PCGs, **a novel joint Euclidean angular probability distribution** is proposed for the association of oriented objects. Simulation experiments show that it outperforms the previous state of the art, affine Grassmannian distances [LH22, LPH23].

The idea of **weakly and self-supervised metrics** presented in Chapter 3 is extended towards data association and localization. This makes it possible to overcome the lack of a sufficiently accurate ground truth for hyperparameter optimization and evaluation.

Previous Publications

A conceptionally similar initial idea for data association has been published previously in a workshop paper [PS22]. Related ideas that show how to extend existing data association methods for associating and localizing via line-like map features have been published with minor contributions by the author [MPH+22a, MPH+22b]. Transferring the ideas to the approach presented in this work is straightforward, but was omitted due to the lack of a suitable feature detector.

4.1 Foundations and Related Work

Before presenting related work, it is necessary to establish the basis for the different fields covered in the course of this chapter. For a comprehensive introduction to graph theory, the reader is referred to Diestel [Die17]. The idea of data association with constraints has been described by Grimson [Gri90]. Thrun et al. [TBF05] cover the basics of the SLAM problem and the challenges of data association. Finally, Mahler [Mah07, Mah14] published comprehensive works in the direction of probabilistic object tracking, serving as framework to model probabilistic data association.

Related work on data association for mapping, localization, or the generalized SLAM problem can be divided into general concepts, reviewed in this section, and approaches to formulate the registration of oriented objects, which will be discussed in the next section. State-of-the-art approaches for highly accurate localization are introduced in Section 4.1.3.

4.1.1 Data Association

To help putting the data association approaches into context, first, the particular requirements and the resulting combinatorial explosion as major challenge are described.

Requirements

Depending on the problem at hand, the desired data association concept needs to fulfill different requirements. When temporally uncorrelated solutions are desired, putative approaches must work without an exact prior pose.

Real data typically contains noise and outliers, *e.g.* from false detections or outdated landmarks. While basically all approaches can handle noisy data, in this thesis, only approaches that can also deal with outliers are discussed.

Man-made structures are often ambiguous in their local patterns [HSS+19]. However, not all approaches are made to cope with ambiguities.

For many applications in safety-critical systems, a mathematically proper, hence usually probabilistic, model is required. To certify safety it is necessary to either fulfill optimality, provide a self-assessment, or both.

Finally, when the data association is involved in interactions with the real world, solutions need to be available in real time.

In Table 4.1, the relevant concepts are listed together with references and their performance w.r.t. these requirements.

Table 4.1: Overview over existing data association concepts, the proposed method, and their performance w.r.t. five important requirements. (Soft) real-time capability is evaluated for typical problem sizes.
Optimality criteria: ML: Maximum likelihood (with strongly varying assumptions). JC: Joint compatibility. #PC: Largest pairwise compatible subgraph. ρPC: Densest pairwise compatible subgraph. R: Minimal residual. #I: Inlier count. DG: Duality gap.

Concept	Related Works	Prior free	Ambiguity aware	Probabilistic	Real-time capable	Globally Optimal	Criterion
Filtering Approaches	EKF-SLAM			✓	✓		ML
	FastSLAM		✓	✓	✓		ML
	RFS-SLAM		✓	✓	✓		ML
Joint Compatibility	JCBB, IPJC, FastJCBB			✓	✓	✓	JC
	MHJCBB		✓	✓	✓	✓	JC
Pairwise Compatibility	MCS	✓	✓[a]		✓	✓	#PC
Densest Subgraph	CLIPPER	✓			✓	✓	ρPC
Iterative Closest Point	ICP				✓		R
	Go-ICP	✓				✓	R
Hypothesize-and-Test	RANSAC	✓	✓[b]		−✓[c]−		#I
	MLESAC	✓	✓[b]	✓	−✓[c]−		ML
	MAGSAC	✓	✓[b]	✓	−✓[c]−		ML
Certifiable Optimization	TEASER	✓	[d]		[e]	✓	DG
	[YC20, YC23]	✓				✓	DG
PCG (Proposed)		✓	✓	✓	✓	✓	ML

[a] Can be implemented as proposed in a previous publication by the author [PS22].
[b] Implementable by tracking multiple best models.
[c] Continuous trade-off between compute time and optimality.
[d] Assumptions for optimality exclude nearly identical ambiguous solutions.
[e] While optimization is real-time capable, optimality certification is not.

Combinatorial Explosion as Challenge

The main challenge to solve the data association problem is the combinatorial explosion: There are about $\binom{n}{m}$ possibilities to associate m measurements with n map elements when, due to misdetections and map changes, both are uncertain to have a correspondence.

To animate these numbers, for a typical urban scenario with 20 measurements and 50 landmarks, this yields 47 trillion possibilities. Clearly, even with modern hardware, these many combinations cannot be explored entirely.

Hence, over time, several strategies emerged to avoid an exhaustive exploration. Filtering approaches use knowledge from a previous time step to reduce the number of combinations and to resolve uncertainties over time. Hypothesize-and-test approaches, like random sample consensus (RANSAC), sample minimal solutions that can then be evaluated. Solving the association problem by certifying the optimality of the transformation that aligns measurements and landmarks is the core idea of certifiable optimization methods. Finally, previous approaches around the ideas of joint compatibility and binary consensus proposed to exploit transformation invariant conditions to guide the search for solutions without computing transformations at all.

Filtering Approaches

When a pose *e.g.* from a previous time step can be assumed, conceivable solutions to the data association problem are restricted considerably, usually by several orders of magnitude. Early approaches, *e.g.* extended Kalman filter (EKF)-SLAM, used the Mahalanobis distance to compute a maximum likelihood association given the previous pose estimate [TBF05]. However, the underlying assumption of a unimodal Gaussian distribution is rarely fulfilled in the real world.

FastSLAM's breakthrough of factorizing the posterior distribution of the SLAM problem into a path and a map factor enabled the use of particle filters in real time [MTK+02, MT03, TBF05]. This allows individual associations for each

particle and, hence, multiple hypotheses for the poses and map that can then be used to detect, track, and resolve ambiguities.

The emergence of random finite sets (RFSs) and random finite set (RFS)-based filters allowed to capture multiple hypotheses even better [MVA+11, DRD15, Deu16, SRD17b, SRD17a, Stü18]. However, the underlying association mechanisms, *e.g.* ranked assignment or sampling, track only the best k hypotheses.

In order to avoid a pose prior and potential drift problems caused by it, this thesis refrains from filters to solve the data association challenge. Instead, the method presented in this chapter may be used as comprehensive front end for a SLAM filter.

Joint Compatibility

Traditional association methods, like (gated) nearest neighbors using Mahalanobis distance, do not incorporate the correlation of the individual measurements' errors [NT01]. As solution, the so-called *joint compatibility* criterion was proposed [NT01]. It is formulated as χ^2 innovation test over the squared Mahalanobis distance [Bai02], also referred to as normalized innovation squared (NIS) [Bar01], of a putative subset of all possible associations. It can be evaluated greedily, called sequential compatibility nearest neighbor (SCNN) association.

As alternative, Neira and Tardos also proposed the joint compatibility branch and bound (JCBB) method [NT01]. It searches the interpretation tree [Gri90] and performs the joint compatibility test for each putative additional association. While scaling vastly worse than SCNN, its restrictive nature proved to be successful even in situations with significant outliers and clutter. In particular, the share of false associations is significantly lower.

In addition to the combinatorial explosion of the interpretation tree, which is only confined by the bounding of the branch-and-bound search, each explored search step requires $O(m^2)$ operations for m associated features. To mitigate this effect, incremental posterior joint compatibility (IPJC) [LO12] and FastJCBB [SFR+16] were proposed, simplifying the computational cost for each test to $O(m)$ and $O(1)$ operations, respectively, by exact reformulation.

Recently, as even more restrictive alternative, conditional compatibility branch and bound (CCBB) [SAR18] has been proposed. In addition to the joint compatibility, it also tests if a putative association should be accepted given the existing associations, hence, limiting not only the total matching cost, but its change with the putative association. Using the fast joint compatibility test from [SFR+16] as foundation, it is two to three orders of magnitude faster than the original JCBB.

All those simplifications cannot solve the core issue of joint compatibility, the combinatorial explosion of the search tree. Hence, for this thesis, joint compatibility approaches were not pursued any further.

If computation time was not an issue, the multi-hypothesis extension MHJCBB [WE18] may provide a useful benefit. It tracks multiple hypotheses ordered first by size and second by Mahalanobis distance. Despite clever ideas, tracking multiple hypothesis increases complexity even further. Hence, even in scenarios with just 32 landmarks only three hypotheses could be tracked in real time [WE18].

Pairwise Geometric Compatibility

Joint compatibility approaches either require a prior pose or need to fully explore a vast combinatorial search tree using a branch-and-bound technique. A fast, yet prior free alternative is to require each putative pair of correspondences to be compatible given their geometric properties, motivating the term pairwise or geometric compatibility. The reasoning behind this idea is that isometric transformations do not change mutual information, such as distances or relative angles. Hence, matching pairs of correspondences need to have equal distance up to noise.

A typical formulation for the *pairwise geometric compatibility* of two correspondences $(\boldsymbol{x}_i, \boldsymbol{y}_k), (\boldsymbol{x}_j, \boldsymbol{y}_l)$ is given by

$$\left| \|\boldsymbol{x}_i - \boldsymbol{x}_j\| - \|\boldsymbol{y}_k - \boldsymbol{y}_l\| \right| < \tau_{PC} \tag{4.1}$$

with τ_{PC} being the compatibility threshold.

This formulation has been reinvented a number of times. Initially, it was proposed as one of many possible constraints for the purpose of object recognition, nicely contextualized by Grimson [Gri90]. One perspective is to use it for searching a so-called interpretation tree that matches elements from each data set [Gri90]. However, already early on, Bolles and Cain [Bol79, BC82] proposed to use the mutually compatible correspondences as vertices in a so-called *compatibility graph*. These vertices are connected by an edge if and only if they are pairwise geometrically compatible. Since the geometric compatibility needs to hold for all correspondence pairs involved in a valid association, one can search the compatibility graph for the largest clique, *i.e.* fully connected subgraph. This so-called maximum clique search (MCS) is substantially faster than exploring the interpretation tree [Bai02].

For the application of robot localization, Bailey et al. [BNR+00] were the first to describe a similar formulation, coining the term *correspondence graph* which stuck in the field of robotics. Later, Bailey [Bai02] referred to the previous works in the field of object recognition. Mangelson et al. [MDE+18] proposed a virtually identical idea a third time for the application of map merging.

In principle, this binary pairwise geometric compatibility is the foundation of the probabilistic correspondence graph proposed in this thesis. Indeed, in a previous publication by the author [PS22], it was proposed to use maximum clique search as data association method. Therein, it was also described how ambiguities can be captured with a simple extension of a standard maximum clique search algorithm [BK73]. In this thesis, instead of maximizing the cardinality of the correspondence clique, its likelihood is maximized. This allows including a probabilistic model and, for the first time, provides a probabilistically globally optimal result without requiring a pose prior.

When comparing compatibility concepts, pairwise geometrically compatible associations are not necessarily jointly compatible [Gri90, NTC03]. Hence, it has been proposed to add a joint compatibility verification step after retrieving the largest pairwise compatible candidate solutions [NTC03]. While it did not show any difference in results for the 2D SLAM scenario examined by Neira et al. [NTC03], the data used in this thesis in fact exhibited a discrepancy in a few

situations. This issue is discussed again in Section 4.2.5, where an explanation and conditions for equality of both compatibility terms are provided.

Recently, both relaxations and stricter versions of pairwise geometric compatibility have been proposed. Retrieving only a k-core of the correspondence graph, *i.e.* a subgraph $\mathcal{K} \subset \mathcal{G}$ with minimal degree $k < |\mathcal{K}|$, can be 2 to 3.4 times faster and, while being overoptimistic in general, remove sufficiently many outliers for best effort solvers to succeed [SYC21]. In contrast, searching for group-k consistent subgraphs, *i.e.* graphs wherein each subset of k vertices is jointly consistent, can significantly lower the false positive rate, but comes at the cost of being about three orders of magnitude slower than pairwise geometric compatibility [FVK+22].

Densest Subgraphs and Spectral Methods

A closely related family of approaches retrieves subgraphs with maximal properties inspired by the spectral characteristics of the correspondence graph. In contrast to the previous methods using pairwise compatibility, this introduces non-binary and even non-integer quality measures based on the real-valued adjacency matrix, also called affinity matrix, that is assumed symmetric and positive. Leordeanu and Hebert [LH05] used spectral methods to find the subgraph with maximal affinity. While being very efficient and yielding good results, this offers no guarantee of optimality.

With CLIPPER, Lusk et al. [LFH21] proposed to use the densest subgraph of mutually consistent associations, *i.e.* the densest clique. This combines the idea of using a weighted measure of compatibility with the NP-hard requirement of retrieving a clique. They provide a relaxation that has proven optimality conditions for binary adjacency matrices, but omit a proof that extends to the actually interesting weighted case. If such a proof would exist and hold even for negative affinities as they occur when using log likelihoods, this would make CLIPPER a very tempting solver for the problem stated in this thesis. However, making CLIPPER ambiguity aware is not straightforward. Still, in its original, non-probabilistic version it is later used as one of the baselines for this thesis.

Iterative Closest Point

As local method, the iterative closest point (ICP) algorithm [BM92] uses an initial pose to find nearest neighbor correspondences that can then be used to refine the transformation and thereby initiate the next iteration, with possibly new nearest neighbors, until convergence. Instead of minimizing the pointwise L^2 residual, like ICP does, G-ICP approximates the local distribution of each point cloud and uses it to perform a local maximum likelihood step [SHT09]. Still, even according to its authors, it is only halfway towards a fully probabilistic model.

Both ICP and G-ICP crucially depend on an initial pose. This motivated Go-ICP [YLJ13, YLC+16] which wraps the ICP algorithm with a branch-and-bound search of a predefined subspace of $\mathrm{SE}(3)$, yielding global optimality within that volume. Unfortunately, Go-ICP's runtime is typically several seconds even for smallest problems, preventing its use in real-time systems.

Hypothesize-and-Test

With the idea of random sample consensus (RANSAC) [FB81], a completely different family of methods has been founded. The core idea is to sample a minimal number of correspondences to compute a transformation hypothesis that can then be applied and evaluated. Initial ideas for evaluation include the cardinality of inliers yielded from gated nearest neighbors or their joint compatibility [NTC03].

Torr and Zisserman [TZ00] proposed an robust M-estimator version, MSAC, and a maximum likelihood variant, MLESAC, which are both closely related. However, like the original RANSAC, they still crucially depend on a correctly chosen inlier threshold. This drawback has been alleviated with MAGSAC(++) [BMN19, BNI+20, BNM22], which marginalizes both model and inlier likelihood over a finite range of thresholds. This yields state-of-the-art results within the family of hypothesize-and-test methods and, thus, is used as baseline for evaluation of this work.

The previously introduced idea of geometric compatibility can efficiently guide the sampling of RANSAC. This has been proposed a number of times. For correspondences pairs in 3D [QY20], correspondence triplets in 3D [YHQ+22], and, in a work co-authored by the author of this thesis, for correspondence pairs on lines [MPH+22a].

Certifiable Optimization

The most recently proposed group of methods for robust data association has evolved around the idea of certifiable algorithms. In the first approach, called TEASER(++), Yang et al. [YC19, YSC21] proposed to combine a scale and transformation estimation, robustified via truncated least squares (TLS), with a subsequent certification based on Douglas-Rachford splitting.

For inlier selection, they seem to reinvent the idea of geometric consistency, using a perspective closely related to the concept of transformation invariant measurements (TIMs) introduced in Section 4.2.3. This allows searching for the largest binary consensus clique or k-core. While the transformation estimation via graduated nonconvexity (GNC) is really efficient, the certification is not real-time capable for many real-world registration problems. The appealing combination of state-of-the-art solution quality at fast computation times with a separate, slower certification motivated the use of TEASER++ as one of the baselines for this work.

In a more general approach, a sparse semidefinite programming (SDP) relaxation has been proposed [YC20, YC23]. While being very generic and globally optimal, it is not even remotely real-time capable. The same publications also again address the idea of certifying results from fast, but not necessarily optimal solvers, like GNC or RANSAC. Experiments show that certifying optimality for typical problems takes tens to hundreds of seconds.

Best Effort Point Cloud Registration

The availability of datasets like 3DMatch [ZSN+17] and 3DLoMatch [HGU+21] enabled widespread research on large-scale point cloud registration problems.

Given a myriad of publications on best effort approaches without guarantees, only the few main innovations are highlighted.

First, to reduce the vast amount of points and potential matches, feature descriptors are computed. Classical features, like fast point feature histograms (FPFHs) [RBB09], were quickly outperformed by learned ones, like fully convolutional geometric features (FCGFs) [CPK19] or Predator [HGU+21]. The availability of transformation ground truth also enabled approaches that integrate more and more parts of the registration pipeline in a (deeply) learned fashion [BLZ+21, QYW+22, QYW+23].

For the classical matching of feature point correspondences two ideas are of interest. The first is using spectral methods on a graph that measures binary spatial compatibility of second order [CSY+22, CSY+23]. This means that the compatibility of two correspondences is measured by the (integer) number of other correspondences that are pairwise compatible with each of the two correspondences in question. It is shown probabilistically that measuring compatibility at second order vastly facilitates distinguishing inliers and outliers. The second order affinity matrix can then by evaluated using spectral matching which renders it a best effort approach without guarantees. Using graphics processing unit (GPU) parallelization, it is real-time capable even for a huge number of correspondences and forms the state of the art for point cloud registration at the time of writing. While the rather complex spectral matching pipeline is debatable, the general concept of using second order compatibility seems to be a very promising idea.

A recent, award-winning paper proposed to compute the maximal cliques (plural!) of a compatibility graph which can then be evaluated by computing the corresponding transformation and residuals [ZYZ+23]. Even claiming to be real-time capable, this seems like the holy grail of data association. Unfortunately, Zhang et al. fail to mention a very essential detail: They require a massive pruning of the first or second order compatibility graph using a number of metrics inspired by network clustering [YZF+23]. Without the heuristic graph pruning, which voids all potential guarantees of optimality, the set of maximal cliques does not even fit into the memory of current high-end computing systems. In an online discussion about parts of the open source implementation, the

authors admitted the absolute necessity of the pruning [Zha23]. While one cannot deny the efficacy of the pruning heuristics, this obliterates the purported major innovation of simply evaluating the maximal cliques.

4.1.2 Registration of Oriented Objects

When not only points, but oriented objects are to be registered, their orientation information can be exploited. This has been done for generic object registration, loop closures, and SLAM in general. It is important to distinguish the goal of global registration from local odometry approaches, which use line or edge features as well as planar patches [ZS14], as the latter assume a pose prior for association or even feature extraction.

Grimson [Gri90] has summarized fundamental ideas to recognize and register oriented objects in 3D space, including edges, cylinders, and planes, by constructing consistency constraints.

Standing upright on the ground, pole-like features can be reduced to a 2D problem. This holds in particular when lidar sensors with a known scale of the environment are used since it enables global pattern matching approaches [Bre09, SB14, CRG+20], sometimes also referred to as fingerprinting.

One of the first methods to use planes as features for 3D SLAM was proposed by Weingarten and Siegwart [WS05, WS06]. They registered planes based on individual compatibility using the Mahalanobis distance in the so-called symmetries and perturbations model [CT99].

Pathak et al. [PBV+10] were the first to compute more than nearest neighbor correspondences for planes. They exploited their size, orientation, and position as well as the corresponding uncertainties to perform a series of six different tests for both pairwise consistency and agreement with the ego motion. Later, Cupec et al. [CNF+15] extended this idea to 3D line segments.

Trevor et al. [TRC12] used 3D planes and 2D lines, associated using JCBB, in a pose graph SLAM. Taguchi et al. [TJR+13] presented a RANSAC approach that can degrade from planes to points on a primitive level. For a detailed

analysis of plane representations for SLAM and their respective numerical advantages during optimization the interested reader is referred to recent papers [GEY+18, ZWK21].

A very interesting perspective on the comparison of oriented objects was recently proposed by Lusk et al. [LH22, LPH23]. Viewing planes and lines as elements of the (affine) Grassmannian manifold enables the computation of mathematically sound transformation invariant distances between them, even if they have different dimensionality, like between planes and lines. Additionally, the proposed distances are invariant to measurement errors in the objects' positions if they are within the plane or along line. This offers robustness against partial occlusions. However, as will be shown in Section 4.3.3, knowing a coarse position can already help to yield better metrics. This makes it possible to challenge this theoretically very appealing perspective in practice.

Previous publications to which the author of this thesis contributed proposed to globally associate oriented objects of non-local extent, such as solid road marking lines or curbs, by sampling points along the line [MPH+22a, MPH+22b]. Pairwise compatibility measures can then incorporate different uncertainties along and across the line.

The approach proposed in this thesis is based on oriented objects with very local extent. As presented in Chapter 3 their position, orientation, and dimensions can be measured with very high accuracy. Hence, orientational and positional information need to be balanced to exploit them optimally. This motivates the proposal of probabilistically modeled joint Euclidean angular distance difference (JEADD).

4.1.3 Highly Accurate Localization

Finally, the state of the art of highly accurate methods for localizing in maps, especially HD or semantic HAD maps, is reviewed to gather a baseline for evaluating the approach proposed in this work. The review focuses first and foremost on localization accuracy and is based on two surveys that include

accuracy numbers [CPA23, SCC23][1]. As the entire landscape of localization approaches, spanned by sensor modalities, map formats, environments, and feature extraction approaches is vast and would burst the boundaries of this thesis, the interested reader is referred to those surveys for a broader overview.

The required localization accuracy can be illustrated with a prototypical function for urban automated driving, the association of mapped to detected traffic lights: To safely associate typical traffic lights with $0.2\,\mathrm{m}$ diameter in $50\,\mathrm{m}$ distance, the maximal positional error is $0.1\,\mathrm{m}$ and the maximal angular error is $0.115°$.

Many localization approaches that use sensor-specific descriptors, *e.g.* visual features or raw point clouds, have reported positional errors in the single digit centimeter range as well as angular errors below $0.1°$ [LS14, SLK+17, SDF+18, SS18, CER+21, CBT+23]. While such approaches are perfectly suited for reference localization, experience at the Institute of Measurement and Control Systems (MRT) has shown that such maps age quickly. Additionally, the highly dense features require three to six orders of magnitude more storage space compared to sparse semantic HAD maps, *i.e.* typically tens of megabytes to gigabytes instead of kilobytes per kilometer, even when using map compression techniques [DLB+15, MBB+16, LYL+23].

Recently, deeply learned localization approaches superseded manual feature extraction, matching, pose estimation, and even filtering [CWL+20, CLF+21]. Being similarly trained on one sensor or at least sensing modality, deep learning approaches can reach the accuracy of conventional sensor-specific methods. But, so far, they require maps with similar sizes, *e.g.* about $5\,\mathrm{MiB}$ [WGV+23] to $14\,\mathrm{MiB}$ [LZW+19] per kilometer.

Achieving similar localization accuracy in more lightweight HAD maps is more challenging. Combining corner, pole, and wall-like features with graph optimization, Cao et al. report a positional RMSE of around $0.1\,\mathrm{m}$ [CRG+20]. Using only pole-like landmarks with a particle filter, Schaefer et al. can reproduce similar errors as well as an angular RMSE of $0.2°$ [SBV+19]. Similarly, with

[1] Note that the author of this thesis found the tabularized references and categorization in [CPA23] partially incorrect.

poles and curbs as features and a particle filter, Cai et al. report $0.08\,\mathrm{m}$ positional and $0.08°$ angular RMSE [CLW+22].

Without the help of temporal filtering or bundle adjustment techniques, only significantly less accurate results have been reported. For feature point cloud registration on the KITTI benchmark [GLU12, GLS+13], thresholds of $0.6\,\mathrm{m}$ and $5°$ are commonly use to consider a registration as successful. With one of the most recent state-of-the-art methods, Lusk et al. report MAEs of $0.2\,\mathrm{m}$ and about $1°$, respectively [LPH23].

This raises the question if the accuracy of storage-intensive sensor-specific approaches can be achieved using easy-to-maintain, sensor-agnostic semantic HAD maps even *without* requiring any temporal filtering.

As presented in Section 4.5.3, with an average RMSE of $0.02\,\mathrm{m}$ and $0.02°$, respectively, in a sensor-agnostic map with a size of about $25\,\mathrm{KiB}$ per kilometer, the localization method proposed in thesis could be considered a possible answer.

4.2 Probabilistic Correspondence Graphs

The core innovation of this chapter is a framework that combines the expressiveness and mathematical soundness of probabilistic modeling with the completeness, exactness and, hence, optimality guarantees of viewing data association as a search problem. Previously, the combination of both required either a prior pose or the expensive computation of a transformation for each association hypothesis. Globally optimal methods that require neither are limited by strong assumptions on noise, infeasible solution times or restrictive approximations when modeling the compatibility problem as binary graph.

Underpinning the search with log likelihoods in a weighted graph, called probabilistic correspondence graph (PCG), enables guiding the search algorithm efficiently and retrieving probabilistically optimal solution in real time. For this, the probabilistic data association problem is reformulated in a transformation invariant probabilistic correspondence space.

4.2.1 Problem Definition and Goal

The formal problem modeled by a PCG is to associate data points from two sets, $\boldsymbol{X}$ and $\boldsymbol{Y}$ that are indexed by the index sets $\boldsymbol{I_X}$ and $\boldsymbol{I_Y}$, respectively. To deal with outliers, *i.e.* points without valid association, one can then split each $\boldsymbol{X}$ and $\boldsymbol{Y}$ disjointly into inlier sets $\boldsymbol{X}^+$, $\boldsymbol{Y}^+$ and outlier sets $\boldsymbol{X}^-$, $\boldsymbol{Y}^-$: $\boldsymbol{X} = \boldsymbol{X}^+ \dot\cup \boldsymbol{X}^-, \boldsymbol{Y} = \boldsymbol{Y}^+ \dot\cup \boldsymbol{Y}^-$. In order to incorporate outliers, the second index set is extended by the null measurement $\emptyset$: $\boldsymbol{I}_{\boldsymbol{Y}}^{\emptyset} = \boldsymbol{I_Y} \cup \{\emptyset\}$.

The positions of inliers can then be related by a transformation T and a measurement noise process $\boldsymbol{W}$. For this work, T is assumed to be an isometry, positions of data points $\boldsymbol{x}_i, \boldsymbol{y}_k$ to be Euclidean $\boldsymbol{c}_i, \boldsymbol{c}_k \in \mathbb{R}^n$, and $\boldsymbol{W}$ to be zero-mean additive Gaussian noise, *i.e.*

$$\forall \boldsymbol{x}_i \in \boldsymbol{X}^+ \exists! \boldsymbol{y}_k \in \boldsymbol{Y}^+ : \boldsymbol{c}_k = T\boldsymbol{c}_i + \boldsymbol{w},\ \boldsymbol{w} \sim \mathcal{N}(0, \boldsymbol{\Sigma_W}). \tag{4.2}$$

The corresponding inliers of both sets can be related via the bijective *true assignment* $\theta^\star : \boldsymbol{I}_{\boldsymbol{X}}^+ \to \boldsymbol{I}_{\boldsymbol{Y}}^+$. As $\theta^\star$ is not known, the formal goal is to retrieve one or multiple best assignments, θ^* or Θ^*, from the set of all valid assignments, Θ. Although this is no actual limitation of PCGs, for this work, one-to-one assignments are assumed, *i.e.*

$$\Theta := \left\{\theta : \boldsymbol{I_X} \to \boldsymbol{I}_{\boldsymbol{Y}}^{\emptyset} : \nexists i,j \in \boldsymbol{I_X}, i \neq j \colon \theta(i) = \theta(j) \neq \emptyset\right\}. \tag{4.3}$$

This implies that the inlier sets are assumed to have equal cardinality *i.e.* $\left|\boldsymbol{X}^+\right| = \left|\boldsymbol{Y}^+\right|$. The (hypothetical) combination of one data point of each set $(\boldsymbol{x}_i, \boldsymbol{y}_k)$ is called *correspondence* and often abbreviated by using only the indices $(i,k) \in \boldsymbol{I_X} \times \boldsymbol{I_Y} =: \boldsymbol{I_{XY}}$. A correspondence (i,k) is part of an assignment θ if and only if $\theta(i) = k$, which allows us to define the correspondence inlier and outlier sets

$$\boldsymbol{I}_\theta^+ := \{(i,k) \in \boldsymbol{I_{XY}} : \theta(i) = k\}, \tag{4.4}$$

$$\boldsymbol{I}_\theta^- := \{(i,k) \in \boldsymbol{I_{XY}} : \theta(i) = \emptyset, k \notin \theta(\boldsymbol{I_X})\}. \tag{4.5}$$

Similar to $\boldsymbol{I}_{\boldsymbol{X}}^{+}, \boldsymbol{I}_{\boldsymbol{Y}}^{+}$, for the true assignment $\theta^{\star}$, the true correspondence inlier and outlier sets, $\boldsymbol{I}_{\boldsymbol{XY}}^{+} := \boldsymbol{I}_{\theta^{\star}}^{+}, \boldsymbol{I}_{\boldsymbol{XY}}^{-} := \boldsymbol{I}_{\theta^{\star}}^{-}$, can be defined. It might be worth noting that $\boldsymbol{I}_{\theta}^{+}$ contains the inlier correspondences *claimed* by θ while $\boldsymbol{I}_{\boldsymbol{XY}}^{+}$ denotes the *true* inlier correspondences.

In order to determine the best assignment(s) θ^{*}, which ideally matches $\theta^{\star}$, a suitable model is required. This thesis proposes to determine the best assignment in a probabilistic maximum likelihood estimation (MLE) sense:

$$\theta^{*} := \underset{\theta \in \Theta}{\arg\max}\, \ell_{\theta}(\theta). \tag{4.6}$$

The underlying probabilistic model that defines the log likelihood $\ell_{\theta}(\theta)$ of an assignment θ will be introduced over the following sections. It will be based on the idea of finite set statistics (FISST) which defines probabilities by so-called *set integrals* [Mah07, p. 296 ff.].

4.2.2 Inlier/Outlier Process

As the goal is a probabilistically optimal assignment, a model for inliers and outliers to occur needs to be specified. In general, the PCG framework proposed in this chapter only requires that individual detections are statistically independent, but assumes no specific stochastic process.

Practically, coarsely following Reuter [Reu14, Chapter 6.3], two multi-Bernoulli (MB) processes [Mah07] are assumed. Hence, the inlier probability can be stated for each data point independently based on the detection probabilities $p_D(\cdot)$:

$$p(\boldsymbol{x} \in \boldsymbol{X}^{+}) = p_D(\boldsymbol{x}), \tag{4.7}$$
$$p(\boldsymbol{x} \in \boldsymbol{X}^{-}) = 1 - p_D(\boldsymbol{x}), \tag{4.8}$$
$$p(\boldsymbol{y} \in \boldsymbol{Y}^{+}) = p_D(\boldsymbol{y}), \tag{4.9}$$
$$p(\boldsymbol{y} \in \boldsymbol{Y}^{-}) = 1 - p_D(\boldsymbol{y}). \tag{4.10}$$

In contrast to the more common combination of an MB process for track detection and a Poisson process for clutter, the choice of two MB processes allows us to

handle both sets symmetrically, involves more intuitive parameters and takes into account that both $\boldsymbol{X}$ and $\boldsymbol{Y}$ are assumed finite with known cardinality.

However, since for valid assignments each inlier in $\boldsymbol{X}$ needs to be matched by an inlier in $\boldsymbol{Y}$, a set integral over the joint space $\boldsymbol{X} \uplus \boldsymbol{Y}$ is not a valid probability distribution anymore. This concern was already raised by Reuter [Reu14, Chapter 6.3] and results from violating the statistical independence assumptions for the convolution of individually valid set integrals [Mah07, p. 385 ff.] over $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively.

While lacking a rigorous proof, after careful examination, the author of this thesis conjectures that the statistical independence assumption holds for all "allowed" combinations, *i.e.* for the set of correspondences. This yields a density which is valid up to a normalization factor. This normalization factor compensates the probability mass that would be allocated to "forbidden" combinations and duplicates that yield identical solutions. Moreover, when considering only relative likelihoods, as done below, the normalization factor cancels out, rendering this formally only approximate coupling of two MB processes exact again.

As relative likelihoods will be sufficient for all applications in this thesis, two coupled MB processes are assumed. For other use cases, correct probabilities can be obtained by replacing one of the MB processes with a Poisson process. This is the standard formulation used by *e.g.* Mahler [Mah07] who proves its correctness comprehensively.

4.2.3 Probabilistic Correspondence Space

The main factor that determines the probability of an assignment is not the inlier/outlier probability, but how well the point sets match, *i.e.* how compatible they are. For just two points $\boldsymbol{x}$,$\boldsymbol{y}$ in Euclidean space that form a correspondence, one cannot reason about their compatibility as they bear no additional information unless a transformation T is assumed.

Traditionally, *e.g.* in multi-target tracking [RVV+14, Mah07], but also in previous filter-based RFS SLAM methods [MVA+11, DRD15, FGS+17], the probability

of each measurement-to-track assignment can be evaluated individually given a prior state predicted from a previous time step. Since this work deliberately does not assume any prior pose, the only alternative feasible in real time is constructive sampling of such a pose, as *e.g.* RANSAC [FB81] does. Another alternative is searching the pose space, which is prohibitively expensive [Gri90, YLJ13, YLC+16].

As solution that does not require any knowledge about the transformation T, transformation invariant measurements (TIMs) have been proposed in a number of variations [Bol79, BC82, Gri90, BNR+00, Bai02, NTC03, MDE+18, YC19, YSC21]. They allow to evaluate the compatibility of correspondence *pairs*, *i.e.* two hypothetical assignments, by measuring transformation invariant information *within* each set $\boldsymbol{X}$ or $\boldsymbol{Y}$, *e.g.* pairwise distances. Since TIMs work on pairs of correspondences and typically measure geometric information, this is also referred to as (geometric) pairwise compatibility or pairwise consistency (PC).

Unfortunately, TIMs do not allow to re-use commonly employed methods to compute the best assignment(s) [Mun57, Mur68, Yen71, JV87, Epp98]. However, they can efficiently guide the search for the optimal assignment in a probabilistic correspondence graph. Before introducing the necessary probability space on TIMs, first, the more commonly used concept of probabilities in residual space is stated.

Likelihoods in Residual Space

When a transformation T is known as prior or computed from a hypothetical assignment, the probability of assumed inliers $(i,k) : \theta(i) = k$ can be judged in terms of the errors *after* alignment via T. Since these errors are residual after transformation, this space will be called *residual space* in this thesis. As it is the very same space that the zero-mean Gaussian noise process $\boldsymbol{W}$ is living in, the evaluation is trivial:

$$f_{\mathrm{RS}}(i,k) \propto f_{\mathcal{N}}(\boldsymbol{c}_k - T\boldsymbol{c}_i \mid 0, \boldsymbol{\Sigma}_{\boldsymbol{W}}). \tag{4.11}$$

However, it either requires knowledge about a prior pose T or comes at the hefty cost of computing it for a vast number of combinations of correspondences.

Heuristics in Correspondence Space

A more efficient alternative is to search for assignments in correspondence space, *i.e.* the space of pairwise compatible associations. Using heuristic measures to guide the search for the optimal assignment in correspondence space is no novel idea. Both continuous subgraph densities [LH05, LFH21] and binary pruning [NTC03] have been proposed previously.

The latter is far more common and conceptually close to the method proposed in this thesis. Correspondences $v_{ik} = (\boldsymbol{x}_i, \boldsymbol{y}_k)$ form vertices in a binary correspondence graph $\mathcal{G}_{\text{bin}} = (\mathcal{V}_{\text{bin}}, \mathcal{E}_{\text{bin}})$. Based on the idea of pairwise compatibility (PC), if two correspondences v_{ik}, v_{jl} are deemed compatible, *e.g.* because their respective distances $\|\boldsymbol{c}_i - \boldsymbol{c}_j\|$,$\|\boldsymbol{c}_k - \boldsymbol{c}_l\|$ are similar up to a certain threshold τ_{bin}, they are connected:

$$(v_{ik}, v_{jl}) \in \mathcal{E}_{\text{bin}} \iff \big|\|\boldsymbol{c}_i - \boldsymbol{c}_j\| - \|\boldsymbol{c}_k - \boldsymbol{c}_l\|\big| < \tau_{\text{bin}}. \tag{4.12}$$

An exemplary binary correspondence graph is depicted in Figure 4.2.

The quality of an assignment is then judged by the cardinality of a fully [NTC03] or mostly [SYC21] connected subgraph. In the example, the connectivity means that the distances between all or most correspondence pairs are similar up to the threshold τ_{bin}. In the limit $\tau_{\text{bin}} \to 0$, it follows that fully connected subgraphs imply that the constellations in $\boldsymbol{X}$ and $\boldsymbol{Y}$ are congruent [Blu53, EGW+04].

The idea of pairwise compatible correspondences builds upon the assumption that the likelihood of a compatible outlier is much lower than the differences in *e.g.* distances being explained by measurement noise. Hence, while few correspondences in the largest PC subgraph might be outliers, the likelihood that only outliers form the largest PC subgraph shrinks exponentially with its size.

The challenge with this idea is to correctly choose the threshold τ_{bin}, which is crucial for inlier retrieval as it trades precision against recall. Still, as this method

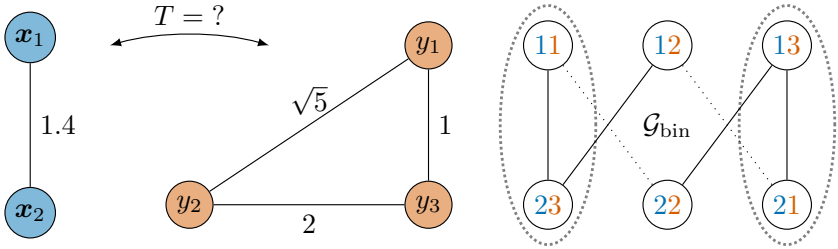


Figure 4.2: On the left, an example problem is depicted. The goal is to associate the blue points $\boldsymbol{X}$ to the orange points $\boldsymbol{Y}$ without knowing the transformation T between them. Edges between points are labeled with pairwise distances.
It can be solved via constructing a *binary* correspondence graph $\mathcal{G}_{\text{bin}}$ as depicted on the right. Each vertex represents a correspondence of one point in each $\boldsymbol{X}$ and $\boldsymbol{Y}$. Two vertices are connected by an edge if and only if the difference of pairwise distances within each set is smaller than a maximum threshold τ_{bin} (*cf.* Equation (4.12)).
Exemplary, for $\tau_{\text{bin}} = 0.6$, four edges (solid) are formed in the correspondence graph on the right. Edges that are excluded by the threshold are depicted dotted. Connections that are missing entirely are "illegal" due to the one-to-one assignment assumption.
This exemplary threshold for a binary correspondence graph yields four seemingly equally optimal assignments, each along one solid edge with a cardinality of 2. Although comparing the pairwise distances suggest to prefer the encircled two assignments over the others, this information is lost during binary thresholding.
As innovation over such binary correspondence graphs, the idea of the method proposed in this chapter is to retain this information and exploit it using a probabilistically optimal formulation.

ranks among the state of the art for large-scale correspondence problems, the binarized cardinality approximation obviously works well.

However, regarding optimality, especially for smaller problems such as when localizing in a sparse map, it raises the question which effect the binarization has on the connectivity and, hence, the cardinality approximation of the true likelihood. Approaching this issue with findings from the field of random graphs seems very elegant and promising. Hence, it was tried by the author, but unfortunately yielded no usable guarantees. A number of potentially interesting facts to continue exploring this approach is gathered in Appendix D.

Likelihoods in Correspondence Space

Instead of modeling the effects of binarization, this thesis proposes to consider continuous probabilities of inliers and outliers directly in correspondence space. Thresholds, comparable to gating in residual space, can still be introduced to trade optimality for speed, but can be arbitrarily conservative in general without losing information as is it the case with existing binary correspondence graphs. As shown in the following sections, continuous probabilities allow to unify the advantages of a fully probabilistic modeling with the ability to directly retrieve the assignment with optimal likelihood without any prior pose.

Likelihoods in correspondence space are based on transformation invariant measurements (TIMs) Δ_{ijkl} which allow to compare two correspondences $(\boldsymbol{x}_i,\boldsymbol{y}_k)$, $(\boldsymbol{x}_j,\boldsymbol{y}_l)$ using transformation invariant information from the respective point pairs $(\boldsymbol{x}_i,\boldsymbol{x}_j)$, $(\boldsymbol{y}_k,\boldsymbol{y}_l)$. As explained in more detail in Section 4.3, typical examples are differences in distances or relative angles. Exemplary, in the noiseless case, for true inlier correspondences $(\boldsymbol{x}_i,\boldsymbol{y}_k)$, $(\boldsymbol{x}_j,\boldsymbol{y}_l)$, isometry T, and Euclidean distance δ, the following holds trivially and by the very meaning of *isometry* being derived from Greek *isos* (equal) and *metron* (measure)

$$\|\boldsymbol{c}_i - \boldsymbol{c}_j\|_2 =: \delta(\boldsymbol{x}_i, \boldsymbol{x}_j) = \delta(T\boldsymbol{x}_i, T\boldsymbol{x}_j) = \delta(\boldsymbol{y}_k, \boldsymbol{y}_l). \tag{4.13}$$

As the individual distances or relative angles are measured separately in each space of $\boldsymbol{X}$ or $\boldsymbol{Y}$, they are transformation invariant and so is their absolute difference

$$\Delta\delta(i,j,k,l) := |\delta(\boldsymbol{x}_i, \boldsymbol{x}_j) - \delta(\boldsymbol{y}_k, \boldsymbol{y}_l)| \tag{4.14}$$

which is referred to as Euclidean distance difference (EDD).

PCGs as data association concept are independent from the underlying TIMs Δ_{ijkl}. This thesis only assumes their symmetry $\Delta_{ijkl} = \Delta_{jikl} = \Delta_{ijlk}$, their transformation invariance and the ability to compile a – possibly merely empirical – statistic over the TIMs of all inliers, outliers, and their combination.

While likelihoods in residual space involve only one correspondence, likelihoods in correspondence space involve a *pair* of correspondences. Given their TIM Δ_{ijkl} one can compute likelihoods for various conditions, *e.g.* if one or both correspondences are inliers or outliers, which have the notation

$$f_\Delta(i,j,k,l) := f_\Delta(\Delta_{ijkl}). \tag{4.15}$$

Likelihood formulations involving correspondence triplets [YHQ+22] or structures of higher order [CSY+22] have been conceived for the retrieval of optimal assignments in binarized correspondence space. As there is no obvious advantage for a *continuous* probabilistic correspondence space, their exploration is left an open issue.

4.2.4 Optimal Assignments in Correspondence Space

Given the definition of TIMs Δ, the (unnormalized!) likelihood of an assignment θ can now be described as follows, coarsely following Mahler [Mah07, Eq. 12.139, Mah14, Eq. 7.32] and Reuter [Reu14, Eq. 6.12]:

$$f_\theta(\theta) = \prod_{i \in \boldsymbol{I_X}} (1 - p_D(\boldsymbol{x}_i)) \prod_{k \in \boldsymbol{I_Y}} (1 - p_D(\boldsymbol{y}_k)) \prod_{(i,k) \in \boldsymbol{I_{XY}}} f_{IC}^-(i,k) \tag{4.16a}$$

$$\prod_{(i,k) \in \boldsymbol{I_{XY}}} \prod_{\substack{(j,l) \in \boldsymbol{I_{XY}}: \\ j>i, k \neq l}} f_\Delta^-(i,j,k,l) \tag{4.16b}$$

$$\prod_{(i,k) \in \boldsymbol{I}_\theta^+} \frac{p_D(\boldsymbol{x}_i)}{1 - p_D(\boldsymbol{x}_i)} \frac{p_D(\boldsymbol{y}_k)}{1 - p_D(\boldsymbol{y}_k)} \frac{f_{IC}^+(i,k)}{f_{IC}^-(i,k)} \tag{4.16c}$$

$$\prod_{(i,k) \in \boldsymbol{I}_\theta^+} \prod_{(j,l) \in \boldsymbol{I}_\theta^+ : j>i} \frac{f_\Delta^+(i,j,k,l)}{f_\Delta^-(i,j,k,l)} \tag{4.16d}$$

$$\prod_{(i,k) \in \boldsymbol{I}_\theta^+} \prod_{\substack{(j,l) \in \boldsymbol{I}_\theta^- : \\ i \neq j, k \neq l}} \frac{f_\Delta^\pm(i,j,k,l)}{f_\Delta^-(i,j,k,l)}. \tag{4.16e}$$

Next to the previously introduced detection probability p_D, the equation involves two kinds of likelihood terms, which will be defined below. For now, it is sufficient to know that $f_{IC}^{+}(i,k), f_{IC}^{-}(i,k)$ captures individual compatibility of correspondences while $f_{\Delta}^{+}(i,j,k,l), f_{\Delta}^{-}(i,j,k,l), f_{\Delta}^{\pm}(i,j,k,l)$ evaluates the compatibility of correspondence pairs based on the TIM Δ. While being rather unhandy, Equation (4.16) can be broken down into its five parts.

Inspired by Mahler [Mah07, Eq. 12.139], Expressions (4.16a) and (4.16b) model the likelihood that all data points are outliers, *i.e.* nothing is detected. As in [Mah07, Eq. 12.140], they are countered by Expressions (4.16c) to (4.16e).

Expression (4.16c) compensates the respective terms in Expression (4.16a) for inliers of θ. In addition, it judges how likely the detection of both points and how compatible each inlier correspondence (i,k) is individually. Similarly, Expression (4.16d) compensates the terms of Expression (4.16b) for pairs of inliers and evaluates their pairwise compatibility.

Finally, Expression (4.16e) compensates and evaluates pairwise compatibility for pairs of inliers and outliers. Note that due to 1 as neutral element of the product operator, their (non-)detection and individual (in-)compatibility is already captured in the previous expressions. Hence, they can be omitted in Expression (4.16e).

The $j > i$ restriction of the inner product terms ensures that each combination is only captured once. A similar $l > k$ restriction would lead to skipping terms. Requiring that $k \neq l$ and $i \neq j$ ensures that only "legal" combinations of correspondences are covered.

It is to note that $f_\theta(\theta)$ is in fact a conditional density $f_{\theta|\boldsymbol{X},\boldsymbol{Y}}(\theta \mid \boldsymbol{X}, \boldsymbol{Y})$. The prior terms $f_{\boldsymbol{X}}(\boldsymbol{X}), f_{\boldsymbol{Y}}(\boldsymbol{Y}), p_\theta$ are assumed independent, p_θ is assumed uniform. While required for the correctness of the set integral, the terms are constant prefactors and hence omitted in favor of clearer notation.

Individual Compatibility

The terms $f_{IC}^{+}(i,k)$ and $f_{IC}^{-}(i,k)$ capture the *individual compatibility (IC)* of a correspondence (i,k). While meaningless for points in a purely Euclidean space, in real applications, they can be used to describe the similarity of computed or learned feature descriptors, semantic classes, or parametric landmarks.

One example are feature vectors $\boldsymbol{\xi}_i, \boldsymbol{\xi}_k$ associated to the points $\boldsymbol{x}_i, \boldsymbol{y}_k$. If the deviations in feature space were normally distributed with different scales $\boldsymbol{\Sigma}^{+}$ and $\boldsymbol{\Sigma}^{-}$ for inliers and outliers, respectively, the according IC terms would read

$$f_{IC}^{+}(i,k) := f_{\mathcal{N}}(\boldsymbol{\xi}_i \mid \boldsymbol{\xi}_k, \boldsymbol{\Sigma}^{+}) \tag{4.17}$$

$$f_{IC}^{-}(i,k) := f_{\mathcal{N}}(\boldsymbol{\xi}_i \mid \boldsymbol{\xi}_k, \boldsymbol{\Sigma}^{-}). \tag{4.18}$$

Pairwise Compatibility

While individual compatibility can exclude grave outliers, current feature descriptors or parameter vectors alone are not sufficient to yield spatially consistent assignments. To retrieve spatially consistent assignments, the *pairwise compatibility (PC)* terms $f_{\Delta}^{+}(i,j,k,l)$, $f_{\Delta}^{\pm}(i,j,k,l)$, and $f_{\Delta}^{-}(i,j,k,l)$ are necessary. In contrast to individual compatibility, pairs of correspondences are not either inliers or outliers, but can consist of one of each. This mixed case will be referred to as *crosslier*, denoted by $\pm$.

Proper modeling of the pairwise likelihoods of two correspondences is part of the major innovation of PCGs. They allow to guide the search for the optimal assignment(s) without losing the probabilistic information by applying thresholds. As shown later, they are the foundation to retrieve probabilistically optimal assignments efficiently.

As stated above, it is assumed that the distribution of TIMs Δ can be described statistically. Doing so separately for inliers, crossliers and outliers allows us to describe the distribution as mixture over the three groups

$$\begin{aligned} &f_\Delta(i,j,k,l) := f_\Delta(\Delta_{ijkl}) \\ &= f^+_\Delta(\Delta_{ijkl} \mid (i,k),(j,l) \in \boldsymbol{I}^+_{\boldsymbol{XY}})\, p((i,k),(j,l) \in \boldsymbol{I}^+_{\boldsymbol{XY}}) + \\ &\quad f^\pm_\Delta(\Delta_{ijkl} \mid (i,k) \in \boldsymbol{I}^+_{\boldsymbol{XY}}, (j,l) \in \boldsymbol{I}^-_{\boldsymbol{XY}}) \cdot \\ &\qquad\qquad p((i,k) \in \boldsymbol{I}^+_{\boldsymbol{XY}}, (j,l) \in \boldsymbol{I}^-_{\boldsymbol{XY}}) + \\ &\quad f^-_\Delta(\Delta_{ijkl} \mid (i,k),(j,l) \in \boldsymbol{I}^-_{\boldsymbol{XY}})\, p((i,k),(j,l) \in \boldsymbol{I}^-_{\boldsymbol{XY}}). \end{aligned} \tag{4.19}$$

The prior terms for each group can be split up under the assumption that correspondences are independent

$$p((i,k),(j,l) \in \boldsymbol{I}^+_{\boldsymbol{XY}}) = p((i,k) \in \boldsymbol{I}^+_{\boldsymbol{XY}}) p((j,l) \in \boldsymbol{I}^+_{\boldsymbol{XY}}), \tag{4.20}$$

$$p((i,k) \in \boldsymbol{I}^+_{\boldsymbol{XY}}, (j,l) \in \boldsymbol{I}^-_{\boldsymbol{XY}}) = p((i,k) \in \boldsymbol{I}^+_{\boldsymbol{XY}}) p((j,l) \in \boldsymbol{I}^-_{\boldsymbol{XY}}), \tag{4.21}$$

$$p((i,k),(j,l) \in \boldsymbol{I}^-_{\boldsymbol{XY}}) = p((i,k) \in \boldsymbol{I}^-_{\boldsymbol{XY}}) p((j,l) \in \boldsymbol{I}^-_{\boldsymbol{XY}}). \tag{4.22}$$

For both inlier and outlier cases, these terms can then be resolved to the (non)detection probabilities of both data points as well as an optional individual compatibility likelihood

$$p((i,k) \in \boldsymbol{I}^+_{\boldsymbol{XY}}) = p_D(\boldsymbol{x}_i) p_D(\boldsymbol{y}_k) f^+_{IC}(i,k), \tag{4.23}$$

$$p((j,l) \in \boldsymbol{I}^-_{\boldsymbol{XY}}) = (1 - p_D(\boldsymbol{x}_j))(1 - p_D(\boldsymbol{y}_l)) f^-_{IC}(j,l). \tag{4.24}$$

As the attentive reader might have noticed, these terms are already captured by the (non)detection probabilities and individual (in)compatibility likelihoods stated in Expressions (4.16a) and (4.16c). Hence, for each correspondence pair

$(\boldsymbol{x}_i, \boldsymbol{y}_k), (\boldsymbol{x}_j, \boldsymbol{y}_l)$ with the TIM Δ_{ijkl}, this leaves the terms

$$f_{\Delta}^{+}(i,j,k,l) := f_{\Delta}(\Delta_{ijkl} \mid (i,k),(j,l) \in \boldsymbol{I}_{\boldsymbol{XY}}^{+}), \tag{4.25}$$

$$f_{\Delta}^{\pm}(i,j,k,l) := f_{\Delta}(\Delta_{ijkl} \mid (i,k) \in \boldsymbol{I}_{\boldsymbol{XY}}^{+}, (j,l) \in \boldsymbol{I}_{\boldsymbol{XY}}^{-}), \tag{4.26}$$

$$f_{\Delta}^{-}(i,j,k,l) := f_{\Delta}(\Delta_{ijkl} \mid (i,k),(j,l) \in \boldsymbol{I}_{\boldsymbol{XY}}^{-}) \tag{4.27}$$

for inlier, crosslier, and outlier pairs, respectively.

Together with the detection probabilities, individual and pairwise compatibility likelihoods can fully describe the likelihood of an assignment *without* knowing, guessing or evaluating a transformation T between putative inliers. Hence, they are the foundation to guide the search for optimal assignments in correspondence space.

Before getting to the retrieval, to better understand the proposed likelihoods in correspondence space, in the next section they will be compared with conventional likelihoods after aligning the point sets $\boldsymbol{X}$ and $\boldsymbol{Y}$.

4.2.5 On Optimality in Correspondence Space

Likelihood formulations in correspondence space assume that the underlying transformation T is unknown, rendering the TIM-based likelihood terms independent. While being formally valid likelihoods, this neglects the fact that the TIMs may not be truly independent.

More precisely, for non-degenerate constellations of points in $\mathbb{R}^3$, knowing distances to three points is sufficient to make distances to all other points linearly dependent [EGW+04]. At the same time, not the actual distances, but only their changes due to positional noise are measured by TIMs. This yields a complex relationship between TIMs that the author of this thesis failed to capture elegantly. Although it seems certain that a true independence must not be assumed, correspondence space likelihoods promise the retrieve supposedly optimal solutions in previously impossible solution times.

In contrast, using likelihoods in residual space, *i.e.* after applying a transformation, comes at the cost of requiring a putative transformation, but nicely

separates the individual error terms for each correspondence. This motivates the question how the easily computable correspondence space likelihoods and the residual space likelihoods, which are "concealed" by a transformation, are related and if necessary or sufficient conditions exist for both to be (approximately) equal. While in this section the likelihoods are only compared, Sections 4.2.8 and 4.4.2 present methods to actually align correspondence space likelihoods to those in residual space.

To evaluate the residual space (RS) likelihood of an assignment θ given a transformation T, the following formulation is assumed

$$f_{\mathrm{RS}}(\theta \mid T) = \prod_{i \in \boldsymbol{I}_{\boldsymbol{X}}} (1 - p_D(\boldsymbol{x}_i)) \prod_{k \in \boldsymbol{I}_{\boldsymbol{Y}}} (1 - p_D(\boldsymbol{y}_k)) \tag{4.28a}$$

$$\prod_{(i,k) \in \boldsymbol{I}_\theta^+} \frac{p_D(\boldsymbol{x}_i)}{1 - p_D(\boldsymbol{x}_i)} \frac{p_D(\boldsymbol{y}_k)}{1 - p_D(\boldsymbol{y}_k)} f_{\mathcal{N}}(\boldsymbol{c}_k - T\boldsymbol{c}_i \mid 0, \boldsymbol{\Sigma}_{\boldsymbol{W}}). \tag{4.28b}$$

In addition to *correspondence space (CS) likelihoods* defined in Equation (4.16), this enables the definition of residual space likelihoods using the transformation estimated via Umeyama alignment [Ume91], simply referred to as *RS likelihood*. Thirdly, as reference, the residual space likelihood given the original transformation can be computed, being referred to as *true likelihood*. All three likelihoods can then be compared qualitatively, quantitatively, and formally.

First, one can examine them qualitatively by measuring real-world landmark distributions or by simulating random point sets. For instance, as depicted in Figure 4.3, 12 point pairs can be sampled as correspondences from the Stanford bunny model [TL94] as described in Section 4.5.1. 6 correspondences are inliers according to Equation (4.2) while the other 6 pairs are unrelated.

Each color encodes one out of five example data association problems. Each point shows the log likelihood of a possible assignment with three to six out of six true inliers. Cluster emerge from assignment groups with equal inlier cardinality. The plots show that while RS and CS have totally different scales, they order and separate different association problems (different colors) as well as assignments within each association problem very similarly.

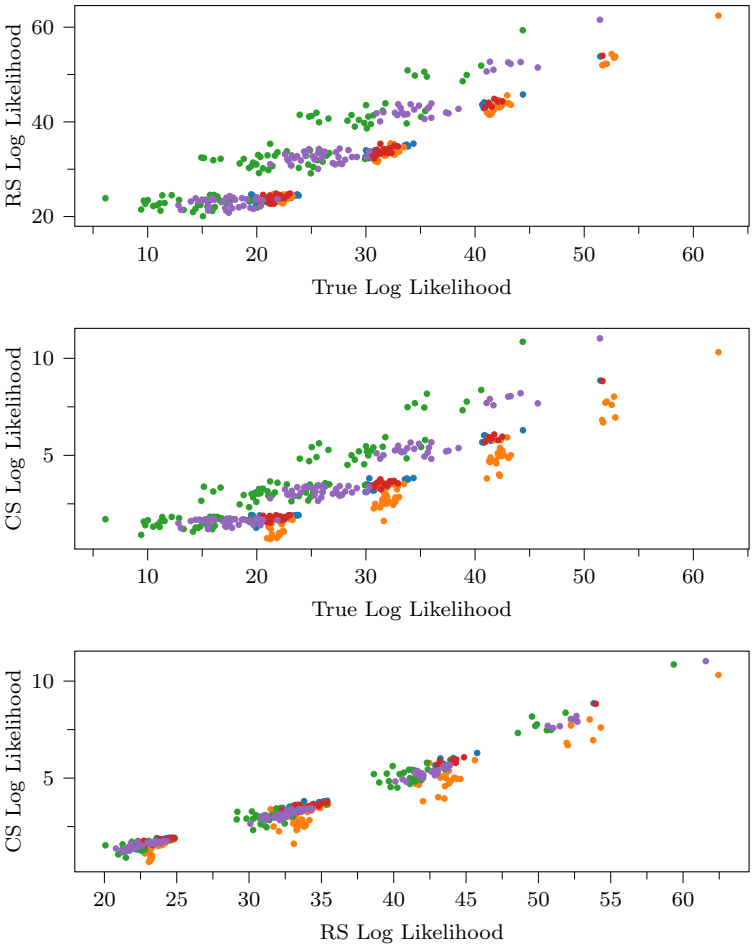


Figure 4.3: Comparison of residual and correspondence space likelihoods for five prototypical association problems, each encoded by its own color. The axes depicted residual space likelihood evaluated using the true transformation (True Log Likelihood) and using an estimated transformation (RS Log Likelihood) as well as the correspondence space likelihoods of each assignment (CS Log Likelihood).
The visual groups of points emerge from assignments with equal inlier cardinality. While different in absolute values, one can see that the proposed CS likelihoods are well-aligned with the commonly used RS likelihoods and separate assignments very similarly.

Using 100 instead of only five random problems shows that the probability spaces are highly correlated. Concretely, two correlation coefficients can be computed an averaged over all samples. Table 4.2 lists the Pearson correlation coefficients which measure a linear dependency. In addition, the Spearman correlation, which measures the rank order of results, is reported in Table 4.3.

Table 4.2: Average Pearson correlation coefficients between CS, RS and true log likelihoods.

	RS	**True**
CS	0.96 ± 0.13	0.96 ± 0.10
RS	-	0.99 ± 0.03

Table 4.3: Average Spearman correlation coefficients between CS, RS and true log likelihoods.

	RS	**True**
CS	0.96 ± 0.09	0.91 ± 0.08
RS	-	0.92 ± 0.04

The RS log likelihoods between estimated and true transformation definitely show a higher linear correlation compared to CS. In contrast, the rank order is almost identical and highest between RS and CS. Yet, in both metrics CS and RS log likelihood are highly correlated by absolute measures, confirming the suitability of CS log likelihoods to replace RS formulations without the need for an estimated transformation.

Besides empirical examinations, geometrical reasoning using the concepts of distance geometry and graph rigidity [Blu53, EGW+04] allows us to establish geometric conditions for equality of optimality in both likelihood formulations when using Euclidean distance differences (EDDs) in correspondence space. The interaction of individual terms, *e.g.* detection probabilities, and spatial likelihoods, as measured in correspondence or residual space, is highly non-trivial. Thus, assignments with equal inlier/outlier cardinality are assumed for the following thoughts.

Findings from network localization show that pairwise distances only lead to a correct (graph) registration if the point sets are not degenerate, *e.g.* coplanar or collinear. In addition, the inlier set is required to be large enough, *i.e.* it needs to contain three correspondences in $\mathbb{R}^2$ and four in $\mathbb{R}^3$. The uniqueness of a registration is then given due to so-called global rigidity of a complete and large enough graph that is not degenerated in position [EGW+04]. Hence, both general position and count of inliers are necessary conditions for correspondence space optimality to validly approximate residual space optimality.

On the other hand, in the asymptotically noiseless case, *i.e.* when there is no difference between any corresponding Euclidean distances in the inlier set, the optimality of both terms is trivial by the definition of congruence [Blu53]. Unfortunately, it is less clear how non-optimal likelihoods behave.

A similar train of thought can be used for joint compatibility (JC) and pairwise compatibility (PC). While joint compatibility (JC) incorporates an initial pose in an EKF manner, there are similarities to the probabilistic residual space likelihood formulation. Hence, it is conjectured that the same geometric conditions that lead to equality of likelihoods in correspondence and residual space imply the equality of pairwise compatibility and joint compatibility.

4.2.6 Constructing a PCG

The probabilistic modeling in correspondence space is the foundation for constructing a probabilistic correspondence graph (PCG). A PCG is a totally weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Vertices $v_{ik} \in \mathcal{V}$ are possible correspondences $(i,k) \in \boldsymbol{I_{XY}}$. Edges $(v_{ik},v_{jl}) \in \mathcal{E}$ connect two vertices $v_{ik} = (i,k)$ and $v_{jl} = (j,l)$ if and only if the coexistence of v_{ik} and v_{jl} does not violate the logic of one-to-one assignments or possibly other user-defined rules.

Both vertices and edges are assumed to be weighted by real-valued weighting functions $\mathtt{w}(v_{ik})\colon \mathcal{V} \to \mathbb{R}$, $\mathtt{w}((v_{ik},v_{jl}))\colon \mathcal{E} \to \mathbb{R}$ that will be defined below. The weights will be used in the next section to introduce the (non-)detection probabilities as well as individual and pairwise compatibility likelihoods. An example is depicted in Figure 4.4.

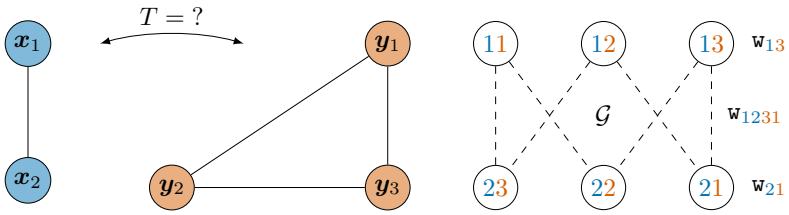


Figure 4.4: Concept of a probabilistic correspondence graph (PCG). All possible correspondences form vertices and are connected unless they violate the assumption of a one-to-one assignment. More restricted correspondence sets, *e.g.* based on feature matching, are possible and can vastly improve solution times.

4.2.7 Optimal Association as Maximum Weight Cliques

In order to model the data association problem and retrieve probabilistically optimal assignment(s) exactly, one needs to bridge the gap between the PCG as weighted correspondence graph and the respective probability and likelihood terms introduced in Equation (4.16).

As major innovation of this thesis, it combines the idea of retrieving feasible solutions in correspondence space, that was considered only binarily in previous works, with the probabilistic modeling used for finite set statistics (FISST) object tracking. The goal is to reformulate the likelihood of an assignment, given as product in Equation (4.16), such that it can be viewed as sum over vertices and edges of a PCG.

The first step is to reformulate the product formulation from Equation (4.16) as a sum over log likelihoods ℓ. Each likelihood term $\ell_\theta, \ell_{IC}, \ell_\Delta$ corresponds to the respectively denoted likelihood $f_\theta, f_{IC}, f_\Delta$.

$$\ell_\theta(\theta) = \sum_{i \in \boldsymbol{I}_{\boldsymbol{X}}} \log(1 - p_D(\boldsymbol{x}_i)) + \sum_{k \in \boldsymbol{I}_{\boldsymbol{Y}}} \log(1 - p_D(\boldsymbol{y}_k)) + \sum_{(i,k) \in \boldsymbol{I}_{\boldsymbol{XY}}} \ell_{IC}^-(i,k) \tag{4.29a}$$

$$+ \sum_{(i,k) \in \boldsymbol{I}_{\boldsymbol{XY}}} \sum_{\substack{(j,l) \in \boldsymbol{I}_{\boldsymbol{XY}}: \\ j > i, k \neq l}} \ell_\Delta^-(i,j,k,l) \tag{4.29b}$$

$$+ \sum_{(i,k) \in \boldsymbol{I}_\theta^+} \log\left(\frac{p_D(\boldsymbol{x}_i)}{1 - p_D(\boldsymbol{x}_i)} \frac{p_D(\boldsymbol{y}_k)}{1 - p_D(\boldsymbol{y}_k)} \right) + \ell_{IC}^+(i,k) - \ell_{IC}^-(i,k) \tag{4.29c}$$

$$+ \sum_{(i,k) \in \boldsymbol{I}_\theta^+} \sum_{(j,l) \in \boldsymbol{I}_\theta^+ : j > i} \left(\ell_\Delta^+(i,j,k,l) - \ell_\Delta^-(i,j,k,l) \right) \tag{4.29d}$$

$$+ \sum_{(i,k) \in \boldsymbol{I}_\theta^+} \sum_{\substack{(j,l) \in \boldsymbol{I}_\theta^-: \\ i \neq j, k \neq l}} \left(\ell_\Delta^\pm(i,j,k,l) - \ell_\Delta^-(i,j,k,l) \right). \tag{4.29e}$$

Sums and double sums over correspondences can be rewritten as sums over vertices and sums over edges between them, respectively. For this, the correspondence vertices $\mathcal{V}$ can be divided disjointly into vertices selected as inliers by an assignment θ, $\mathcal{V}_\theta^+ := \boldsymbol{I}_\theta^+$, and those not selected, $\mathcal{V}_\theta^- := \boldsymbol{I}_\theta^-$.

$$\ell_\theta(\theta) = \sum_{i \in \boldsymbol{I}_{\boldsymbol{X}}} \log(1 - p_D(\boldsymbol{x}_i)) + \sum_{k \in \boldsymbol{I}_{\boldsymbol{Y}}} \log(1 - p_D(\boldsymbol{y}_k)) + \sum_{(i,k) \in \boldsymbol{I}_{\boldsymbol{XY}}} \ell_{IC}^-(i,k) \tag{4.30a}$$

$$+ \sum_{(i,k) \in \boldsymbol{I}_{\boldsymbol{XY}}} \sum_{\substack{(j,l) \in \boldsymbol{I}_{\boldsymbol{XY}}: \\ j > i, k \neq l}} \ell_\Delta^-(i,j,k,l) \tag{4.30b}$$

$$+ \sum_{(i,k)\in\mathcal{V}_\theta^+} \log\left(\frac{p_D(\boldsymbol{x}_i)}{1-p_D(\boldsymbol{x}_i)}\frac{p_D(\boldsymbol{y}_k)}{1-p_D(\boldsymbol{y}_k)}\right) + \ell_{IC}^+(i,k) - \ell_{IC}^-(i,k) \tag{4.30c}$$

$$+ \sum_{(i,k)\in\mathcal{V}_\theta^+} \sum_{(j,l)\in\mathcal{V}_\theta^+ : j>i} \left(\ell_\Delta^+(i,j,k,l) - \ell_\Delta^-(i,j,k,l)\right) \tag{4.30d}$$

$$+ \sum_{(i,k)\in\mathcal{V}_\theta^+} \sum_{\substack{(j,l)\in\mathcal{V}_\theta^- : \\ i\neq j, k\neq l}} \left(\ell_\Delta^\pm(i,j,k,l) - \ell_\Delta^-(i,j,k,l)\right). \tag{4.30e}$$

Approximating $\ell_\Delta^\pm(i,j,k,l) \approx \ell_\Delta^-(i,j,k,l)$ allows us to write this as a part that is independent from the assignment θ, Expressions (4.31a) and (4.31b), and a part that only depends on inlier vertices and edges between them, Expressions (4.31c) and (4.31d):

$$\ell_\theta(\theta) = \sum_{i\in\boldsymbol{I}_\boldsymbol{X}} \log(1-p_D(\boldsymbol{x}_i)) + \sum_{k\in\boldsymbol{I}_\boldsymbol{Y}} \log(1-p_D(\boldsymbol{y}_k)) + \sum_{(i,k)\in\boldsymbol{I}_{\boldsymbol{XY}}} \ell_{IC}^-(i,k) \tag{4.31a}$$

$$+ \sum_{(i,k)\in\boldsymbol{I}_{\boldsymbol{XY}}} \sum_{\substack{(j,l)\in\boldsymbol{I}_{\boldsymbol{XY}} : \\ j>i, k\neq l}} \ell_\Delta^-(i,j,k,l) \tag{4.31b}$$

$$+ \sum_{(i,k)\in\mathcal{V}_\theta^+} \log\left(\frac{p_D(\boldsymbol{x}_i)}{1-p_D(\boldsymbol{x}_i)}\frac{p_D(\boldsymbol{y}_k)}{1-p_D(\boldsymbol{y}_k)}\right) + \ell_{IC}^+(i,k) - \ell_{IC}^-(i,k) \tag{4.31c}$$

$$+ \sum_{(i,k)\in\mathcal{V}_\theta^+} \sum_{(j,l)\in\mathcal{V}_\theta^+ : j>i} \left(\ell_\Delta^+(i,j,k,l) - \ell_\Delta^-(i,j,k,l)\right). \tag{4.31d}$$

Now, one can simply neglect Expressions (4.31a) and (4.31b) which are independent from the assignment θ. Instead, the quality of an assignment θ can simply be judged by its *relative* (log) likelihood w.r.t. this case of no inliers.

This relative log likelihood can be computed as sum over all inlier vertices and edges between them, *i.e.* Expressions (4.31c) and (4.31d). This can also be

viewed as weight of a subgraph $\mathcal{V}_\theta^+$ with vertex weights

$$\mathtt{w}(v_{ik}) = \log\left(\frac{p_D(\boldsymbol{x}_i)}{1-p_D(\boldsymbol{x}_i)}\frac{p_D(\boldsymbol{y}_k)}{1-p_D(\boldsymbol{y}_k)}\right) + \ell_{IC}^+(i,k) - \ell_{IC}^-(i,k) \quad (4.32)$$

and edge weights

$$\mathtt{w}((v_{ik},v_{jl})) = \ell_\Delta^+(i,j,k,l) - \ell_\Delta^-(i,j,k,l). \quad (4.33)$$

Viewing an assignment θ as subgraph $\mathcal{V}_\theta^+$ raises two possible issues that need to be addressed: edges with zero probability and avoidance of forbidden combinations. Since edges with zero probability have negative infinite log likelihood, they would crush any sum involving it. Instead, they can be omitted during PCG construction.

To ensure correctness and consider edges which are missing in the PCG, *e.g.* due to their negative infinite weight or to fulfill the assumption of a one-to-one association, one can retrieve only *cliques*, *i.e.* fully connected subgraphs. This ensures both finite likelihoods and restricts the results to valid assignments. As shown in the next section, existing maximum weighted clique solvers can be adapted to retrieve the optimal assignment(s) efficiently.

4.2.8 Fast Maximum Weighted Clique Retrieval

The fundamental problem to retrieve a maximum weighted clique from a graph is a generalization of the same problem for unweighted maximum cliques. This in turn is a generalization of the clique decision problem, one of the first famous NP-complete problems [Kar72]. Hence, the problem is expected to scale very badly with the amount of correspondences. However, for graphs with the typical size and density of many real problems, state-of-the-art maximum clique solvers can compute solutions in milliseconds.

Real-Valued Maximum Totally Weighted Cliques

A problem with most solvers is that they support only weighted vertices or, rarely, only weighted edges, but not both. One solution that retrieves cliques from graphs with vertex and edge weights, also referred to as totally weighted graphs, is called MECQ [SYM20]. Additionally, it was kindly made available as open source code.

MECQ is a branch-and-bound algorithm that builds upon two ideas. To incorporate edge weights, it distributes them to adjacent vertices for bounding the inclusion of putative additional vertices during branching. Furthermore, it builds upon the previously existing idea of vertex coloring to efficiently compute bounds for individual independent sets within the graph.

While the original source code and algorithm only supports positive integer weights, its working principle has proven to be extensible to support any real-valued weights. This can be implemented by replacing negative edge and vertex weights with zero weights during the coloring phase, which leads to an overestimation of upper bounds.

For this thesis, MECQ's highly efficient source code has been adapted. As this was far from trivial, an empiric verification procedure was performed. For this, several thousands of randomly weighted random graphs were generated and their maximum totally weighted clique was then computed by solving an equivalent quadratic integer programming formulation [SYM19] with Gurobi [Gur22]. While Gurobi takes about three orders of magnitude longer than MECQ, its problem formulation is trivial and, hence, its result can serve as reference to compare the correctness of the adapted MECQ.

To the best knowledge of the author, the adapted MECQ is the only solver that can efficiently retrieve totally weighted cliques for not only positive real-valued weights, but for weights from the full spectrum of finite real numbers. While for this thesis, it is only a tool to retrieve assignments, the adapted MECQ might be a valuable contribution for applications from plenty of other fields as well.

Retrieving Multiple Cliques

Adapting MECQ to real-valued weights enables the efficient retrieval of the clique with maximum total weight from a PCG, corresponding to the optimal assignment θ^* in correspondence space. However, as described in Section 4.2.5, probabilities in correspondence and in residual space are not identical, but only highly correlated. This may render the assignment which is optimal in residual space slightly suboptimal in correspondence space. Additionally, due to the ambiguity of man-made structures, clutter and missed detections, multiple assignments might be almost equiprobable.

In order to detect those cases, not only the one optimal assignment θ^*, but all assignments Θ^* within a certain likelihood window ℓ_w can be retrieved

$$\Theta^* := \{\theta \in \Theta : \ell_\theta(\theta) > \ell_\theta(\theta^*) - \ell_w\}. \tag{4.34}$$

This makes it possible to post-process the result in three ways. One can certify the optimality of the best assignment with a specifiable likelihood gap, marginalize over all best assignments, or avoid any association at all in case of ambiguities.

To retrieve the best assignments within a log likelihood window ℓ_w, the bounding part of MECQ is extended to explore and return all cliques that are at most ℓ_w less likely than the clique with the largest weight, *i.e.* the assignment with highest likelihood in correspondence space. This extension was comparatively trivial and hence not verified further. In the following, the adapted version of MECQ, extended to both real-valued weights and multiple cliques, will be referred to as MECQ++.

Overestimation Compensation

A structural comparison of Equation (4.16) and Equation (4.28) shows that in the former, the number of pairwise likelihood terms grows quadratically in the size of a hypothetical assignment while the number of similar terms grows linearly in the latter. This leads to a relative overestimation that needs to be compensated by normalizing for assignment cardinality.

While such a compensation is trivial after retrieving a clique, it would require careful adaptation of MECQ's bounding steps or weaken their tightness significantly. Instead, within MECQ++, the likelihood window ℓ_w is grown with the size of the so far largest known clique. This does yield the same set of cliques, but potentially provides tighter bounds, resulting in faster computing times.

4.2.9 Efficient PCG Construction and Approximations

The retrieval of a solution in short time is favorable for many real-world applications. For autonomous systems that need to interact with their environment, it is often even crucial. In order to offer a trade-off between solution times and exact optimality, the PCG can be pruned during construction by skipping extremely unlikely vertices, *i.e*. correspondences, or sufficiently improbable edges. This not only accelerates retrieving the best assignment, but also graph construction, which is often even more time-consuming than retrieving cliques within it.

The pruning can be implemented in two ways. The first is using a threshold on the individual and pairwise compatibility *inlier* cumulative density functions (cdfs), $F^+_{IC}(i,k)$ and $F^+_{\Delta}(i,j,k,l)$, respectively, *i.e*.

$$v_{ik} \in \mathcal{V} \iff F^+_{IC}(i,k) < \tau^{cdf}_{IC} \tag{4.35}$$

$$(v_{ik},v_{jl}) \in \mathcal{E} \iff F^+_{\Delta}(i,j,k,l) < \tau^{cdf}_{PC}. \tag{4.36}$$

This allows us to reason nicely about the probability that an inlier vertex or edge is missed, which is $1 - \tau^{cdf}$.

Alternatively, a threshold on the *relative* likelihood, which would be used as vertex and edge weight (*cf*. Equations (4.32) and (4.33)), can be applied

$$v_{ik} \in \mathcal{V} \iff \mathtt{w}(v_{ik}) > \tau^{\delta\ell}_{IC} \tag{4.37}$$

$$(v_{ik},v_{jl}) \in \mathcal{E} \iff \mathtt{w}((v_{ik},v_{jl})) > \tau^{\delta\ell}_{PC}. \tag{4.38}$$

While less expressive, in experiments, the latter thresholding turned out to be more effective in distinguishing inliers from outliers.

The thresholds, and hence the probability mass lost in this pruning, can be chosen arbitrarily conservative, trading graph construction and clique retrieval times against exactness and probabilistic optimality.

Note that this makes the pruning considerably different from previously existing binarization. While binarizing approaches need to pick the optimal threshold to balance recall and precision, the thresholds τ_{IC},τ_{PC} can be chosen arbitrarily conservative, possibly including many outliers. This is because MECQ++ does not consider the cardinality of the largest clique, but its total weight. Hence, not all vertices and edges are equal, and MECQ++ can retrieve the optimal assignment even if another clique is larger by size, but not by weight.

If retrieval times are still too high, the branching and coloring steps can be modified to abandon traversing the graph after a specified time limit. This acts as a timeout that retains a correct view on the best cliques found so far. While missing a thorough investigation, when using such early stopping, the author never observed a maximum weight clique missing. Instead, it is the suboptimal cliques which would still fall into MECQ++'s likelihood window which have been observed missing.

4.3 Probabilistic Compatibility Distributions

So far, transformation invariant measurements (TIMs) were simply assumed to be abstract, transformation invariant measures that act upon disjoint correspondence pairs. In this section, three concrete examples are presented, one of which has already been introduced briefly.

4.3.1 Squared Euclidean Distance Differences

As first TIM, the squared Euclidean distance difference (SEDD) is defined by the difference of squared Euclidean distances:

$$\Delta\delta^2((i,k),(j,l)) := \delta(\boldsymbol{x}_i,\boldsymbol{x}_j)^2 - \delta(\boldsymbol{y}_k,\boldsymbol{y}_l)^2. \tag{4.39}$$

While they perform worse than Euclidean distance differences (EDDs) in practice, SEDDs can be used as example to derive a TIM analytically and exactly. This nicely explains how and why inliers and outliers can be distinguished probabilistically in the TIM space. Due to its length, this derivation is presented in Appendix E.

Unfortunately, the author did not find the necessary distributional tools to replicate the same analytical derivation for other TIMs. While none of the steps seem impossible, it is believed that analytical descriptions of the necessary correlated difference distributions have simply not yet been derived. Given the extent of the derivations referenced in Appendix E for the SEDD case, the necessary work can be expected to fill multiple publications on its own.

4.3.2 Euclidean Distance Differences

The use of EDDs to distinguish inliers and outliers in a correspondence graph (or interpretation tree) has been proposed by many previous works. However, they either applied a hard threshold [Gri90, NTC03] to binarize the problem or used negative EDDs as measure of affinity that can then be maximized greedily [LH05] or exactly [LFH21]. In contrast, this work proposes to examine, model, and exploit the probability distribution of EDD, which is significantly different for inliers and outliers. This not only enables a sound probabilistic model with enticing guarantees. It also allows the results to be retrieved with a conscious probabilistic trade-off between provable optimality and retrieval speed.

The (absolute) difference of Euclidean distances is defined as

$$\Delta\delta_{ijkl} = \Delta\delta((i,k),(j,l)) := |\delta(\boldsymbol{x}_i,\boldsymbol{x}_j) - \delta(\boldsymbol{y}_k,\boldsymbol{y}_l)|. \tag{4.40}$$

Since an analytical derivation of EDDs does not seem to be viable at the point of writing, half-normal distributions are proposed as empirically valid approximation. The validity of this approximation has not only been confirmed for simulated normally distributed data as depicted in Figure 4.5. It also holds empirically in various real-world scenarios, *i.e.* for point cloud registration on the Stanford bunny model [TL94], for feature point alignment on the KITTI

dataset [GLU12, GLS+13], and for localization in a semantic HAD map (*cf*. Section 4.5.3).

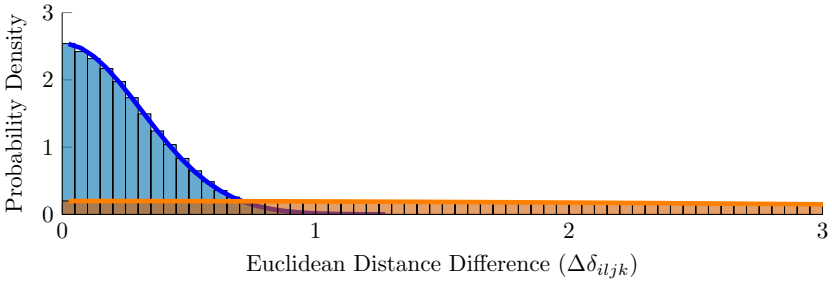


Figure 4.5: EDDs in a prototypical scenario. 3D points $\boldsymbol{X}$ were generated using isotropic zero-mean Gaussian distribution with magnitude $\sigma_{\boldsymbol{X}} = 1$. Then, isotropic zero-mean Gaussian noise $\boldsymbol{W}$ with magnitude $\sigma_{\boldsymbol{W}} = 0.05$ was added to derive $\boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{W}$. Orange shows a histogram of the EDDs among outliers, *i.e*. false correspondences, with the fitted half-normal distribution in according color. In blue, the inlier histogram, *i.e*. correct correspondences, and fitted half-normal distribution are depicted.

4.3.3 Joint Euclidean Angular Distance Differences

While EDDs work well for unoriented objects, such as points, they fail to capture the directional information that is available for oriented objects, such as planes (*e.g*. traffic signs, walls) or lines (*e.g*. poles).

As explained in Section 4.1.2, previous works have proposed plenty of methods to register oriented objects. A recently proposed method is a very generic approach that views lines and planes as elements of a Grassmannian manifolds [LH22, LPH23]. This has the advantage that the center point of a plane or line does not need to be determined. However, while achieving state-of-the-art results when using sufficiently many objects, Grassmannian space fails to exploit potentially valuable position information which could allow data association and localization using only few but accurately detected landmarks.

To exploit both orientation and position information, a novel joint Euclidean angular distance difference (JEADD) is proposed. It is reasoned probabilistically and outperforms the Grassmannian state of the art without being less generic.

The JEADD $\Delta\delta^\circ$ is defined on the spatial and orientation coordinates of each data point $\boldsymbol{x}_i$, *e.g.* corresponding to the center point $\boldsymbol{c}_i$ and normal or axis vector $\boldsymbol{o}_i$, respectively,

$$\Delta\delta^\circ_{ijkl}((i,k),(j,l)) := \begin{pmatrix} |\delta(\boldsymbol{c}_i,\boldsymbol{c}_j) - \delta(\boldsymbol{c}_k,\boldsymbol{c}_l)| \\ |\measuredangle(\boldsymbol{o}_i,\boldsymbol{o}_j) - \measuredangle(\boldsymbol{o}_k,\boldsymbol{o}_j)| \end{pmatrix}. \tag{4.41}$$

Here, $\measuredangle(\boldsymbol{o}_i,\boldsymbol{o}_j)$ denotes the angle of the minimal rotation to rotate $\boldsymbol{o}_i$ into $\boldsymbol{o}_j$.

While real-world experiments showed that angular *out*liers could be approximated better by a generalized gamma mixture or log-Cauchy distribution, both inliers and outliers are assumed half-normally distributed for simplicity. In fact, as long as the inliers are modeled correctly and there is a sufficiently large scale difference between inliers and outliers, this approximation has shown to work well in practice.

In order to compare the proposed JEADDs with the distribution of pairwise differences in Grassmannian space, as proposed by Lusk et al. [LH22, LPH23], oriented objects are sampled in 3D space with random position and orientation.

To simulate a measurement process, rotational noise sampled from a highly concentrated von Mises-Fisher distribution is applied to the orientation. Additive zero-mean Gaussian positional noise is varied in magnitude, allowing to examine both TIMs over decreasing measurement accuracy for the position of the objects.

Comparing the JEADD and affine Grassmannian distance between correctly and incorrectly associated correspondences between the original and the noisy object set allows computing statistical distributions of inliers and outliers, respectively. As metrics, one can compare the symmetric Kullback-Leibler divergence (KLD) between the inlier/outlier distribution of each TIM and the F_1 score of an optimal 1D linear classifier to distinguish inliers from outliers.

To evaluate the approaches fairly, the scaling factor required for affine Grassmannian distances is optimized by sampling from a logarithmic space of size 50.

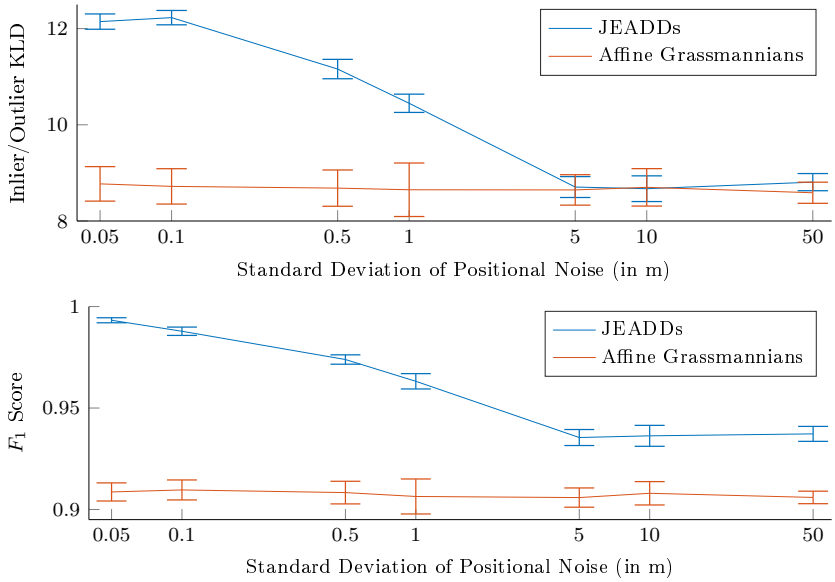


Figure 4.6: Comparison of joint Euclidean angular distance differences (JEADDs) against affine Grassmannian distances, which are the current state of the art for oriented object registration.
Over exponentially increasing positional noise, in the upper plot, the Kullback-Leibler divergence (KLD) between the respective inlier/outlier distribution of each distance measure is depicted. The lower plot shows the F_1 score of an optimal 1D classifier to separate inliers and outliers based on the respective TIM. Both plots depict the mean $\pm$ one standard deviation over 10 experiments.

For simplicity, the threshold which classifies inliers and outliers is determined by sampling, too, using a linear space of size 100. Both variables were checked for their insensitivity.

As depicted in Figure 4.6, in terms of KLD, the proposed joint Euclidean angular distance difference can distinguish inliers and outliers far better for any remotely useful position measurement. Asymptotically, when orientation information is far more relevant than any position measurement, its performance converges against that of distances in Grassmannian space. This proves their advantage for probabilistic applications such as in a PCG.

In terms of F_1 score, JEADDs dominate over the full range of position noise. This shows that the proposed distance measure can be assumed superior even for approaches that rely on a binarizing threshold.

4.4 Interpretation

Given a PCG filled with suitable TIMs, the adapted clique solver MECQ++ retrieves the assignments $\theta \in \Theta^*$ that are at most a predefined log likelihood window ℓ_w worse than the best assignment θ^*. While θ^* is the best solution in correspondence space, slightly suboptimal solutions with entirely different associations might exist. This can be either due to ambiguous landmark constellations or due to the non-identity of correspondence and residual space.

This section presents how to evaluate the set of assignments Θ^* which involves computing a transformation for each assignment, computing the likelihood in residual space, marginalizing over Θ^*, and the actual self-assessment.

4.4.1 Pose Estimation

Considering only associated center points $\boldsymbol{c}$ would offer a number of readily available closed-form solutions [Hor87, AHB87, Ume91] to compute the 6D ego pose. However, uncertainties, such as in the z component of pole center points, can induce errors that could be resolved by exploiting additional orientation information. To do so, a two-stage approach is proposed.

Weighted Alignment of Points and Oriented Objects

As first stage and initial estimate a weighted registration is used. A version of the approach by Umeyama [Ume91] was adapted by Karney [Kar14] to include weights, minimizing the residual

$$J^{ik}(\boldsymbol{R}, \boldsymbol{t}) = \mathtt{w}_{ik} \|\boldsymbol{c}_k - (\boldsymbol{R}\boldsymbol{c}_i + \boldsymbol{t})\|^2 \tag{4.42}$$

using the corresponding vertex and edge weights as residual weights

$$\mathtt{w}_{ik} = \mathtt{w}(v_{ik}) + \sum_{(j,l)\in\mathcal{V}_\theta} \mathtt{w}((v_{ik},v_{jl})). \tag{4.43}$$

Note that this involves the knowledge that all vertices in θ are part of a clique and, hence, direct neighbors. The author has to admit that this log likelihood weighted registration has no proper reasoning other than being an intuitive heuristic that exploits knowledge from correspondence space. While it improved registration results in some examples, statistical tests failed to prove a significant advantage over the unweighted Umeyama alignment.

In addition to these weights from correspondence space, as proposed by Lusk and How [LH22], one can exploit the orientation information of oriented objects by minimizing the relative orientation differences corresponding to the residual

$$J(\boldsymbol{R},\boldsymbol{t}) = \mathtt{w}_{ik}\|\boldsymbol{o}_k - \boldsymbol{R}\boldsymbol{o}_i\|^2. \tag{4.44}$$

This can be implemented by adding purely rotational terms to the singular value decomposition (SVD) matrix Σ_{XY} (*cf*. Eqn. (38) in [Ume91]).

As approximation of the certainty of orientation information, the spatial extent perpendicular to the orientation vector can be used if known. In case of signs and poles, this is their width and height, leading to the residuals

$$J^{ik}_{\text{pole}}(\boldsymbol{R},\boldsymbol{t}) = \mathtt{w}_{ik}\|h_k\boldsymbol{o}_k - \boldsymbol{R}h_i\boldsymbol{o}_i\|^2 \tag{4.45}$$

$$J^{ik}_{\text{sign}}(\boldsymbol{R},\boldsymbol{t}) = \mathtt{w}_{ik}\|w_k\boldsymbol{o}_k - \boldsymbol{R}w_i\boldsymbol{o}_i\|^2. \tag{4.46}$$

Robust Local Non-Linear Alignment

As second stage for the alignment of oriented objects, the initial estimates of $\boldsymbol{R},\boldsymbol{t}$ are used to locally solve a robustified non-linear alignment of the associated measurements and landmarks using the ceres solver [AMT20]. It minimizes the covariance-weighted residual

$$J^{ik}(\boldsymbol{R},\boldsymbol{t}) = \rho\Big(\|\boldsymbol{c}_k - \boldsymbol{R}\boldsymbol{c}_i + \boldsymbol{t}\|^2_{\boldsymbol{\Sigma}_{\mathbf{c},\mathfrak{c}}} + \measuredangle(\boldsymbol{o}_k,\boldsymbol{R}\boldsymbol{o}_i)^2_{\sigma^2_{\measuredangle,\mathfrak{c}}}\Big). \tag{4.47}$$

The robustifier ρ is the Cauchy loss function and for simplicity, uncorrelated errors are assumed. This leaves five hyperparameters per class $\mathfrak{c}$: the Cauchy scale and four variances for x, y, z and the relative angle. As recommended by ceres for its manifold parametrization, orientation differences $\measuredangle(\cdot)$ are translated into residuals via the 3D vector part of the quaternion which describes the minimal rotation delta. Each vector component is weighted equally with the inverse relative angular variance.

In practice, the robustified solution was able to outperform the closed-form solution especially in case of grave detection errors, *e.g.* if parts of a pole or traffic sign were occluded. For the alignment of unoriented point clouds, a similar but simpler iterative least squares method [HW77, BE14] is commonly used.

4.4.2 Residual Space Evaluation

While, as explained in Section 4.2.5, probabilities in correspondence and residual space are highly correlated, there are cases in which small shifts in landmarks are not reflected well in correspondences space since the direction of change is orthogonal to the distances between objects. Exemplary, this is observed when a sign's mounting position has changed slightly, but all other landmarks are tens of meters away.

This can be avoided by computing a transformation T_θ for each assignment $\theta \in \Theta^*$ retrieved in correspondence space. Using these transformations, one can then re-evaluate all assignments in residual space according to Equation (4.28). This combines the advantages of fast retrieval of assignment candidates in a well-aligned likelihood space with the actual evaluation using a proven likelihood formulation.

4.4.3 Ambiguity-Awareness by Marginalization

In case of false detections, map changes or ambiguous structures, even the best pose estimate can be wrong and should be discarded. The autonomous system can then either fall back to *e.g.* odometry measurements or deactivate

functions adaptively. Both is better than falsely relying on a wrong localization or map verification result.

Naively, one could exclude false detections by requiring a sufficiently large number of matched landmarks which corresponds to an inflated probability that includes a wide safety margin. However, when relying on comparatively sparse semantic landmarks instead of sensor-specific descriptors, this heavily impairs availability. This becomes especially obvious on rural roads.

At the same time, even a large number of matched landmarks can still be ambiguous. This is typically the case on highways with plenty, but periodic and hence spatially ambiguous road markings and other infrastructure.

Instead, this work proposes to compute the *marginalized* distribution, marginalizing over the best associations $\theta \in \Theta^*$. This makes it possible to include the safety margin in the choice of the best association set Θ^*. While that increases computing times to retrieve Θ^*, it in turn enables localization even with as few as five landmarks if and only if their constellation is locally sufficiently unique.

4.4.4 Marginal Distribution of the Ego Pose

To determine whether the results in Θ^* are unambiguous, the marginalized pose can be computed. Chapter 5 will show how similarly marginalized evidences can be used to verify map elements.

More specifically, the marginalized distribution $f_T(T)$ is approximated by a mixture over the most likely assignments Θ^*

$$f_T(T) = \frac{1}{\sum_{\theta \in \Theta} f_\theta(\theta)} \sum_{\theta \in \Theta} f_\theta(\theta) f_{T|\theta}(T \mid \theta) \tag{4.48}$$

$$\approx \frac{1}{\sum_{\theta \in \Theta^*} f_\theta(\theta)} \sum_{\theta \in \Theta^*} f_\theta(\theta) f_{T_\theta}(T_\theta). \tag{4.49}$$

The prefactor normalizes the sum over unnormalized likelihood terms $f_\theta(\theta)$, yielding an approximated but proper density.

The absolute likelihood $f_\theta(\theta) = \exp(\ell_\theta(\theta))$ might be too large for typical numeric types. Instead, it can be computed by splitting it into a part $f_\theta^\Delta(\theta)$ which is relative to the assignment $\theta_0 = \arg\min_{\theta \in \Theta^*} \ell_\theta(\theta)$ with the smallest likelihood in the retrieved log likelihood window

$$f_\theta(\theta) = \exp(\ell_\theta(\theta)) = \exp\big(\ell_\theta^\Delta(\theta) + \ell_\theta(\theta_0)\big) \tag{4.50}$$

$$= \exp\big(\ell_\theta^\Delta(\theta)\big) \exp(\ell_\theta(\theta_0)). \tag{4.51}$$

The common factor $\exp(\ell_\theta(\theta_0))$ can be eliminated, leaving a numerically manageable $f_\theta^\Delta(\theta) = \exp\big(\ell_\theta^\Delta(\theta)\big)$. Less likely assignments $\theta \notin \Theta^*$ are neglected. While their total probability mass might be significant, each individual contribution to the marginal distribution is not.

One challenge is the efficient representation of the distribution f_T of $T \in \mathrm{SE}(3) \simeq \mathbb{R}^3 \times \mathrm{SO}(3)$. While the translational part can be represented using a 3D normal distribution, the distribution of rotations is not as trivial. A mathematically sound solution could be a von Mises-Fisher *matrix* distribution [Dow72, KM77], but the author failed to grasp the existing publications to a degree that allowed him to derive trustworthy formulas for estimating its parameters.

Instead, three independent von Mises-Fisher distributions [MJ00] are used that each model the directional distribution of each of the orthogonal axes using the corresponding unit vectors $\boldsymbol{u}_x, \boldsymbol{u}_y, \boldsymbol{u}_z$ on the unit sphere $\mathbb{S}^2 \subset \mathbb{R}^3$. Every such distribution, for $\boldsymbol{u}$ be any of $\boldsymbol{u}_x, \boldsymbol{u}_y, \boldsymbol{u}_z$, has the density

$$p(\boldsymbol{u} \mid \boldsymbol{\mu}, \kappa) = \frac{\kappa}{4\pi \sinh(\kappa)} \exp(\kappa \boldsymbol{\mu}^T \boldsymbol{u}). \tag{4.52}$$

Its mean direction can be estimated from probability weighted samples $\boldsymbol{u}_i$ via

$$\hat{\boldsymbol{\mu}} = \frac{\bar{\boldsymbol{u}}}{\|\bar{\boldsymbol{u}}\|} \tag{4.53}$$

using the sample mean direction

$$\bar{\boldsymbol{u}} = \frac{1}{\sum_i p(\boldsymbol{u}_i)} \sum_i p(\boldsymbol{u}_i) \boldsymbol{u}_i. \tag{4.54}$$

The concentration κ can be approximated [Sra11] from

$$\hat{\kappa}_0 = \frac{\bar{R}(3 - \bar{R}^2)}{1 - \bar{R}^2} \tag{4.55}$$

$$\bar{R} = \|\bar{\boldsymbol{u}}\| \tag{4.56}$$

following a couple of Newton steps

$$\hat{\kappa}_{k+1} = \hat{\kappa}_k - \frac{A_3(\hat{\kappa}_k) - \bar{R}}{1 - A_3(\hat{\kappa}_k)^2 - \frac{2}{\hat{\kappa}_k} A_3(\hat{\kappa}_k)}. \tag{4.57}$$

For the implementation and assuming $\boldsymbol{u} \in \mathbb{S}^2$, it is worth noting that $A_3(\hat{\kappa}_k)$ should be approximated using the Langevin function [MJ00]

$$A_3(\hat{\kappa}_k) = \frac{1}{\tanh \hat{\kappa}_k} - \frac{1}{\hat{\kappa}_k}. \tag{4.58}$$

The fraction of modified Bessel functions, which is stated by Mardia and Jupp [MJ00] for the general case, cannot be resolved well numerically.

4.4.5 Self-Assessment

While it is admitted that a statistically more rigid self-assessment should involve statistical tests, *e.g.* of normality, in this work, concentrated normal and von Mises-Fisher distributions, respectively, are assumed. This allows testing only their concentration parameters. For the normal distribution that models the 3D translation, maximum thresholds on the standard deviations in x, y and z direction are introduced:

$$\sigma_x < \tau_{\sigma_x}, \quad \sigma_y < \tau_{\sigma_y}, \quad \sigma_z < \tau_{\sigma_z}. \tag{4.59}$$

This makes it possible to be more flexible in the z direction which can be difficult to estimate accurately.

Since the three directional distributions jointly observe the rotations around two axes, *e.g.* $p(\boldsymbol{u}_x \mid \Theta^*)$ observes pitch and yaw, a common minimum threshold

across all rotational concentrations, κ, is used:

$$\kappa_x > \tau_\kappa, \quad \kappa_y > \tau_\kappa, \quad \kappa_z > \tau_\kappa. \tag{4.60}$$

As final requirement, at least one assignment needs to exceed a minimal size and a likelihood threshold

$$\exists\, \theta \in \Theta^* : |\theta| > n_{\min} \wedge \ell_\theta > \ell_{\min}(|\theta|). \tag{4.61}$$

Only if all inequalities are satisfied, one can assume that the ego pose has been estimated with a sufficient accuracy and only then the localization result is assumed unambiguous enough to be accepted.

4.5 Evaluation

The performance of the proposed approach can be evaluated using three different settings. First, using an artificial simulation environment, PCG can be compared with various other state-of-the-art approaches for data association. Second, for the well-established task of point cloud registration, its performance on the KITTI dataset [GLU12, GLS+13] can be evaluated. Finally, the parametric detections proposed in Chapter 3 and maps created from them can be combined for a highly accurate 6D localization in an HAD map.

On simulation results, PCG is parameterized using the nominal values of simulated distributions. The hyperparameters used for real-world point cloud registration and localization in HAD maps are tuned using the optimization idea presented in Section 3.8. Details are described in Appendix F.

4.5.1 Simulation Results

Evaluating PCG against state-of-the-art frameworks for data association in a setting with known data and noise distributions allows parameterizing all of

them correctly and without possibly unfair or noisy hyperparameter tuning. Additionally, it enables computing an F_1 score to evaluate association quality.

As 3D model, the Stanford bunny [TL94], scaled to the unit cube, is used. This allows sampling inlier points as well as two disjoint sets of outlier points. One of them and the inliers are combined to make up the source points. The inliers are copied and merged with the other outlier set to form the target point cloud. The target points are transformed with a random transformation and zero-mean additive noise is applied. As MAGSAC++ and TEASER++ assume bounded noise, it is sampled from an isotropic truncated normal distribution [Bot17, Bru22] with standard deviation σ that is capped such that every noise vector has at most length 3σ. As example is depicted in Figure 4.7.

Under this setting, PCG can be compared against three state-of-the-art methods for robust data association, CLIPPER [LFH21], MAGSAC++ [BMN19, BNI+20, BNM22], and TEASER++ [YC19, YSC21], as well as two traditional maximum compatible clique algorithms. The first one, called MCC, uses the TIMs proposed for TEASER++ combined with PMC [RGG+14] for clique retrieval. For the second, called uPCG, the same probabilistically motivated graph as for PCG is constructed. In contrast to PCG, uPCG ignores edge weights during retrieval but selects the clique of maximum cardinality instead. While SC2-PCR [CSY+22] and Go-ICP [YLJ13, YLC+16] were tried as well, the author was unable to find parametrizations that yielded competitive results. It is assumed that both are better suited for larger problems.

To compare the approaches, three metrics are used. Association quality can be measured via F_1 score. The accuracy of the estimated transformation is evaluated via a combined error which adds translational in m and rotational error in rad with a weight of $10{:}1$ which puts typical errors in the same order of magnitude. Finally, computational effort is compared in walltime measured on an Intel i7-8565U CPU.

Two of the baseline methods offer self-assessment. As MAGSAC++ claims to self-assess its performance, all results indicated as failure are discarded. TEASER++ offers certification which takes significantly longer than the registration. Hence, the certified version is measured separately as TEASER++C. If

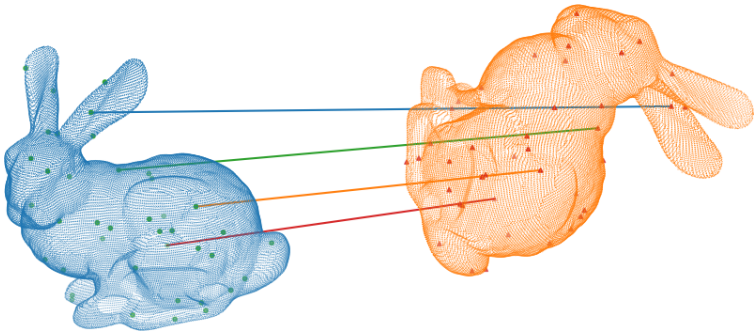


Figure 4.7: Example of the data association problem on the Stanford bunny model [TL94]. Bold points are sampled from the 3D model to yield unconnected outliers and connected inliers, the latter of which are subject to noise. The goal is to identify the true matching inliers without any given correspondences.

the certification failed, the output is discarded. PCG and uPCG reject unlikely assignments by only retrieving cliques with positive total weight, *i.e.* correspondence space log likelihood, as well as by detecting failures determined by a negative residual space log likelihood. Additionally, the transformation errors of all methods were discarded when two or fewer inliers were found.

While MAGSAC++ and TEASER++ yield both association inliers and a transformation, uPCG, CLIPPER and MCC only retrieve inliers. Thus, an Umeyama [Ume91] alignment is used with their inlier sets to determine a transformation. For PCG the weighted Umeyama alignment proposed in Section 4.4.1 is used to exploit knowledge from correspondence space.

To parameterize PCG, detection probabilities p_D are set according to the true inlier/outlier ratios, TIM distributions $f(\Delta\delta)$ are calculated from separate, statistically identical "training" sets and uninformed individual consistency, *i.e.* $f_{IC} = 1$, is assumed. The cdf threshold for PCG sparsification introduced in Section 4.2.9 was set to $\tau_{PC}^{cdf} = 0.99999$. All experiments were conducted 100 times.

Unknown Correspondences

For unknown correspondences, *i.e.* no feature space or similar prior matching, $N = 40$ points are sampled at increasing outlier rates. Truncated Gaussian noise of magnitude $\sigma = 0.002$ is added as described above. The likelihood window is chosen as $\ell_w = 2$ and scaled linearly with the inlier set. While PCG, uPCG and CLIPPER natively support fully unknown correspondences, all possible correspondences are created as potential correspondences for MAGSAC++, MCC and TEASER++.

Figure 4.8 shows that PCG is not only optimal in theory. It is the strongest approach in F_1 score even at high outlier rates and translates the assignments in leading transformation errors.

While CLIPPER is fast, it cannot compete with either PCG or TEASER++. TEASER++ can be seen as the strongest competitor for PCG with low solution times at good results up to $80\,\%$ outliers. MAGSAC++ requires at least 1 million iterations, listed as MAGSAC++1M, to match the results of other approaches. In terms of computation times, it is not competitive at all.

Comparing MCC and uPCG shows that PCG's statistical inlier selection already improves result quality significantly. At the same time, despite using MECQ as single-threaded clique solver, it is faster than MCC which uses a parallelized solver.

When focusing on certifiable or self-assessed success, all three approaches, MAGSAC++ and TEASER++, but also PCG seem to fail utterly, leading to zero F_1 scores and catastrophic transformation errors. However, this can only truly be said for MAGSAC++. In fact, TEASER++C successfully detects about $70\,\%$ of its failures which occur at $90\,\%$ outlier rate, but its certification procedure takes considerably longer than real-time applications can allow. Below, PCG's seeming failures are discussed in detail.

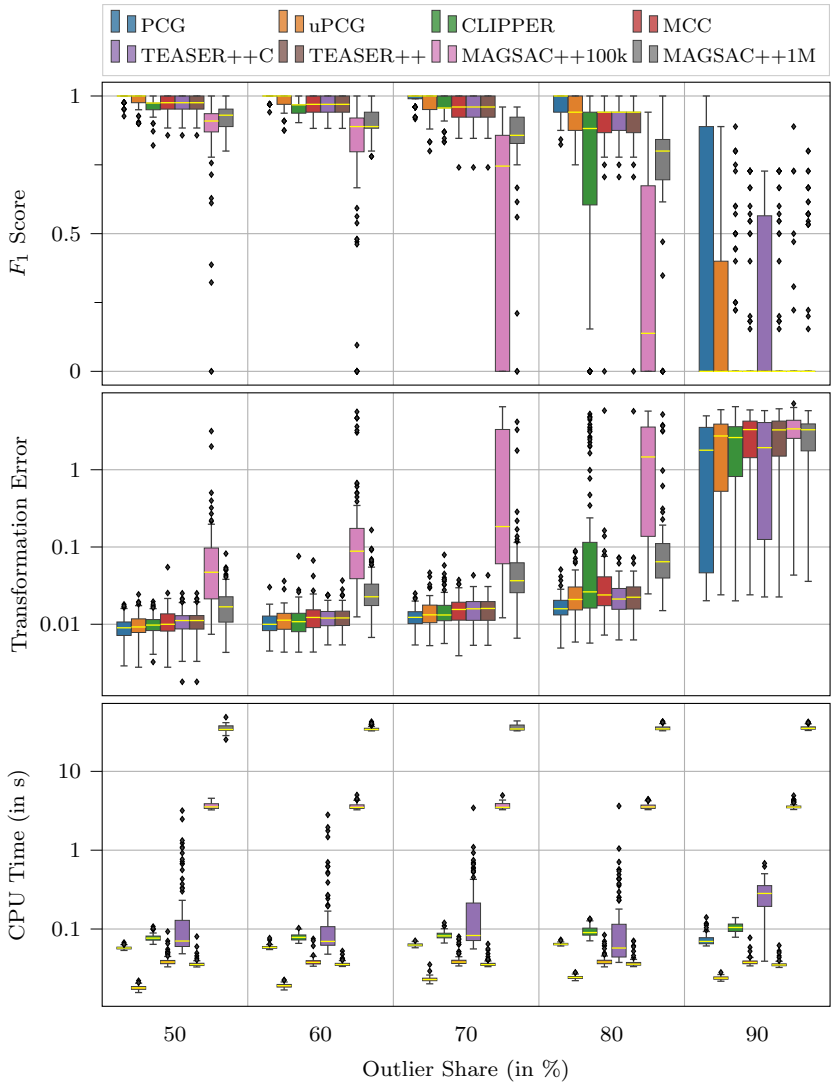


Figure 4.8: Simulation results for $N = 40$ unknown correspondences on the Stanford bunny model [TL94]. F_1 score, transformation error, and computation time are plotted over an increasing amount of outliers. Note that the y axes for the latter two metrics are scaled logarithmically.

Outlier Rejection for Known Correspondences

When correspondences are known, but possibly contain outliers, *e.g.* using feature descriptors, the correspondence graph is quadratically smaller. This makes it possible to associate $N = 500$ correspondences at higher noise rates of $\sigma = 0.005$. PCG's likelihood window is chosen as $\ell_w = 100$ and not scaled.

When comparing the plot in Figure 4.9, it becomes apparent that this problem is easier for most approaches. Only in the most difficult setting, with only 5 correct correspondences, all approaches but PCG fail.

MAGSAC++'s early stopping makes it competitive, even in computation times, for up to $90\,\%$ outliers. The best competitor to PCG is still TEASER++. However, in this setting, TEASER++'s certification takes even longer. Additionally the experiment showed that at $99\,\%$ outliers it rejected $42\,\%$ of cases without leading to better results than the vastly faster uPCG. In summary, PCG's performance is superior across all level of outliers. Its only issue might be the runtime in few cases, which could be traded against optimality as described in Section 4.2.9.

Failure Analysis

When looking at the transformation errors failure cases become apparent. At first glance, this seems to contradict PCG's probabilistic optimality claims. Analyzing those cases shows that there are three possibilities why the true assignment has not been retrieved.

First, due to thresholding when constructing the graph (*cf.* Section 4.2.9), the true assignment might not be fully connected in the PCG and, hence, cannot be retrieved. This can be mitigated by increasing τ_{PC}^{cdf}, primarily at the cost of slower graph construction.

The second source of failures is the misalignment of correspondence space and residual space likelihoods. This happens especially for very small inlier sets. It can be mitigated by increasing the likelihood window ℓ_w for clique retrieval, but negatively impacts retrieval times.

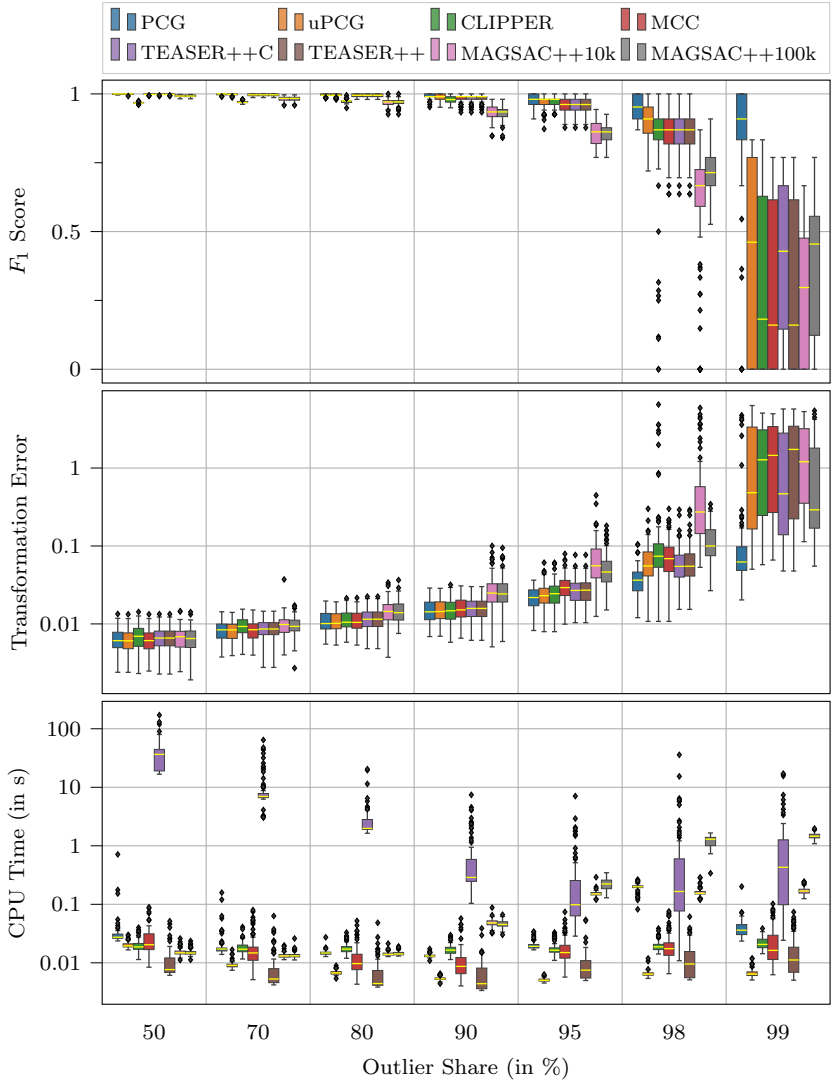


Figure 4.9: Simulation results for $N = 500$ given correspondences on the Stanford bunny model [TL94]. F_1 score, transformation error, and computation time are plotted over an increasing share of wrong correspondences. Note that the y axes for the latter two metrics are scaled logarithmically.

Finally, the source of failures with the by far largest share are false assignments which are not only better in correspondence space, but even have a larger log likelihood in residual space. This is an artifact from the random experiment generation. Due to very strong noise and especially for few inliers, there is a significant probability that some other assignment fits better than the true one. It has deliberately not been corrected since not every method uses residual space likelihoods, hence, potentially giving the proposed PCG an unfair advantage. However, if those cases were excluded, PCG would either reach a perfect F_1 score or self-assess its failure as expected.

4.5.2 Point Cloud Registration

Since the artificial simulation environment could fail to replicate real-world problems, in this section, data from the KITTI odometry benchmark [GLU12, GLS+13] is used to perform point cloud registration between 555 pairs of lidar scans. In recent years, this problem has been widely used as standard benchmark for point cloud registration approaches.

As the registration can only be solved on raw lidar points if a pose prior is known, which often is not the case, point cloud descriptors are used to pre-compute correspondences between two lidar scans. While novel point cloud descriptors are still being researched [HGU+21], for this benchmark, a well-established combination of traditional FPFH [RBB09] and deeply learned FCGF [CPK19] descriptors is used.

Previous state-of-the-art approaches [BLZ+21, CSY+22] have shown that solutions can only be achieved in reasonable time when using a subset of correspondences to estimate an initial pose that is then refined locally using an iterative least squares approach [HW77, BE14]. Hence, for the task of point cloud registration, this procedure has been adapted to be used with PCGs.

The individual consistency of correspondences is evaluated using the similarity of FPFH and FCGF feature descriptors. While more complex probability distributions are imaginable and could certainly improve performance, for simplicity, a uniform distribution is fitted. It basically restricts the feature

descriptors to the similarity region that maximizes inlier density relative to the density of outliers. Since the feature points are unoriented objects, a purely Euclidean distance difference (EDD) is used to measure pairwise consistency.

To compare approaches, the so-called registration recall (RR) which measures the share of successful registrations is the primary metric. For KITTI, a registration is commonly considered successful when the rotation error (RE) is smaller than $5°$ and the translation error (TE) is smaller than $60\,\mathrm{cm}$. Additionally, rotation error, translation error, and computation time are reported by most approaches.

The evaluation is done using the framework provided by Bai et al. [BLZ+21] and Chen et al. [CSY+22], respectively. This allows listing numbers of older baselines as reported in [CSY+22]. More recent methods used comparable setups. Hence, their numbers are from the respective papers [ZYZ+23, CSY+23]. Since the author of this thesis was unable to reproduce the results of Zhang et al. [ZYZ+23] with the available source code, their numbers have been taken from the paper which unfortunately does not report any timing information.

Table 4.4: Point cloud registration performance on the KITTI odometry benchmark. The reported metrics are registration recall (RR), rotation error (RE), translation error (TE), and computation time. Best numbers are marked bold, second best numbers are underlined. The first section contains deep learning approaches while the second part lists conventional methods.
†: As reported by Chen et al. [CSY+22]. ‡: No time reported, see text.

	FPFH			**FCGF**			
	RR (in %)	**RE** (in °)	**TE** (in cm)	**RR** (in %)	**RE** (in °)	**TE** (in cm)	**Time** (in s)
DHVR [LKC+21]†	-	-	-	<u>99.10</u>	**0.29**	19.80	0.83
PointDSC [BLZ+21]†	98.20	0.35	8.13	98.02	0.33	21.03	0.45
FGR [ZPK16]†	5.23	0.86	43.84	89.54	0.46	25.72	3.88
CG-SAC [QY20]†	74.23	0.73	14.02	83.24	0.56	22.96	0.73
SC^2-PCR [CSY+22]	<u>99.64</u>	<u>0.32</u>	7.23	98.20	0.33	20.95	<u>0.31</u>
SC^2-PCR++ [CSY+23]	<u>99.64</u>	<u>0.32</u>	<u>7.19</u>	98.56	<u>0.32</u>	20.61	0.86
MAC [ZYZ+23]	99.46	0.40	8.46	97.84	0.34	<u>19.34</u>	-‡
PCG (proposed)	**99.82**	**0.29**	**6.68**	**99.28**	0.38	**18.90**	**0.04**

The results in Table 4.4 show that PCG represents a new state of the art for point cloud registration on the KITTI benchmark, outperforming both deeply learned and traditional approaches on almost every metric for result quality. In terms of computation time, it is the only state-of-the-art approach that can match lidar scans faster than they are produced at $10\,\mathrm{Hz}$.

Additionally, in contrast to most previous methods, PCG can detect ambiguities and failure cases when parameterized accordingly. For FPFH features, this works well with a RR only slightly reduced to $98.74\,\%$. Hence, next to leading performance in best effort registration, PCG makes it possible to detect and avoid all potentially catastrophic failure cases with barely degraded availability.

Unfortunately, using FCGF features, no parametrization could be found that enables the detection of all failure cases at acceptable RR rates. This hints towards a structural weakness of FCGF descriptors which, in some cases, simply lead to an overwhelming number of false correspondences.

4.5.3 Localization for Autonomous Vehicles

In order to evaluate the proposed approach for localizing an autonomous vehicle in a pre-built map, both suitable ground truth and metrics need to be defined.

Localization Ground Truth as Open Problem

Evaluating localization performance requires a reference system that is superior to the system under test. Acquiring such ground truth for state-of-the-art localization in all six degrees of freedom is challenging. Limited range or indoor SLAM/odometry benchmarks use motion capture systems for highly accurate 6D ground truth [BNG+16, SGD+18], optionally paired with laser trackers for highly accurate positions [BNG+16]. Due to the necessary line of sight to one or more stationary reference points, this is hardly feasible for robotic vehicles. Here, pose measurements from an RTK-GNSS/INS unit are commonly used [GLU12, GLS+13].

Hence, using the OxTS RT3000 RTK-GNSS/INS unit available in the 2023 "Joy" sensor setup seems to be an obvious choice. Indeed, experiments by the author showed that such a system is globally almost entirely driftless.

However, the limitation of using the OxTS RT3000 as reference becomes obvious when considering the self-estimated accuracy, which is about $30\,\mathrm{cm}$ on average and $10\,\mathrm{cm}$ at best in each direction despite a dynamic driving warm-up before recording. Even worse, probably due to multipath effects, in an urban environment its poses jump with the same order of magnitude even during standstill.

A complementary alternative are solutions that provide a *locally* highly accurate 6D motion. One example is a state-of-the-art lidar odometry, like KISS-ICP [VGM+23] introduced in the previous chapter. Compared in a local coordinate frame over limited time, as illustrated in Figure 4.10, it is at least as accurate as poses from an expensive GNSS reference system.

Comparing Figures 4.10b and 4.10c with Figure 4.10d shows that neither the OxTS RT3000 RTK-GNSS/INS nor the state-of-the-art KISS-ICP framework achieve the necessary superiority commonly required for ground truth systems.

Still, while being a pure odometry method that has global drift, for local 6D motion as used for mapping, KISS-ICP seems to be superior to the OxTS RT3000. This holds for localization with high accuracy in all six degrees of freedom, but can also be evaluated using the Rendering Instance IoU (RIIoU) metric proposed in Section 3.7, which measures the reprojection error of map elements. To compare the quality of both possible references, maps were created and evaluated from the same sensor data but using either GNSS or KISS-ICP poses. The experiment showed that maps created using the GNSS poses, with and without additional refinement using the localization proposed in this chapter, have about $15\,\%$ lower RIIoU score compared to the same map created using KISS-ICP poses.

Both qualitative and quantitative differences could also be explained by calibration or synchronization problems between GNSS and lidar or camera rather than a systematic superiority of KISS-ICP. This would however still suggest

(a) Overview with crop location highlighted in red

(b) OxTS RT3000 RTK-GNSS/INS poses

(c) KISS-ICP [VGM+23] lidar odometry poses

(d) KISS-ICP poses with PCG refinement (proposed)

Figure 4.10: Comparison of possible localization reference systems. The 6D poses of each system / method were used to create a map as described in Chapter 3.
Projecting traffic lights and signs from about 150 m distance into the camera image reveals slight, but visible differences. In particular the projections of the traffic lights mapped using OxTS RT3000 poses, as depicted in Figure (b), lie almost entirely top left of the true elements.

using KISS-ICP over GNSS poses as reference since it runs directly on lidar data as does the localization method under test.

Unfortunately, the unsuitability of RTK-GNSS/INS systems as reference prevents using poses in a global coordinate frame. Since a lidar odometry only outputs relative poses in local coordinates, any comparison across drives is futile. As solution, this work proposes a metric that only requires locally accurate poses.

Localization Metrics as Solution

Despite the absence of a true global ground truth, the performance of a localization algorithm can be evaluated using two metrics. The first one is the map mean RIIoU (mRIIoU) w.r.t. a localization result T as explained in Section 3.7.3. By rendering the same map into an image using different localization methods, it compares the reprojection error using semantic instance detections. However, since this metric evaluates each frame independently, it is not robust against simulated or actual map changes. It would be fine with a wrong localization result as long as the overlap between reprojected map and semantic instances is high.

The reprojection could be limited to the unchanged parts of the map, but this would require ground truth knowledge about the changes, which is only known for simulated, but not for real changes. Instead, it is complemented with another novel metric, the so-called delta pose error.

Pose Error

Only when the sensor data from the very same drive that was used for mapping is also used for localization, an absolute pose can be assumed known and an absolute pose error can be computed. The absolute pose error can be measured in each coordinate ξ separately by comparing the pose at frame i estimated by PCG, $\hat{T}_i \in \mathrm{SE}(3)$, with the (pseudo) ground truth pose, $T_i \in \mathrm{SE}(3)$:

$$e_i^{\xi} := \left\| \hat{T}_i^{\xi} - T_i^{\xi} \right\|. \tag{4.62}$$

Taking the average along a drive is referred to as average (absolute) pose error (APE) and defined as

$$\overline{e}^{\xi} := \frac{1}{N} \sum_{i=1}^{N} e_i^{\xi}. \tag{4.63}$$

Additionally, as upper bound, one can define the maximal pose error along a drive, called maximum (absolute) pose error (MPE), which is defined as

$$^{+}e^{\xi} := \max_{i \in [1,N]} e_i^{\xi}. \tag{4.64}$$

Delta Pose Error

Comparing absolute poses between drives, sometimes years apart, would require a highly accurate, sensor agnostic, and long-term stable multi-drive mapping framework. For this challenging combination of requirements, the author is not aware of any readily available solutions. Instead, to evaluate localization performance in a map created in a different drive, delta poses $\delta\hat{T}_{ij} \in \mathrm{SE}(3)$ can be computed between localization results $\hat{T}_i$ and $\hat{T}_j$. The delta poses are defined via the relation

$$\hat{T}_j = \delta\hat{T}_{ij} \oplus \hat{T}_i. \tag{4.65}$$

These delta poses can then be compared with the respective delta poses from the KISS-ICP odometry, δT_{ij}. Since the PCG localization has no knowledge of the odometry, the odometry poses can serve as an independent reference. A suitable delta pose can only be achieved if both poses, from which the delta pose is computed, are correct.

The difference of estimated and odometry delta poses can then be used as delta pose error (DPE) δe_{ij}. As a full $\mathrm{SE}(3)$ pose is difficult to interpret, one or more meaningful parameters ξ, like translation or yaw angle, can be considered independently

$$\delta e_{ij}^{\xi} := \left\| \delta\hat{T}_{ij}^{\xi} - \delta T_{ij}^{\xi} \right\|. \tag{4.66}$$

Note that this difference is implemented in the respective parametrization space, *i.e.* on 2D or 3D translation vectors, yaw angles or smallest 3D rotations.

Naively, one can compute it between two successive localization results, *i.e.* $j = i + 1$, and average it over a drive with N measurement frames. This is

referred to as average delta pose error (ADPE) and defined as

$$\overline{\delta e}^{\xi} := \frac{1}{N-1} \sum_{i=1}^{N-1} \delta e_{i,i+1}^{\xi}. \tag{4.67}$$

The probability that two 6D poses occur by chance, that are wrong in absolute values but match with high accuracy relative to each other, is almost negligible. Yet it has been observed for very few successive poses at overly optimistic parametrizations.

While it was not observed for the parameters used for evaluation, it motivated a second metric, called maximum delta pose error (MDPE). The MDPE measures the maximum of all pose errors between pose pairs within a window of ± 10 poses relative to a localization result:

$${}^{+}\delta e^{\xi} := \max_{i \in [1,N]} \max_{j \in [i-10, i+10] \setminus i} \delta e_{ij}^{\xi}. \tag{4.68}$$

Comparing both, the ADPE can be seen as locally accurate measure of typical deviations between two poses while the MDPE is an upper limit of two localization errors.

Both error terms are interesting in four variants. For most driving tasks, planar translational and yaw error, ADPE^{xy}/MDPE^{xy} and ADPE$^{\varphi}$/MDPE$^{\varphi}$, respectively, are most relevant. Additionally, 3D translational and 3D rotational error are reported, being referred to as ADPE$^{\boldsymbol{t}}$/MDPE$^{\boldsymbol{t}}$ and ADPE$^{\boldsymbol{R}}$/MDPE$^{\boldsymbol{R}}$, respectively.

Localization In Mapping Drives

To confirm the suitability of delta pose errors as metric, they can be compared with absolute pose errors which are available if and only if the same drive is used for both mapping and localization.

Tables 4.5 and 4.6 show that the delta pose errors are indeed a valid metric as they approximate absolute errors, that are measured w.r.t. pseudo ground truth

mapping poses, with high precision and with a tendency to overestimate them. This justifies their use in the following, actually meaningful evaluations.

Table 4.5: Comparison of average (absolute) pose error (APE) and average delta pose error (ADPE) on validation sequences using the 2023 sensor setup. Each adjacent pair of columns compares APE (left number) with ADPE (right number). *E.g.* the first two data columns compare APE^{xy} with ADPE^{xy}. Sequences along the same route are averaged, being weighted according to their respective temporal length.

	APE / ADPE							
Sequence	xy (in cm)		φ (in °)		$\boldsymbol{t}$ (in cm)		$\boldsymbol{R}$ (in °)	
Adenauer 01/02	1.7	2.0	0.017	0.018	8.2	8.0	0.20	0.17
Moltke Big 01/02	1.6	2.2	0.018	0.020	6.2	6.6	0.16	0.15
∅	1.7	2.1	0.018	0.019	7.0	7.2	0.18	0.16
APE - ADPE	-0.4		-0.002		-0.1		0.016	

Table 4.6: Comparison of maximum (absolute) pose error (MPE) and maximum delta pose error (MDPE) on validation sequences using the 2023 sensor setup. Each adjacent pair of columns compares MPE (left number) with MDPE (right number). *E.g.* the first two data columns compare MPE^{xy} with MDPE^{xy}. Sequences along the same route are averaged, being weighted according to their respective temporal length.

	MPE / MDPE							
Sequence	xy (in cm)		φ (in °)		$\boldsymbol{t}$ (in cm)		$\boldsymbol{R}$ (in °)	
Adenauer 01/02	24.2	36.2	0.24	0.56	178.0	209.7	1.78	2.07
Moltke Big 01/02	24.9	44.0	0.40	0.36	117.7	175.8	1.97	3.79
∅	24.9	44.0	0.40	0.56	178.0	209.7	1.97	3.79
MPE - MDPE	-3.0		-0.027		-10.4		-0.175	

Localization in Up-to-date Maps

The most common task for localization approaches is to localize in a map that has been created or updated recently enough that no major changes have occurred. Now, in contrast to the previous experiment, the map has been created using sensor data recorded during a *different* drive than the one used for localization.

Table 4.7: Localization results in up-to-date maps. Reported are the average delta pose error (ADPE) and maximum delta pose error (MDPE) as well as the availability (av.) of localization results, indicated by the self-assessment of the approach, on validation sequences using the 2023 sensor setup.

	ADPE / MDPE								Av.
Sequence	xy (in cm)		φ (in °)		$\boldsymbol{t}$ (in cm)		$\boldsymbol{R}$ (in °)		(in %)
Adenauer 01/02	2.1	27.8	0.02	0.55	8.2	214.5	0.18	2.94	77.6
Moltke Big 01/02	2.2	31.9	0.02	0.40	7.0	213.0	0.16	4.67	85.6
∅	2.2	31.9	0.02	0.55	7.5	214.5	0.17	4.67	81.8

The numbers in Table 4.7 show that for localization in up-to-date maps, PCG can achieve exceptionally high accuracy on average. The error distribution is also depicted in more detail in Figure 4.11. Regarding the MDPE as conservative upper bound, the proposed approach can still always achieve lane-level accuracy. This indicates that especially 2D translational and yaw errors are usable for automated driving if any localization result is returned.

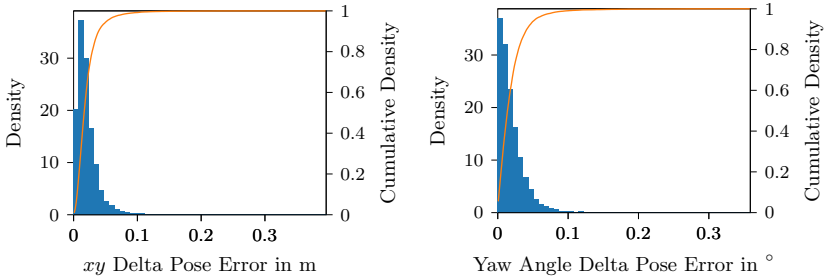


Figure 4.11: Density histogram (blue) and cumulative density (orange) of 2D translational (left) and yaw delta pose error (right) for localization in up-to-date maps, *i.e.* all four validation sequences with the 2023 sensor setup.

While average translational errors in 3D and overall rotational error are usable for any conceivable application, the comparatively large maximum errors show that they suffer in places where height and pitch angle cannot be resolved well. Using either a rear camera or additional features on the ground, like road borders and markings, is expected to alleviate this issue to a level of negligibility.

The average performance of the proposed method is significantly better than that of any similar localization approach the author is aware of. This is particularly noteworthy as PCG uses only a single frame of measurements and does not involve any filtering. The reported performance has been verified by various additional experiments including using relative GNSS measurements instead of KISS-ICP or computing ADPE over larger windows which all confirmed the findings.

One core design feature of the approach is a very conscious trade-off between accuracy and availability. The ability to self-assess localization performance makes it possible to rather avoid any output instead of producing a possibly erroneous result. This increases the safety of the overall system since it can knowingly extrapolate using *e.g.* wheel or lidar odometry. Hence, the while the reported availability does not seem sufficient at first glance, it is an artifact of actually enabling safer localization.

Moreover, it is to note that availability is not uniformly distributed. The plot in Figure 4.12 shows that segments with few landmarks, mostly straight roads with neither signage nor traffic lights, are most affected by outages. From an automated driving point of view, it is on those very segments where highly accurate localization is least important.

The results can be concluded with a short anecdote. Due to a slight deviation between the routes of sequences Moltke Big 01 and 02, there is a non-overlapping part w.r.t. the respective other sequences, which is omitted when computing metrics. However, this allows us to observe that the last estimated localization result before the deviation is so accurate, especially in orientation, that using dead reckoning via lidar odometry over more than $400\,\mathrm{m}$ leaves less than $40\,\mathrm{cm}$ error when both routes merge again. From this perspective, localization with PCG can be viewed as good contender even for highly available best effort localization approaches.

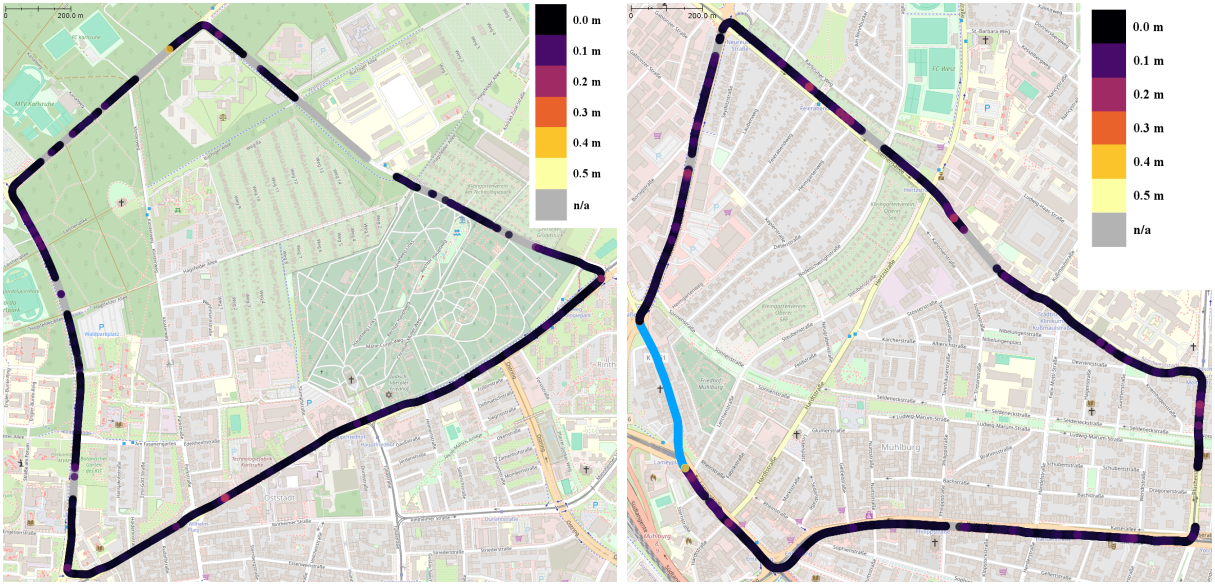


Figure 4.12: Localization results in up-to-date maps. Linearly color-coded xy translational error is depicted over aerial imagery on the validation sequences Adenauer 01 (left) and Moltke Big 01 (right) using the 2023 sensor setup. To highlight them, larger errors are painted over smaller errors. Gray parts indicate that the self-assessment failed and localization is not available, *e.g.* due to a lack of features. Due to a small deviation in the drives, blue parts have no counterpart map in the respective other drive.
Background image: © OpenStreetMap contributors

Localization in Outdated Maps

As examined in detail in Chapter 2, the real world changes over time, gradually rendering maps outdated. By making use of man-made semantic landmarks, the approach proposed in this thesis is designed and expected to be significantly more robust than visual or fine-grained structural features that age in weeks or months with *e.g.* changes in foliage or leaf coloration. However, even man-made infrastructure undergoes changes eventually. This poses the question how well localization performance is in aging maps.

For this, mapping drives from 2020 were used to create maps. Sequences from 2023 can then be used to localize therein. This allows testing the proposed localization method with real-world changes accumulated over about three years.

Table 4.8: Localization results in outdated maps. Reported are the average delta pose error (ADPE) and maximum delta pose error (MDPE) as well as the availability (av.) of localization results, indicated by the self-assessment of the approach. Validation sequences with the 2023 sensor setup are used to localize in maps from 2020.

	ADPE / MDPE								Av.
Sequence	xy (in cm)		φ (in °)		$\boldsymbol{t}$ (in cm)		$\boldsymbol{R}$ (in °)		(in %)
Adenauer 01/02	2.9	41.4	0.03	0.64	11.4	227.2	0.26	3.75	54.9
Moltke Big 01/02	2.9	56.6	0.03	0.62	9.5	224.7	0.21	2.69	66.6
∅	2.9	56.6	0.03	0.64	10.3	227.2	0.23	3.75	61.2

The numbers in Table 4.8 show that average and maximum error increase slightly, but are still excellent on average and able to achieve lane-level accuracy even in the worst case. Not unexpectedly, availability has decreased compared to up-to-date maps. While few changes are sufficient to render localization unavailable in areas with sparse landmarks, also some intersections and places with high landmark density have been reconstructed entirely.

Qualitative examples of successful localization despite changes are depicted in Figures 4.13 and 4.14. An overview is provided in Figure 4.15.

Figure 4.13: Qualitative examples of a successful localization in a severely outdated HAD map. Associations are depicted in blue (detection) and green (landmark). Unassociated detections are white and undetected/outdated landmarks are orange.

Figure 4.14: Qualitative examples of a successful localization in a severely outdated HAD map. Associations are depicted in blue (detection) and green (landmark). Unassociated detections are white and undetected/outdated landmarks are orange.

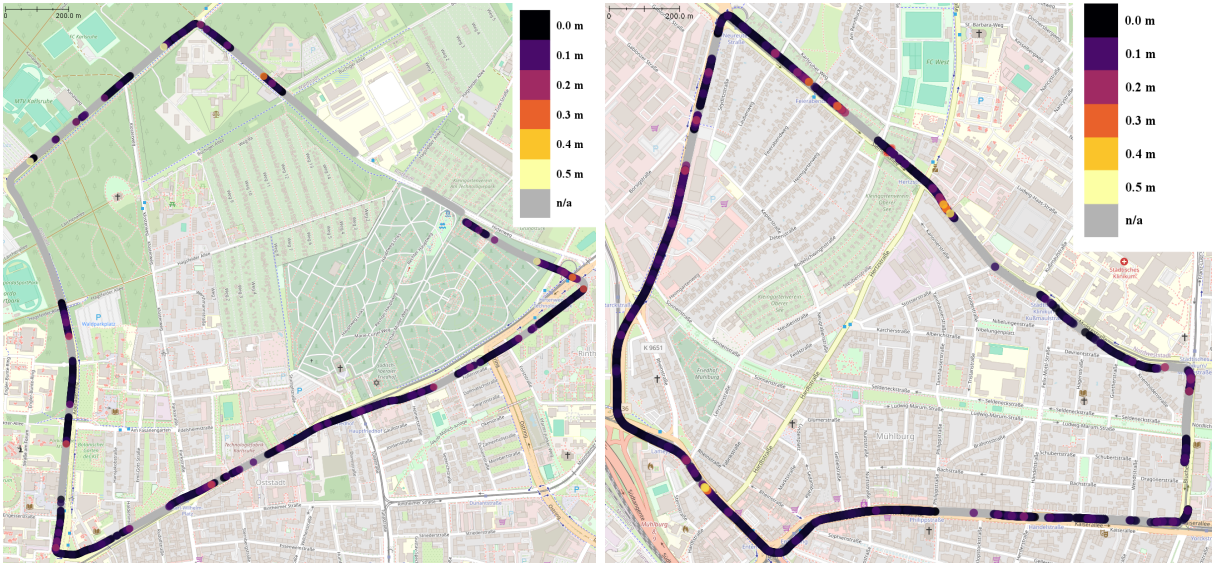


Figure 4.15: Localization results in outdated maps. Linearly color-coded xy translational error is depicted over aerial imagery on the validation sequences Adenauer 01 (left) and Moltke Big 02 (right) using the 2023 sensor setup. To highlight them, larger errors are painted over smaller errors. Gray parts indicate that the self-assessment failed and localization is not available, *e.g.* due to a lack of features. The visible alignment issues w.r.t. the background image are due to odometry drift occurred during mapping. While it could be compensated, doing so was not deemed necessary as the error is within the required position prior for the approach to succeed. Background image: © OpenStreetMap contributors

Timing Analysis

Besides accuracy and availability, real-time solutions are an important factor to enable deployment in automated vehicles. To ensure a limited latency, PCG's solution retrieval has been abandoned after 100 ms as described in Section 4.2.9. Figure 4.16 depicts the computation times of the approach measured on an AMD EPYC 7702P 64-core processor as it is used in the measurement vehicle at MRT. Note that the many cores may be deceptive. Due to the single-threaded clique solver, they only accelerate the marginalization.

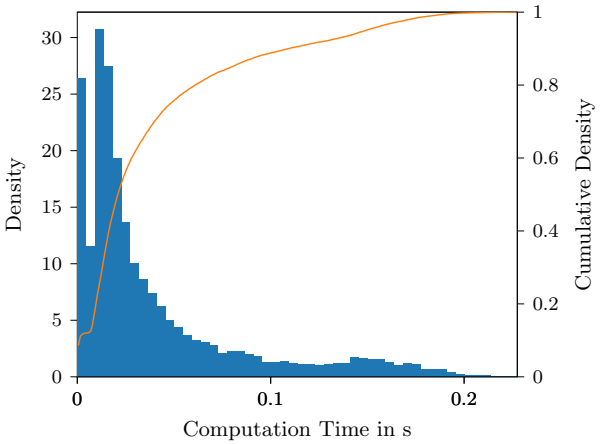


Figure 4.16: Density histogram (blue) and cumulative density (orange) of computation times for localization in up-to-date maps, *i.e.* all four validation sequences with the 2023 sensor setup.

The timing histogram can be explained by three behaviors. In case of really few or badly detected landmarks, PCG quits without result, leading to the first peak close to zero. The major peak is due to normal operation, *i.e.* all solutions within the likelihood window are retrieved in less than the time limit. Due to marginalization, the overall latency may exceed the retrieval timeout of 100 ms. This is also what explains the rather bold tail between 100 ms and 200 ms:

PCG has been stopped early, but marginalization, especially solving the robust non-linear optimization problems in ceres, still takes an additional $100\,\mathrm{ms}$.

Storage Requirements

When large-scale maps are stored onboard automated vehicles or map updates need to be transmitted wirelessly, the storage requirements of a localization approach can become an interesting aspect that has already been discussed in Section 4.1.3. To evaluate minimal storage requirements, the maps were stored in binary form and compressed using a standard Lempel-Ziv-Markov chain algorithm (LZMA) compression contained in the Linux `xz` tool. Results are reported in Table 4.9.

Table 4.9: Storage requirements of the parametric maps used for localization.

Year	Sequence	Length (in km)	Map Size (in KiB)	Relative Size (in KiB per km)
2020	Adenauer 01	5.13	97.3	19.0
2020	Moltke Big 01	4.95	129.5	26.2
2023	Adenauer 01	5.23	102.1	19.5
2023	Adenauer 02	5.22	101.4	19.4
2023	Moltke Big 01	4.87	128.4	26.4
2023	Moltke Big 02	4.97	131.5	26.5
	∅			22.7

Previous sections showed that parametric maps proposed in this work enable excellent localization accuracy on par with conventional sensor-specific or even deeply learned approaches. At the same time, with an average map size of $22.7\,\mathrm{KiB}$ per km, parametric maps are orders of magnitude more compact. This offers state-of-the-art localization performance in large-scale onboard maps as well as efficient wireless map updates.

4.6 Limitations and Discussion

While PCG, the proposed approach for data association and localization, is able to produce strong results, it is still important to point out its limitations.

First, probabilities in transformation invariant correspondence space are not identical to those probabilities given a transformation, but merely highly correlated. While this has been addressed by retrieving multiple solutions within a sufficiently large and scalable probability window, it hinges on a non-degenerate distribution of feature points. However, the author is not aware of any alternative independent from transformations and is pessimistic that any such exists.

Another limitation is the scalability of the clique solver. While solution times are sufficient for the presented problems, they become infeasible for significantly large or denser ones. This is due to the goal of optimality which requires an exhaustive search of either transformation or association space. Increasing semantic resolution, *e.g.* more fine-grained traffic light or sign classes, or object parameters, *e.g.* by including the orientation of traffic lights, would make the association problem sparser and, hence, improve not only solution quality but especially retrieval times.

In this thesis, only features with a spatially limited extent have been used for data association and an extension to locally unlimited features, such as curbs or solid road markings, might seem unclear. A possible solution has been proposed for existing pairwise compatibility data association approaches in two previous publications with the participation of the author [MPH+22a, MPH+22b]. It discretizes line features but considers the effects of discretization, *i.e.* higher uncertainty along than across the line.

Finally, with only one camera and only the proposed three classes of map features, there is a significant trade-off between availability and localization accuracy, especially when localizing in outdated maps. This can be improved in three ways. First, more semantic classes, especially road markings, can increase the amount of available features significantly. Using the same classes, a similar effect can be achieved by making use of a rear camera, which is also expected to lower errors in pitch angle and height significantly, or even all surround cameras

proposed in the 2023 sensor setup. Finally, if neither is desired or feasible, the mapping method proposed in Chapter 3 could be used to build a *local* map across few a frames to collect features until a unique localization is possible.

4.7 Conclusion and Outlook

In this chapter, the coupled data association and localization problem has been approached under very challenging preconditions, *i.e.* no prior pose or filtering, ambiguity awareness, fully probabilistic modeling, real-time capability, and global optimality. At the same time, relatively sparse and partially outdated landmarks were supposed to be associated to noisy detections that may contain clutter.

The core idea that enables to solve data association despite these difficulties is a novel probabilistic formulation of the problem in a transformation invariant space. Combining existing ideas of individual and pairwise compatibility with probabilistic formulations inspired by random finite set multi object tracking allows probabilistically globally optimal data association to be formulated as totally weighted maximum clique problem.

Modifying a state-of-the-art clique solver to support totally weighted cliques with real-valued weights allows retrieving not only one best solution, but all assignments in a probability window in real time. Using them to compute a marginalized pose with uncertainties in position and orientation makes it possible to achieve ambiguity-aware localization results from a single frame of measurements. The only requirement is a very coarse initial position with decameter accuracy to restrict the number of possibly observable landmarks.

A novel 2D probability distribution, called joint Euclidean angular distance difference (JEADD), has been proposed to improve the association of oriented objects. In simulation, it clearly outperforms affine Grassmannian distances, the current state of the art.

For the association of unknown correspondences as well as for mere outlier rejection, the proposed PCG is superior to existing data association approaches.

On the KITTI dataset, PCG represents the new state of the art for point cloud registration using both conventional and learned feature point descriptors.

While those benchmarks enable comparison with previous approaches on neutral ground, PCG has actually been invented to safely and verifiably solve the coupled data association and localization problem in sparse semantic HAD maps. With an average planar translation error of about $2\,\mathrm{cm}$ and a yaw angle error of $0.02°$, localization in up-to-date maps is extremely precise and touches the boundaries of available ground truth. As ambiguous associations are rejected and there are not enough landmarks available everywhere, overall availability is at about $80\,\%$. However, it is consistently high in or before intersections where highly accurate localization is particularly important to use the map for interpreting the complex static environments.

While availability decreases in outdated or artificially changed maps, the self-assessment still allows achieving an almost unimpaired average localization error. Even in the worst cases, the maximal error still enables lane-level accuracy.

In the next chapter, the data association results will be used to power the continuous verification of HAD maps via a marginalization over assignments.

5 Continuous Verification of HD Maps using Ternary Evidences

With highly accurate detections and a suitable data association and localization approach, finally, the actual verification can be tackled. When verifying an HAD map, there are several goals to achieve simultaneously.

First, the map should be verified sufficiently far *ahead* of the automated vehicle since this enables to react on a change, *e.g.* by adapting the driving behavior or exit the automated mode. Assuming urban velocities of at most $70\,\frac{\mathrm{km}}{\mathrm{h}}$ and comfortable breaking deceleration of $3\,\frac{\mathrm{m}}{\mathrm{s}^2}$ [BB15], a desired verification distance of $50\,\mathrm{m}$ can be considered a conservative value. Implicitly, this goal induces the requirement of fast computation times.

The second aspect is the level of detail. Verification and change detection on the level of whole roads or road sections is too conservative. The more details are contained in a map, the higher the probability that at least one element along the route might have changed. Disabling all automated functions which somehow rely on the map would not help scaling automated driving or increase its acceptance. Instead, verification and change detection should happen as fine-grained as possible, optimally on the level of individual map elements. This allows the automated vehicle to adapt or degrade only those very driving functions that rely on the changed detail, *e.g.* by choosing a lane with verified borders and traffic rules.

Next, verification and change detection are not to be formulated as complementary possibilities. Many previous approaches specialize on detecting deviations between map elements and measurements. While this is very relevant, it is only half of the battle. Due to occlusions or limited sensor range, it might have been impossible to detect a change from the vehicle's point of view. Hence, the

absence of changes must never be interpreted as verification. This lead to the proposal of a ternary belief system [JKS18], which separates the belief masses for change, verification, and the unknown and is adapted for this thesis.

Finally, the verification needs to work with extremely high reliability. While falsely detected changes may lead to partially or temporarily unavailable maps, a falsely verified map element might have catastrophic results. For instance, falsely confirming a removed priority traffic sign at an intersection where the vehicle now is required to yield can lead to a fatal accident with the automated vehicle obviously being at fault.

Hence, the challenge is to verify individual physical map elements in real time and at high range, but with exceptional reliability. This chapter will show how the contributions of the previous two chapters can be used to achieve all these goals. Detecting changes optimally will be no explicit goal, but is only necessary as counterbalance for spurious verification evidence.

As a conceptual outlook, it is described how an appropriately designed map framework, such as Lanelet2 [PPJ+18], allows the transfer of verification results from the physical to more abstract layers. This makes it possible to verify not only landmarks for localization, but also the particularly interesting semantic map information, like traffic rules or traffic light to lane assignments, *without* inferring those relationships online.

Contributions

The first contribution is to **formulate verification of physical map elements as marginalization over association hypotheses** computed for localization. Combined with the contributions from the previous two chapters, this enables the verification of traffic lights and road signs with sufficient certainty at distances that enable to trust the verified map elements for safe and comfortable automated driving.

To **actively detect the absence of map elements** a fast lidar visibility check via ray casting is proposed. For change detection, lidar visibility information is fed into a random forest classifier.

Non-complementary beliefs in changes and verification are reflected by **adapting a ternary evidence system**. It allows the decoupling of both belief masses which is necessary due to limited fields of view and occlusions where neither verification can succeed nor changes can be detected. Combination and aggregation over time are implemented using Dempster-Shafer evidence theory which allows the system to output probabilistically meaningful results to other driving functions.

Evaluation shows that traffic lights and road signs can be **verified with exceptional certainty at ranges significantly beyond** $50\,\mathrm{m}$. At the same time, for more than 1000 simulated changes, **not a single changed map element was falsely verified**.

Finally, using a relational bottom-up map design, it is proposed how to **propagate verification results from physical map elements to abstract map content**, such as traffic rules.

Previous Publications

The evidential verification has first been explored in a student thesis [Fan21] supervised by the author. In a workshop paper [PS22], the essential verification concept was already published. With Lanelet2 [PPJ+18], a suitable map design for the propagation of verification results from physical to abstract layers has been published previously.

5.1 Related Work

As foundation, this chapter builds upon the *theory of belief functions*, also known as *evidence theory* or Dempster-Shafer theory introduced by Dempster [Dem67] and expanded by Shafer [Sha76].

Previous approaches that tackled outdated maps as well as their detection and verification can be subdivided into pure change detection approaches, methods that are capable of some kind of verification, and other works.

For digital map providers, the detection of changes, map verification, and map updates are ancient topics. Corresponding research in the field of photogrammetry reaches back to the last millennium. Similarly, in indoor robotics where semi-static objects are far more common, changing localization maps have been tackled manifold. Due to the gap in sensors and resulting methods, related work presented in this thesis only focuses on change detection in the context of automated driving and ADAS.

It has been noted early on that any task which compares map and sensor data is tightly coupled with the problem of localizing in the map [DJM+93]. Hence, most approaches simply assume a highly accurate localization although achieving it is particularly challenging when the map is outdated. The method proposed in this thesis is based on a localization approach that is particularly tailored to deal with outdated maps. If the localization's self-assessment fails, any verification is omitted. This helps to avoid falsely verifying the map due to a wrong localization result.

An alternative is to not assume a highly accurate localization. For instance, to detect changes in highly ambiguous man-made environments such as highways, the author of this thesis proposed a number of approaches to detect changes by comparing periodicity invariant map features [PSH+20a, PSH+20b, PSH+21]. Ha et al. [HOL+23] predict a lane graph and image semantics that can be compared with the map graph and rendered semantics using graph and semantic similarity metrics, respectively. While image semantics crucially rely on pixel accurate 6D localization, inference and comparison on a structural or topological level is a robust yet promising idea.

5.1.1 Map Change Detection

Using GNSS traces of vehicles is sufficient to detect severe enough changes that would render SD maps outdated [ZBI12b, ZBI12a, ZBI16]. Since at least some traffic rules and intersection parameters can be inferred from floating car data [RBP+17], so can their changes be detected [EME+20]. To detect changed map elements in HD or HAD maps, like signs or land boundaries, however, sensors that perceive the environment are necessary.

Since some approaches refer to aggregated point clouds as HD maps, there are respective change detection and update approaches [KCS+21, GCL24]. For this thesis, such simple point clouds are not even remotely covered by the definition of semantic HD maps (*cf*. Definition 2.1).

To detect changes in a semantic HD map, there are two main categories of approaches. The first, here called *passive* change detection, is detecting landmarks and comparing detections with the map. To distinguish changes from occluded and consequently undetected map elements, this either requires to incorporate detectability or collect sufficiently many data that it is no issue.

The second kind is to *actively* infer changes by comparing raw sensor data and map content. While this typically requires a highly accurate localization, it can directly include occlusion handling and use the potentially outdated map element as input for *e.g*. deep learning methods. In addition, active change detection can be designed to be orthogonal to the landmark detections which in this work will be used to verify map elements. This will be important when tracking change and verification beliefs independently. Hence, in this thesis, an active change detection approach based on lidar ray casting is proposed.

Passive Change Detection

Processing floating car data with two particle filters for localization allows Pannen et al. [PLB19] to detect map changes in the backend. Deriving metrics based on the localization results and using them in a boosted classifier makes it possible to determine how well the map matches with road marking observations. Combining the results of multiple drives improves change detection performance significantly. The approach was later extended to detect topological changes and create map updates [PLH+20].

Detecting changes *ahead* of the vehicle allows adapting driving behavior in automated modes or exiting them if necessary. Early works focused on the detection of special changes, *e.g*. newly constructed roundabouts [RB14, RB15].

Berrio et al. [BWS+22] recently published a map maintenance framework to which the reader is referred for similar related work. Like the approach proposed

in this work, they use a lidar ray casting to determine the detectability of map features, but reduce visibility information to a mere BEV grid.

Active Change Detection

There are at least two ways to actively detect changes. The first, previously used in robotics [UGB+13] or for point clouds [KCS+21], is using measurement principles like lidar ray casting or depth estimation.

The second one are deep learning approaches that directly compare map content and raw sensor data. They range from deep metric learning [HKK20] over Siamese networks [DPS20] to conventional networks with temporal fusion [HJL+22]. Particularly worth mentioning is the work of Lambert and Hays [LH21] who not only compared various approaches for change detection, but also published the first publicly available dataset with raw sensor data and sufficiently many changes to train DNNs.

Using such a deep learning approach as change detector is expected to easily surpass the performance of the approach proposed in this thesis, which is mostly seen as proof of concept. However, as the evaluation will show, using physical measurement principles with a conventional classifier can still lead to surprisingly strong results.

5.1.2 Map Verification

Map verification goes beyond the mere detection of changes since the absence of changes does not necessarily allow to deduce a map's verification. Verifying SD maps ahead of the vehicle can be achieved by perceiving the road course, *e.g.* using camera sensors [HGS+14b, HGS+14a].

Assuming a perfect localization, Raaijmakers [Raa17] proposed to match detections with map elements in parameter space. Other approaches verify the map in a grid-based representation, *e.g.* by comparing shape and semantics of the map with sensor fusion output [BVR+20].

Plachetka et al. [PSF+23] proposed three approaches, ranging from a conventional comparison of detections and map elements to a deep learning framework that has the map and virtual lidar scans as input to predict either change or verification. In terms of properly evaluated performance, it can be considered the state of the art for map verification. But, its use of virtual, *i.e.* subsequently aggregated, lidar scans makes it debatable to directly transfer its results to applications onboard a vehicle.

Already mentioned as idea by Raaijmakers [Raa17], coupling the localization and verification problem has been proposed by Jo et al. [JKS18]. They model the matching of detections and map elements in a ternary formulation using Dempster-Shafer theory, elegantly incorporating detectability in theory. In practice, they only model it with a coarse field of view. Unfortunately, Jo et al. do not report any evaluation results that demonstrate more than the basic viability of the approach. Still, their theoretical model is adapted for the ternary evidential verification used in this thesis.

5.1.3 Other Works

There are a few other closely related works around the field of map verification. One example is the fusion of multiple hypotheses for the road layout [DRS+15, NSU+16, NSV+18], including knowledge from prior maps. Jomrich [Jom20] comprehensively covered the issue of how to handle map updates and Maierhofer et al. [MBA23] proposed a method for formal map verification and repairing. For similar works, the reader is referred to the respective related work sections.

5.2 Ternary Evidential Verification

When maps are supposed to be verified or, on the contrary, changes should be detected, it is crucial to understand that it is not a binary problem. Of course, given global world knowledge, a map element can only either be changed or verified. However, having only data from onboard sensors, occlusions and limited sensor range may prevent both the recognition of map elements and

the detection of changes. As proposed by Jo et al. [JKS18], this motivates a *ternary* method that tracks the beliefs for verification and change independently over time using evidence theory.

5.2.1 Concept

While the possibilities of change and verification are mutually exclusive, they are not complementary, *i.e.* the sum of their beliefs might be less than one. Additionally, their measurements are independent from each other and, hence, might be conflicting. That means, for the very same frame of measurements, it could happen that a map element can be verified but the change detector also detects a change . As an information fusion concept that covers both incomplete evidences and conflicting measurements, the evidence theory proposed by Dempster and Shafer is commonly used.

In evidence theory, the mutually exclusive possibilities span the so-called *frame of discernment* $\mathbf{\Omega}$. The set of each combination of possibilities as well as the empty set $\emptyset$, formally called the power set $2^{\mathbf{\Omega}}$, can then be assigned a belief using a mass function m. This work assumes mathematically well-behaving *basic belief assignments (BBAs)*, which fulfill the conditions

$$m(\emptyset) = 0, \tag{5.1}$$

$$\sum_{\omega \in 2^{\mathbf{\Omega}}} m(\omega) = 1. \tag{5.2}$$

The set of possibilities is that a map element $\boldsymbol{\ell}$ has changed or not and, hence, could be verified

$$\mathbf{\Omega} = \left\{\omega_{\mathrm{ch}}^{\boldsymbol{\ell}}, \omega_{\mathrm{ver}}^{\boldsymbol{\ell}}\right\}. \tag{5.3}$$

In addition, in evidence theory, there are two more formal possibilities, $\emptyset$ and the trivial element $\mathbf{\Omega} = \left\{\omega_{\mathrm{ch}}^{\boldsymbol{\ell}}, \omega_{\mathrm{ver}}^{\boldsymbol{\ell}}\right\}$. While the first is usually not assigned any belief mass, the latter captures the unknown or uncertainty in the decision between the concrete possibilities $\omega_{\mathrm{ch}}^{\boldsymbol{\ell}}$ and $\omega_{\mathrm{ver}}^{\boldsymbol{\ell}}$.

In order to collect evidence for the possibility $\omega^{\boldsymbol{\ell}}_{\mathrm{ch}}$, one can distinguish two kinds of methods, here called active and passive. As discussed in related work, many approaches for change detection use the absence of a suitable detection as evidence for changes. This not only blends change detection and verification, but also requires reasoning about the detectability of landmarks in order to avoid false positives.

In contrast, active methods directly measure the absence of a landmark. This can happen through physical measurement principles, like lidar ray casting or visual depth estimation. But also training a change detector based on map and sensor data can be considered an active method as it is expected to learn how the absence of map elements manifests in sensor data.

In this work, a change detector is trained using lidar visibility information. First the lidar ray casting and then the change detector are presented in the next two sections. Evidence for $\omega^{\boldsymbol{\ell}}_{\mathrm{ver}}$ can be collected by marginalization over those association hypotheses which include an assignment between any detection and $\boldsymbol{\ell}$. This is described in the section after. An overview is depicted in Figure 5.1.

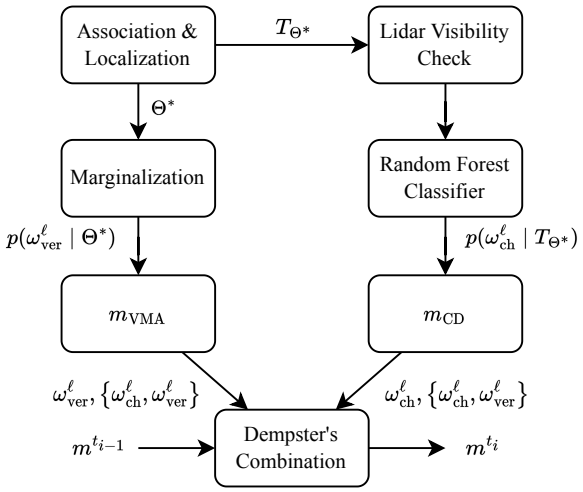


Figure 5.1: Overview of the evidential map verification approach.

5.2.2 Lidar Visibility via Ray Casting in Range Images

In order to actively collect evidence of changes, a lidar ray casting is performed. It is inspired by previous approaches for change detection in robotics [UGB+13].

Most lidar ray casting solutions use an *un*ordered 3D point cloud generated from raw measurements and project it in a spherical coordinate system. This is not only slow, but also makes it hard to distinguish no-return from missing measurements since in the point cloud no-returns are omitted. At the same time, modern lidars have irregular firing angles, *e.g.* due to slight variations in rotation rate or due to dynamic firing pauses, which were introduced to mitigate cross-sensor interference. Hence, when projected into a range image, it is unclear whether a certain pixel had no return within measurement range or was in fact never measured.

A customized processing chain for the Velodyne lidar sensor, jointly developed by Sven Richter and the author of this thesis, stores the same point clouds as ordered range images with spherical coordinates and equiangular columns directly from raw measurements, so-called packets. This has two advantages. First, it can encode no-returns and unmeasured pixels differently, preserving this information. At the same time, it enables extremely fast 2D ray casting in spherical coordinates.

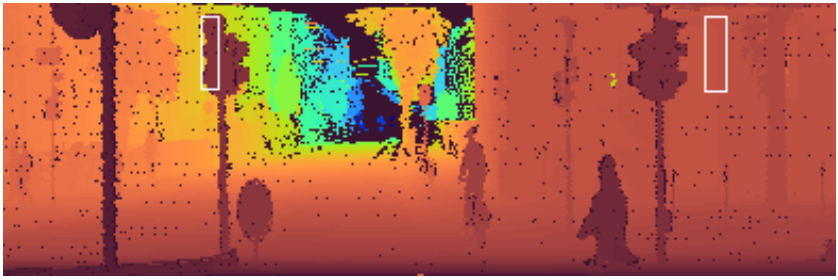


Figure 5.2: Example crop of the ordered range image used to measure visibility by ray casting corresponding to the upper camera image in Figure 5.6. The white exemplary mask borders depict projections of unchanged (left) and changed (right) traffic lights into the range image. The color of all other pixels encodes distance of corresponding lidar returns. Black pixels contain no return.

Using the pose T_{Θ^*} estimated in the localization step, each map element in question is projected into the range image as depicted in Figure 5.2. The parametric representations proposed in Chapter 3 can be used to derive a minimum and maximum distance, $d_{\min}(\boldsymbol{\ell})$ and $d_{\max}(\boldsymbol{\ell})$, for any part of the map element.

Range images with spherical coordinates make it trivial to compute a mask which crops the relevant part of lidar measurements that could potentially hit the map element, called $\mathfrak{L}_{\boldsymbol{\ell}}$. Each return has one of three possibilities: It can be closer than $d_{\min}(\boldsymbol{\ell})$, hinting towards an occlusion of $\boldsymbol{\ell}$, farther than $d_{\max}(\boldsymbol{\ell})$, hinting towards $\boldsymbol{\ell}$'s absence, or in between, implying a hit on $\boldsymbol{\ell}$. Hence, for each map element $\boldsymbol{\ell}$, three return sets can be computed based on the masked lidar points $\mathfrak{L}_{\boldsymbol{\ell}}$:

$$\mathfrak{L}_{\boldsymbol{\ell}}^{-} = \{\mathfrak{l} \in \mathfrak{L}_{\boldsymbol{\ell}} : d(\mathfrak{l}) < d_{\min}(\boldsymbol{\ell})\} \tag{5.4}$$

$$\mathfrak{L}_{\boldsymbol{\ell}}^{+} = \{\mathfrak{l} \in \mathfrak{L}_{\boldsymbol{\ell}} : d(\mathfrak{l}) > d_{\max}(\boldsymbol{\ell})\} \tag{5.5}$$

$$\mathfrak{L}_{\boldsymbol{\ell}}^{0} = \{\mathfrak{l} \in \mathfrak{L}_{\boldsymbol{\ell}} : d_{\min}(\boldsymbol{\ell}) \leq d(\mathfrak{l}) \leq d_{\max}(\boldsymbol{\ell})\}. \tag{5.6}$$

In general, no-returns can be caused by the absence of obstacles within sensing range, but also by poorly reflecting objects. Due to the Velodyne Alpha Prime's strong performance even at low reflectivity, no-returns fully contribute to $\mathfrak{L}_{\boldsymbol{\ell}}^{+}$.

The whole visibility check takes only few microseconds per landmark. It relies on the customized processing chain for Velodyne raw messages which is however significantly faster than the default processing chain. Hence, the additional latency of the procedure can be considered in between negative and negligible.

5.2.3 Change Classifier

As described in the related work section, a number of works proposed a variety of methods to detect map changes, ranging from simple classifiers to deep learning methods. Since the development of an optimal change classifier might fill a dissertation on its own, in this work, easily available methods in OpenCV [Bra00], *i.e.* k nearest neighbors, boosted decision trees, random forests, SVMs, and a naive Bayes classifier, were compared. Boosted trees and

random forests turned out to perform equally best and the latter was chosen as proof of concept change classifier for this thesis.

It is trained and evaluated using 11D feature vectors. Using the lidar visibility test, the number and share of each of the three return classes, $\mathfrak{L}_{\ell}^{-}$, $\mathfrak{L}_{\ell}^{0}$, and $\mathfrak{L}_{\ell}^{+}$, comprise six feature dimensions. In addition, the total number of lidar returns, $\mathfrak{L}_{\ell}$, and the respective shares of points on and through over their sum, *i.e.* $\mathfrak{L}_{\ell}^{0}/\mathfrak{L}_{\ell}^{0} + \mathfrak{L}_{\ell}^{+}$ and $\mathfrak{L}_{\ell}^{+}/\mathfrak{L}_{\ell}^{0} + \mathfrak{L}_{\ell}^{+}$, are used. The remaining two dimensions are distance of the landmark to the sensor, $d(\ell)$, and the size of the landmark's backprojection in pixels. Evaluating and refining the feature selection is an open issue since the change detector merely serves as proof of concept. When considering this, training a lean change detection DNN directly on camera and range images might be more promising.

Using the simulated changes which will be described in Section 5.3.1 as ground truth, the training sequences Ostring 01/02 and Moltke Small 01 can be used to generate training data. This yields around 300k samples for poles, 416k samples for traffic lights, and 129k samples for road signs.

Results on the validation sequences Adenauer 01/02 and Moltke Big 01/02 are reported in Table 5.1. While being merely intended as a proof-of-concept classifier that can be used to demonstrate the concept of ternary evidential verification, its performance actually turned out to be in the same ballpark as state-of-the-art deep learning approaches [PSF+23].

Table 5.1: Performance of the random forest classifiers used to detect changes. All values are averaged across the four validation sequences and stated in %.

Type	Precision	Recall	F_1 Score	Accuracy
Poles	74.2	70.2	72.1	94.9
Traffic Lights	85.2	83.2	84.2	96.4
Traffic Signs	87.8	88.4	88.1	97.7

The basic belief assignment (BBA) for the change detector, m_{CD}, can be formulated as

$$m_{\mathrm{CD}}(\emptyset) = 0 \tag{5.7}$$
$$m_{\mathrm{CD}}\big(\omega_{\mathrm{ch}}^{\boldsymbol{\ell}}\big) = \beta_{\mathrm{CD}}^{\mathfrak{c}}(d(\boldsymbol{\ell}))\, p\big(\omega_{\mathrm{ch}}^{\boldsymbol{\ell}} \mid T_{\Theta^*}\big) \tag{5.8}$$
$$m_{\mathrm{CD}}\big(\omega_{\mathrm{ver}}^{\boldsymbol{\ell}}\big) = 0 \tag{5.9}$$
$$m_{\mathrm{CD}}\big(\big\{\omega_{\mathrm{ch}}^{\boldsymbol{\ell}}, \omega_{\mathrm{ver}}^{\boldsymbol{\ell}}\big\}\big) = 1 - m_{\mathrm{CD}}(\omega_{\mathrm{ch}}^{\boldsymbol{\ell}}). \tag{5.10}$$

The prefactor $\beta_{\mathrm{CD}}^{\mathfrak{c}}(d(\boldsymbol{\ell})) \in [0,1]$ lowers the evidence mass linearly with increasing distance. Its exact slope and operating range are class dependent hyperparameters. The probability of a change, $p\big(\omega_{\mathrm{ch}}^{\boldsymbol{\ell}} \mid T_{\Theta^*}\big)$, is simply the binary classifier output.

5.2.4 Verification by Marginalized Association

The basic idea for the verification of a map element $\boldsymbol{\ell}$ is that if a semantically identical object is measured sufficiently close in parameter space, the map element is identical w.r.t. Definition 2.2. Whether such a matching detection exists can be determined using the data association and localization approach proposed in Chapter 4.

It outputs multiple hypotheses $\theta \in \Theta^*$ each of which contains a probability for $\boldsymbol{\ell}$ being detected and associated, which is equated with $\boldsymbol{\ell}$ being unchanged. This makes it possible to marginalize the probability of $\boldsymbol{\ell}$ being unchanged over all hypotheses, similar to the pose marginalization presented in Section 4.4.4. Since not all, but only the best hypotheses Θ^* are used, this leads to the approximation

$$p\big(\omega_{\mathrm{ver}}^{\boldsymbol{\ell}}\big) \approx \frac{1}{\sum_{\theta\in\Theta^*} f_\theta(\theta)} \sum_{\theta\in\Theta^*} f_\theta(\theta)\mathbf{1}_{\theta(\mathcal{D})}(\boldsymbol{\ell}). \tag{5.11}$$

The indicator function $\mathbf{1}_{\theta(\mathcal{D})}(\boldsymbol{\ell})$ denotes that landmark $\boldsymbol{\ell}$ was associated by θ with any detection $\boldsymbol{d} \in \mathcal{D}$. Assignments not contained in Θ^* are assumed negligible due to their insignificant probability mass.

The BBA of this verification by marginalized association, m_{VMA}, then reads

$$m_{\mathrm{VMA}}(\emptyset) = 0 \tag{5.12}$$

$$m_{\mathrm{VMA}}\big(\omega_{\mathrm{ch}}^{\boldsymbol{\ell}}\big) = 0 \tag{5.13}$$

$$m_{\mathrm{VMA}}\big(\omega_{\mathrm{ver}}^{\boldsymbol{\ell}}\big) = \begin{cases} \beta_{\mathrm{VMA}}^{\mathfrak{c}}\, p\big(\omega_{\mathrm{ver}}^{\boldsymbol{\ell}}\big) & \text{if } p\big(\omega_{\mathrm{ver}}^{\boldsymbol{\ell}}\big) > \tau_{\mathrm{VMA}}^{\mathfrak{c}} \\ 0 & \text{else} \end{cases} \tag{5.14}$$

$$m_{\mathrm{VMA}}\big(\big\{\omega_{\mathrm{ch}}^{\boldsymbol{\ell}}, \omega_{\mathrm{ver}}^{\boldsymbol{\ell}}\big\}\big) = 1 - m_{\mathrm{VMA}}(\omega_{\mathrm{ver}}^{\boldsymbol{\ell}}). \tag{5.15}$$

Since this should rather underestimate than overestimate that $\boldsymbol{\ell}$ is unchanged, a threshold $\tau_{\mathrm{VMA}}^{\mathfrak{c}}$ is used. The prefactor β_{VMA} compensates false positive associations by discounting the evidence mass.

5.2.5 Evidential Combination

Both BBAs can now be combined as well as aggregated over time, *i.e.* combined with the BBA of the previous time frame, m^{i-1}, using *Dempster's rule of combination*, denoted by $\otimes$

$$m^i(\omega) = m_{\mathrm{CD}}^i(\omega) \otimes m_{\mathrm{VMA}}^i(\omega) \otimes m^{i-1}(\omega), \quad \omega \in 2^{\boldsymbol{\Omega}}. \tag{5.16}$$

Initially, *i.e.* when a landmark comes in sensing range, all belief mass is assigned to the trivial possibility

$$m^0\big(\big\{\omega_{\mathrm{ch}}^{\boldsymbol{\ell}}, \omega_{\mathrm{ver}}^{\boldsymbol{\ell}}\big\}\big) = 1. \tag{5.17}$$

Dempster's rule of combination has the assumption that the sources of evidence are independent. For the change detection and verification by association, this is not obvious, since both rely on lidar measurements. However, due to the random forest's comprehensible nature, one can examine the importance of individual features for classifying a landmark as changed. In fact, the points on it, $\mathfrak{L}_{\boldsymbol{\ell}}^0$, have only little influence. The returns before the landmark, $\mathfrak{L}_{\boldsymbol{\ell}}^-$, and especially the rays through it, $\mathfrak{L}_{\boldsymbol{\ell}}^+$, however are significant for classification performance.

This information is orthogonal to the distribution of lidar points on the landmark, which is used to derive the parametric detections used for association. Since this is the only evidence source that a landmark could be verified, it can be argued that the evidence sources within each time frame are not totally, but largely independent.

Regarding the aggregation over time, it is important to recall that each localization and association result is derived independently using a single frame of measurements. Also the change detection only uses sensor data from the current time step. This makes the evidence sources as uncorrelated over time as possible given inherent correlations in raw sensor data, *e.g.* due to similar points of view.

To evaluate the aggregated belief masses, in evidence theory, the probability of a possibility ω can be conservatively underestimated by the so-called *belief*. It can be computed by adding the belief mass allocated to all subsets of the possibility in question

$$\mathrm{bel}(\omega) = \sum_{\acute{\omega}:\acute{\omega}\subseteq\omega} m(\acute{\omega}) < p(\omega). \tag{5.18}$$

Since there are only two interesting possibilities, that a landmark has changed, ω_{ch}, and that it could be verified, ω_{ver}, belief computation is trivial.

The belief of those two possibilities can now finally be output and used by driving functions to estimate how reliable or outdated certain physical map elements are. An outlook how to propagate these results to abstract layers will be presented in Section 5.4.

5.3 Evaluation

To evaluate the proposed approach, ideally, one would need two sets of recorded sequences months or years apart. In addition, the corresponding map would exist in two versions with unchanged map elements having identical identifiers. This would allow to measure true changes in the world and use them as ground truth.

Unfortunately, the creation of a single ground truth map is estimated to take multiple days if not weeks and, hence, was not considered an option. At the same time, while the data association approach proposed in Chapter 4 would be an excellent tool for the task, the accurate co-referencing of maps across years and changes is still a research topic on its own.

Instead, one can take an automatically created map and simulate changes by changing parameters of map elements. This not only serves as pseudo ground truth, but also allows creating far more change examples than typically observed across years on the same section.

5.3.1 Spatially Consistent Change Simulation

Changes of individual map elements, *e.g.* a single sign mounted on a pole, happens occasionally. However, since all other unchanged map elements serve as spatially consistent reference, such changes are easy to detect. The challenging case are almost identical replacements of traffic light or sign constellations at another position. In fact, even in the comparatively humble collection of sequences, this was observed a couple of times. Hence, to make the approach robust against those difficult changes, changes are simulated consistently for small groups of map elements.

The basis for a change simulation is a map and parameters about the changes, *i.e.* a change ratio, minimal and maximal translational magnitude, and a selection radius to determine groups of landmarks. To simulate a change, an unchanged map element is picked at random. All unchanged elements within the change radius are then changed consistently.

The local transformation consists of a random rotation around the up axis as well as a planar shift of random magnitude in specified limits. The minimal magnitude ensures that changes fulfill Definition 2.2. To obtain somewhat realistic changes that yield changed landmarks which are still close to the route, rotations are limited in their axis and shifts are planar in a local scope. This procedure is repeated until the desired change ratio is reached.

While most simulated changes are realistic, some changes are easy to spot for the human eye. For instance, a pole with signs in the middle of the ego lane cannot be up-to-date anymore. For this, it is important to note that such common knowledge is not used by the proposed method. Also, when comparing maps from 2020 with sensor data from 2023, such seemingly strange changes can indeed be observed *e.g.* when a lane has been moved to an area where poles, signs or traffic lights have been placed previously. Hence, with the conscious focus on difficult changes in mind, simulated changes are deemed sufficient for evaluation.

5.3.2 Quantitative Results

To obtain quantitative results, up-to-date maps from 2023 are used under the assumption that between two drives on the same route no change has occurred. As the mapping and verification sequences on the same route are mere hours apart, this is almost certainly true.

Using the change simulation described above, at least $15\,\%$ of map elements of each semantic class are changed. As examined in Chapter 2, this corresponds to the typical change rate over one year. The group selection radius is $2\,\mathrm{m}$ and shifts range from $1\,\mathrm{m}$ to $5\,\mathrm{m}$.

To evaluate the approach, two metrics are of interest. The first is the range at which a certain verification belief is reached. Verification at maximal sensing ranges is desirable, but trusting a single measurement could lead to wrong results. Hence, the second metric measures overly optimistic verification by evaluating the verification belief $\mathrm{bel}(\omega_{\mathrm{ver}})$ erroneously assigned to *changed* map elements. In practice, belief is carefully aggregated over multiple measurements and there is a trade-off between both metrics.

In Figure 5.3 the share of visible landmarks which have surpassed a certain minimal verification belief is depicted over distance to the map element. Visibility is approximated by more than 2 lidar returns on the landmark, *i.e.* $\left|\mathfrak{L}_{\ell}^{0}\right| > 2$. Additionally, the shares in safe and comfortable breaking distance are reported in Table 5.2.

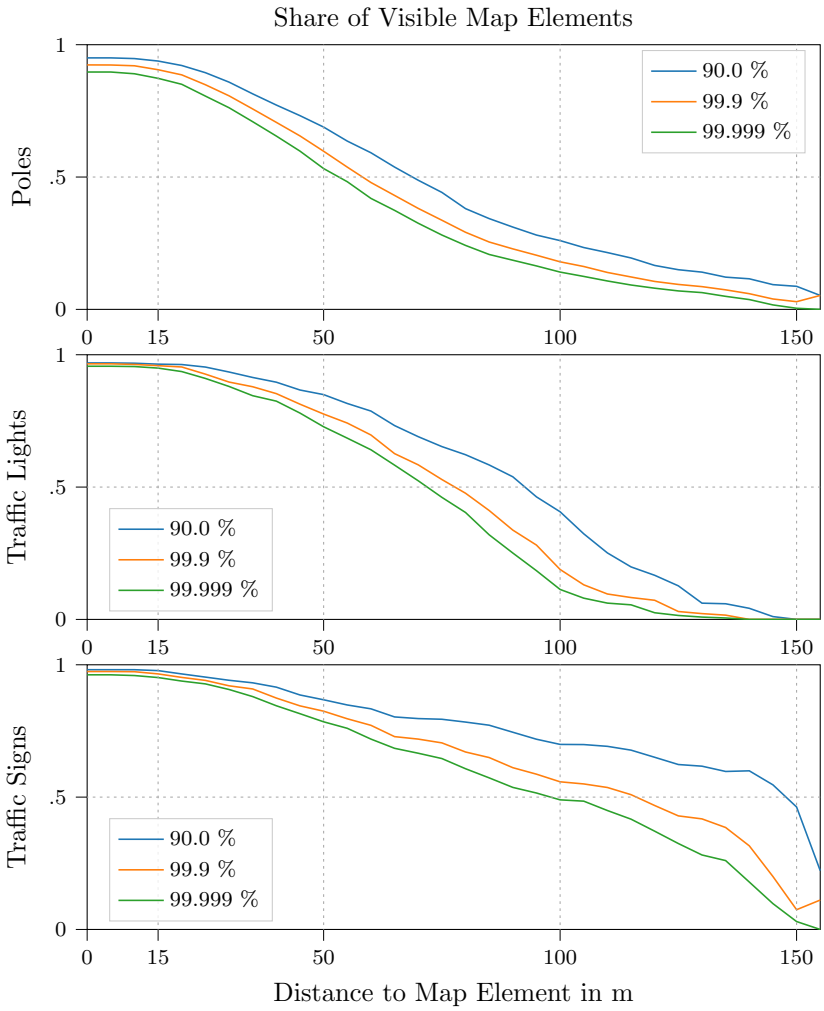


Figure 5.3: Share of visible map elements which have been verified with at least a certain belief, encoded in different colors, depicted over distance from landmark to vehicle. The higher the desired verification belief, the more frames are required and the closer the landmark can only be verified.
Note that only map elements in 150 m radius around the coarse vehicle's position were retrieved for potential association, posing an upper limit for verification.

Table 5.2: Shares of visible landmarks which are verified with at least a a certain minimal verification belief $\mathrm{bel}(\omega_{\mathrm{ver}})$ at certain distances. The distances, $15\,\mathrm{m}$ and $50\,\mathrm{m}$, are safe and comfortable breaking distances in urban traffic, respectively. All numbers are averaged over all four validation sequences using the 2023 sensor setup and reported in $\%$.

Minimal	**Poles**		**Traffic Lights**		**Traffic Signs**	
Belief	$15\,\mathrm{m}$	$50\,\mathrm{m}$	$15\,\mathrm{m}$	$50\,\mathrm{m}$	$15\,\mathrm{m}$	$50\,\mathrm{m}$
90.0	93.8	68.9	96.4	84.9	97.8	86.8
99.9	90.5	59.7	95.9	77.6	96.6	82.5
99.999	87.3	53.2	95.0	72.8	95.2	78.5

Poles can only be verified at rather close range, but have little semantic meaning except for localization. A significant share of map elements which are relevant for driving behavior, traffic lights and traffic signs, are verified with $99.999\,\%$ certainty beyond the comfortable breaking distance of $50\,\mathrm{m}$. Every second visible traffic sign is already verified at this remarkable level in $100\,\mathrm{m}$ distance.

Unfortunately, the map used for evaluation does not contain any information about the relevance of traffic lights and signs for the ego lane, *i.e.* those elements that are actually interesting for driving behavior. However, qualitative results show that such relevant elements are usually visible far in advance and, hence, can be verified early.

Successful verification at large distances is desirable and necessary for comfortable driving by relying on map information. However, its opposite, falsely confirming actually changed map elements, compromises safety and might have catastrophic consequences. Hence, it is even more important to evaluate falsely attributed verification belief.

For individual map elements over one or few time steps, there are wrong associations that could potentially result in wrong verification belief. However, they either vanish in the overall set of more likely association hypotheses, leading to marginalized probabilities smaller than $\tau^{\mathrm{c}}_{\mathrm{VMA}}$. Or the temporally consistent evidence from change detection is overwhelming the temporally spurious evidence from false associations. Hence, over all four sequences and more than 1000 simulated changes, the largest accumulated verification belief $\mathrm{bel}(\omega_{\mathrm{ver}})$ assigned to a changed map element was observed at $0.006\,\%$. This number is in

strong contrast with the minimal beliefs for verification assumed in Figure 5.3 and Table 5.2. As a result, it can be concluded that the proposed method for map verification yields trustworthy results at more than sufficient ranges.

If the verification or change detection only needs to happen eventually, *e.g.* to trigger a map update, the last known aggregated belief for each landmark can be evaluated to classify their state. Table 5.3 shows classification metrics using a belief threshold of 0.99, *i.e.* if the changed or verification belief is larger, a landmark is detected as changed and verified, respectively. To include additional map elements in an update, the verification step could be complemented with the mapping approach proposed in Chapter 3. It could be used on board a vehicle, then using a causal processing order, to track and estimate detections not associated to any known map element.

In terms of computation time, one first needs to note that the code for map verification has not been optimized or parallelized since a permanent deployment in the software stack of MRT's research vehicles is not expected for this part of the thesis. Still, for all landmarks in $150\,\mathrm{m}$ radius, the verification step takes around $10\,\mathrm{ms}$, which is acceptable even for use in a real-time system.

Table 5.3: Performance of the proposed approach to only eventually classify the landmark state. For this, the last known temporally aggregated belief for each landmark is evaluated with a threshold of 0.99. If the verification/change belief is higher, a map element counts as verified/changed. All numbers are reported in %.

	Verification			Change		
Type	**Precision**	**Recall**	**F_1 Score**	**Precision**	**Recall**	**F_1 Score**
Poles	100.0	91.8	95.7	89.6	93.5	91.5
Traffic Lights	100.0	95.9	97.9	95.7	98.3	97.0
Traffic Signs	100.0	96.6	98.3	94.4	95.8	95.1

5.3.3 Qualitative Results

Using maps from 2020 and sensor data from 2023 allows us to examine the detection of real world changes over three years. While there is no ground truth available, in Figures 5.4 to 5.6 qualitative examples are depicted.

Figures 5.4 and 5.5 shows that most changes can be resolved well. As can be seen especially in the lower image of Figure 5.5, even in largely outdated maps, the unchanged parts can be verified successfully.

In Figure 5.6, the two open issues of the method become apparent: Traffic lights with changed orientation as well as signs or traffic lights with changed semantics cannot be detected. Both are discussed in detail in Section 5.5.

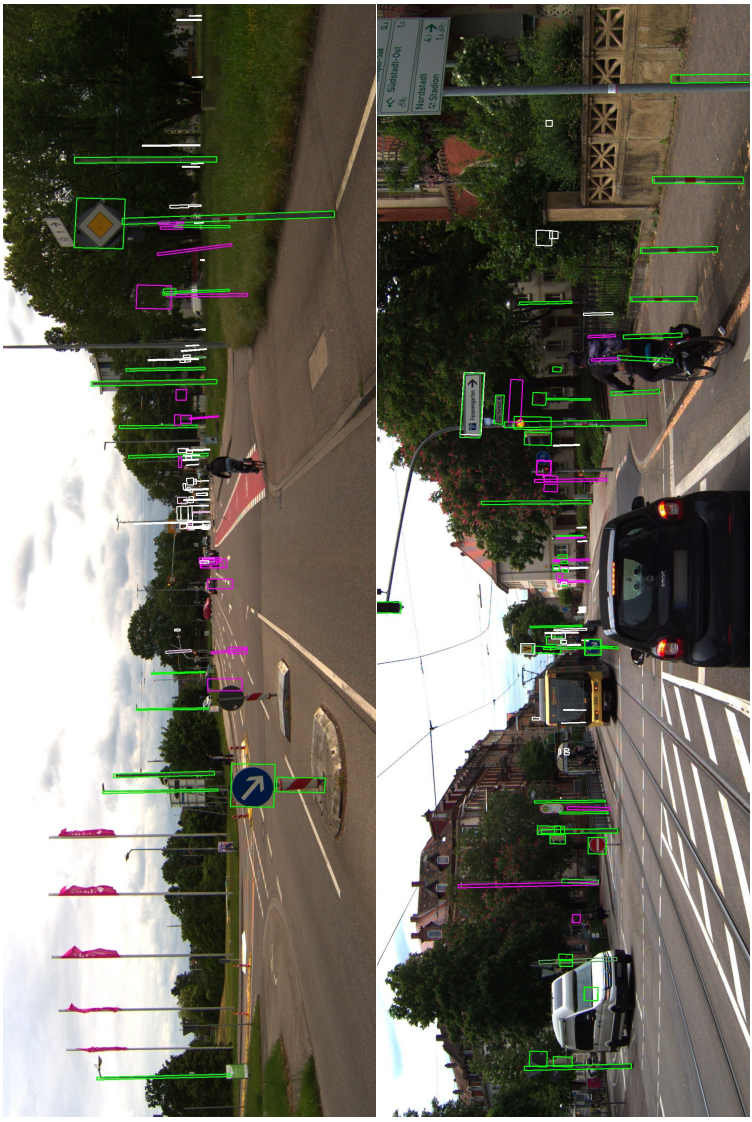


Figure 5.4: Qualitative examples of the map verification results. Verified map elements are colored in shades of green; outdated map elements are depicted magenta. Map elements without any significant belief are painted white.

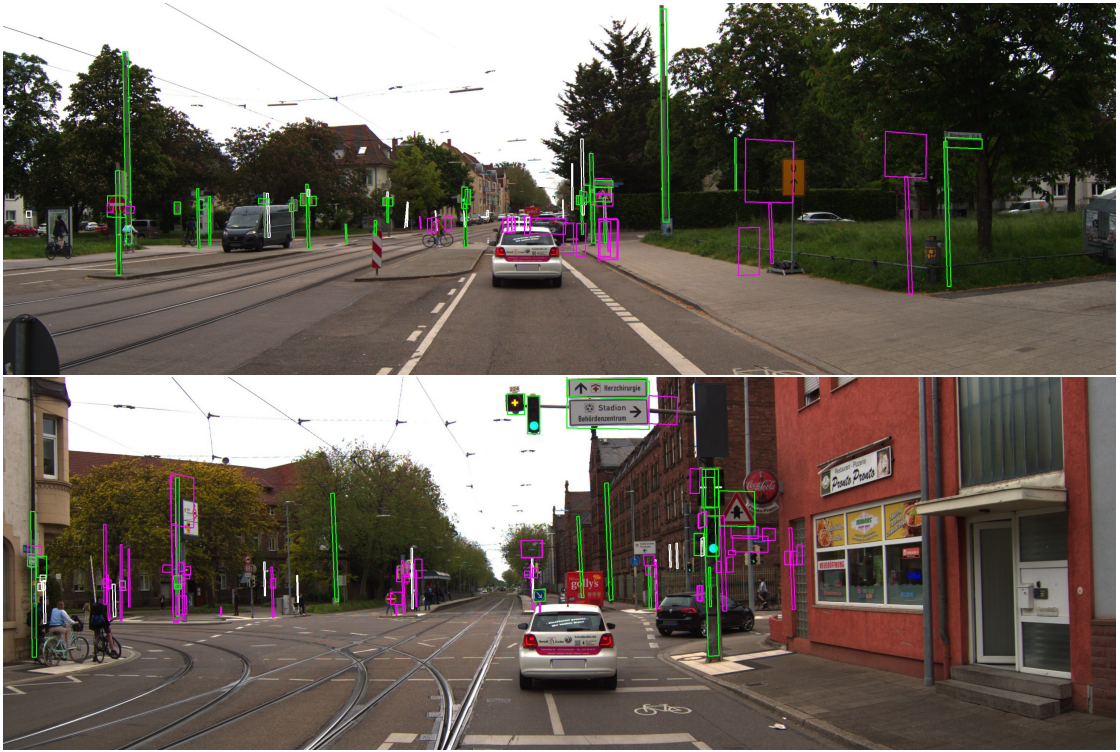


Figure 5.5: Qualitative examples of the map verification results. Verified map elements are colored in shades of green; outdated map elements are depicted magenta. Map elements without any significant belief are painted white.

Figure 5.6: Examples depicting the main limitations of the proposed approach. In the upper image, map elements falsely mapped closely on or within the wall are prone to false verification since the lidar change detector cannot counteract spurious false detections (1). Since traffic lights are measured without orientation or detailed semantics, replacements with similar positions and suitable extent are falsely matched (2).
The crop in the lower image, extracted from the 2020 mapping drive, shows that the semantic resolution needs to be improved significantly in order to distinguish traffic signs with different meanings.

5.4 Verification Beyond the Physical Layer

The proposed approach is the first to verify semantic map elements significantly ahead of the vehicle using onboard sensors. However, by its principle, it is limited to physically detectable map elements. It is certainly conceivable to collect and assign evidence for abstract map elements, *e.g.* by detecting other traffic participants which can be assigned lanes. But, not every abstract element can be verified in this way and most such indirect evidence clues rely crucially on map compliant behavior of others, which is highly questionable given the desirable reliability of a verification system. Approaches which reliably infer abstract elements, like lanes, without other traffic participants are still limited to the close vicinity of the ego vehicle, hence, lacking the necessary range to react on verification results.

As an alternative and readily available interim solution, this section describes how an appropriately designed HAD map enables propagating evidence from the physical to abstract layers. This is particularly important since the major benefit of HAD maps lies not in physical features, but in abstract semantic information, such as lane topology, traffic light assignments or traffic rules.

The idea builds upon two requirements. First, it is assumed that the abstract elements are stored with a semantic relation to the physical element they emerged from. For instance, a lane is defined by its lane borders which again are defined of individual markings or curb stones. For traffic light assignments, each lane is in a semantic relation with all individual traffic lights which act in unison as well as the stop line(s) belonging to the traffic lights. An example of such a map framework is Lanelet2 [PPJ+18], which was co-developed by the author of this thesis.

The second assumption is a deterministic mapping process that only uses information from physical map elements to infer abstract layers. A concrete example is the framework developed by Poggenhans [Pog19]. Other approaches, including manual annotations by humans and deep learning methods, are conceivable as well as long as they only process physical map elements which are contained in the map.

Naively, one could use all verified map elements to infer abstract elements again. However, with limited sensor data, it might not be possible to resolve all ambiguities for *e.g.* sign or traffic light to lane assignment.

Instead, it can be argued that unchanged physical elements always yield identical abstract relationships between them. The idea of the argument is that abstract map information, like traffic rules or traffic light to lane assignments, need to be inferable unambiguously for a human driver given the physical map elements, like markings or road signs. Hence, if at the time of map creation a certain abstract information could be inferred, the same result would be produced by running the deterministic mapping process again.

Even after years of research, the author of this thesis is not aware of any change where unchanged physical elements lead to different abstract information, even when independent map elements were added close by. Still, the idea might best be viewed as conjecture that is to be confirmed using a large-scale map dataset with long-term changes. It also raises the issue how to handle additional map elements. Trivially, if no new map elements are detected, the verification results can be propagated to abstract layers without restrictions. The difficult question will be how to determine how newly added map elements interfere with this procedure.

5.5 Limitations

While the performance of the proposed map verification method is sufficient for the goals set for this thesis, currently, it still has three major limitations that need to be discussed.

First, since the orientation of traffic lights is hardly visible from lidar point clouds and not contained in the output of the DNN used for object detection, it is not yet incorporated into the parametric representation. Hence, traffic lights at similar positions and with compatible dimensions but with changed orientation seem identical to the proposed method and lead to falsely verified map elements. While Plachetka et al. [PSF+22, PSF+23] claim to extract traffic light orientation from lidar, the author of this thesis doubts that their method

will show sufficient performance when used on real lidar scans at high range, *i.e*. low point density. Instead, using information from camera images seems far more promising as ongoing work at the MRT shows.

The second issue is the very limited semantic resolution for both road signs and traffic lights. For instance, if a traffic light with two lights is replaced by one with three lights, this cannot be distinguished reliably in parameter space. However, state-of-the-art traffic light datasets [FMK+18] contain such information, arrows, and special symbols, which all could be used to train a DNN with vastly more fine-grained output. Similarly, for road signs, there are cases where the position and shape of a sign remained almost identical, but the meaning of a sign was changed, *e.g*. with a different speed limit. Only if those semantic differences are resolved in the classes of the DNN they can be detected. For suitable, often country specific datasets, the reader is referred to two surveys [LLC+19, JGB+20].

The last issue is that the change detector is mainly based on lidar visibility information. If the absence of a map element cannot be measured in this way, like for the traffic light (1) in Figure 5.6, spurious detections can accumulate verification belief without any conflicting evidence. Using a change detector that uses visual clues, as proposed by related works, can solve this issue. Even using the largest IoU with any detection mask as feature for the change classifier could do so, but comes at the cost of correlating both evidence sources significantly.

5.6 Conclusion

This chapter presented a method for verifying a semantic HAD map. The intricacies of change detection and map verification were discussed, leading to the adaptation of a ternary evidence framework.

To collect temporally uncorrelated evidence against changes, the globally probabilistically optimal data association and localization approach proposed in Chapter 4 is used. The evidence is gained by marginalization over the most likely association hypotheses.

For the detection of changes a random forest based on a lidar visibility check is proposed. While being a traditional classifier, it is very fast and sufficiently accurate to detect changes.

Combining both evidences over time makes it possible to master the challenge of achieving high verification range without false verification results in simulation. Poles, which are unimportant for driving behavior, are typically verified with high certainty in about $50\,\mathrm{m}$ distance. Actually relevant traffic lights and road signs are verified $64\,\mathrm{m}$ and $72\,\mathrm{m}$ ahead of the vehicle, respectively. At the same time, false positive verification beliefs have not been accumulated for any of the more than 1000 simulated changes.

In addition to the verification of physical map elements, a concept has been drafted to propagate verification results from the physical to abstract map layers. It makes use of a relational semantic HAD map framework and allows reasoning about changes or verification of abstract semantic map information, such as traffic rules, without requiring the still complex map perception and inference onboard the vehicle.

To conclude, in contrast to the previous state of the art, the proposed method offers sufficient range to allow an automated vehicle to safely react on changes or enable comfortable driving by relying on verified map information. For the first time, this makes it possible to dependably rely on map elements merely based on sensor information collected and processed onboard the ego vehicle.

As an outlook, newly added map elements could be aggregated using the mapping approach presented earlier, allowing a fusion of online perception and verified map content. While this thesis is restricted to only three kinds of physical map elements, extending the approach to road markings seems trivial given a suitable DNN for detection.

6 Conclusion and Outlook

For the foreseeable future, automated vehicles will not be able to interpret the often complex static environment with satisfactory reliability on board in real time. This necessitates the use of semantic HD maps which act as powerful virtual sensor that provides access to the static world far beyond on-board sensing and processing capabilities, thus, enabling safe and comfortable automated driving. Changes in the world, however, render maps outdated. When an automated vehicle falsely relies on an outdated map, the consequences can be fatal.

6.1 Conclusion

To prevent this and enable trustworthy maps, the work at hand proposed a system to continuously verify a semantic HD map in advance using only on-board sensors. In order to verify relevant map elements sufficiently far ahead of the vehicle, three major advancements were necessary.

The first advancement concerns the precise detection of map elements sufficiently far in advance. To achieve this, semantic instance detections from a DNN processing camera images are fused with lidar point clouds. This hybrid late fusion approach enables detections at ranges of up to $180\,\mathrm{m}$ with a precision close to human annotations.

The key innovation are parametric detections which are tailored to model each semantic class of map elements specifically. They are detailed enough to enable meaningful projections into camera or range images, which is necessary for active change detection. At the same time, the limited number of parameters can be estimated robustly even at large distances. Furthermore, the parametric representations facilitate association over time even across grave appearance

changes and, thus, allow creating an highly automated driving (HAD) map fully automatically by mere robust averaging.

Unfortunately, the presented survey on map changes made it clear that accurate detections are at most half the battle. The real core challenge is the coupled localization and data association problem in partially outdated maps. To solve it, a novel concept for probabilistically data association, called probabilistic correspondence graph (PCG), has been proposed as second advancement.

With merely a coarse initial position and using only a single frame of measurements it can achieve globally probabilistically optimal data association in real time. Moreover, it can self-assess its performance and avoid false localization in ambiguous environments like partially changed maps or periodic highway sections. On both simulated data association problems and on the KITTI benchmark [GLU12, GLS+13], it outperforms the previous state of the art.

With only a single frame of measurements PCG achieves about $2\,\mathrm{cm}$ and $0.02°$ average error. Thus, it outperforms previous filtering and graph optimization approaches that use similar HAD maps, but require multiple frames. When compared to approaches that use sensor-specific localization layers, it achieves similar accuracy. However, its parametric map is orders of magnitude more compact. When localizing in three years old and therefore partially outdated maps, localization availability is reduced, but accuracy is almost unimpaired at about $3\,\mathrm{cm}$ and $0.03°$ average error.

Highly accurate detections and a powerful data association method enable a ternary map verification approach as third major contribution. Using evidence theory, it tracks belief masses of verification and change independently, which makes it possible to distinguish changes and occlusions. By resolving verification results to individual landmarks, verified parts of the map can still be used safely even in the presence of changes that are irrelevant for the ego route.

Evidence for the verification of a landmark is marginalized over all association hypotheses that assigned a detection to the map element in question. To measure changes, an active approach based on ray casting in range images is proposed. Both together enable the verification of traffic lights and signs with exceptional certainty significantly beyond $50\,\mathrm{m}$, which can be seen as comfortable breaking

distance in the urban context. At the same time, none of more than 1000 simulated changes was falsely verified.

Conceptually, using Lanelet2 [PPJ+18] as map framework, this work described how verification results can be transferred from the physical to abstract map layers without the need to infer information on the abstract layer.

Currently, map providers outdo each other regarding "map freshness" as limiting factor for the safe use of HAD maps. The work at hand offered an alternative perspective on semantic HD maps and their verification. Using only on-board sensors and processing, it enables the identification of up-to-date parts at a comfortable breaking distance. Regardless of other drives or map updates, these map elements can then be safely trusted.

6.2 Outlook

One limitation of the approach is the semantic resolution of traffic lights and signs. More fine-grained semantic classes, *e.g.* the number of lights or the exact type of traffic sign, have the potential to accelerate the data association and possibly even improve its quality even further. But, more importantly, they would remedy current semantic confusion errors visible in the qualitative results of the map verification.

In this thesis, the fully automated mapping was only used to create an exemplary HAD map. Its human-like quality also allows it to serve as pseudo ground truth to train deep learning approaches that are currently hindered by the lack of suitable ground truth. Learning to directly predict parametric representations could improve detection range and quality even further. An alternative or possibly complementary path to end-to-end deep learning is the proposed weakly and self-supervised Rendering Instance IoU (RIIoU) metric.

When using maps as pseudo ground truth for deep learning approaches, outdated map elements will erroneously induce false losses. In this regard, the approaches proposed in the work at hand can be used in three ways. PCG's self-assessment can determine if localization quality is sufficient to correctly reproject the

map into sensor data or to fuse both. The evidential map verification, ideally performed acausally over all frames, can then tell which map elements are still up-to-date. Finally, the proposed ray casting can be used to estimate detectability given the respective sensor pose.

Lastly, map updates are often tackled in parallel with map change detection, but were disregarded in this work. However, the fully automated mapping could be used during runtime to track unassociated parametric detections. This would not only enable map updates that can be shared with other vehicles or a server backend. It could also provide a comprehensive view that combines verified map content with newly detected elements.

Bibliography

[AGM+13] Arbelaitz, Olatz; Gurrutxaga, Ibai; Muguerza, Javier; Pérez, Jesús M. and Perona, Iñigo: "An extensive comparative study of cluster validity indices". In: *Pattern Recognition* 46.1 (2013), pp. 243–256. doi: 10.1016/j.patcog.2012.07.021 (cit. on p. 291).

[AHB87] Arun, K. S.; Huang, T. S. and Blostein, S. D.: "Least-Squares Fitting of Two 3-D Point Sets". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-9.5 (1987), pp. 698–700. doi: 10.1109/TPAMI.1987.4767965 (cit. on pp. 107, 155).

[AL17] Asadi, Kavosh and Littman, Michael L.: "An Alternative Softmax Operator for Reinforcement Learning". In: *Proceedings of the 34th International Conference on Machine Learning*. Sydney, NSW, Australia. Aug. 6–11, 2017. Ed. by Precup, Doina and Teh, Yee Whye. Vol. 70. Proceedings of Machine Learning Research. 2017, pp. 243–252. url: https://proceedings.mlr.press/v70/asadi17a.html (last retrieved 2024-03-24) (cit. on p. 81).

[AMH21] Awad, N.; Mallik, N. and Hutter, F.: "DEHB: Evolutionary Hyberband for Scalable, Robust and Efficient Hyperparameter Optimization". In: *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*. Virtual Conference. Aug. 19–26, 2021. Ed. by Zhou, Z. ijcai.org, 2021, pp. 2147–2153 (cit. on p. 79).

[AMT20] Agarwal, Sameer; Mierle, Keir and The Ceres Solver Team: Ceres Solver. Version 2.0. Oct. 2020. url: https://

github.com/ceres-solver/ceres-solver (last retrieved 2023-03-24) (cit. on pp. 71, 156).

[Bai02] Bailey, Tim: "Mobile Robot Localisation and Mapping in Extensive Outdoor Environments". Doctoral Dissertation. Sydney, NSW, Australia: Australian Centre for Field Robotics, Department of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney, 2002 (cit. on pp. 116, 118, 130).

[Bar01] Bar-Shalom, Yaakov: Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software. Ed. by Li, X. Rong and Kirubarajan, Thiagalingam. New York City, NY, USA: John Wiley & Sons, Inc, 2001. doi: 10.1002/0471221279 (cit. on p. 116).

[BB15] Bubb, Heiner and Bengler, Klaus: Automobilergonomie. ATZ/MTZ-Fachbuch. Wiesbaden, Germany: Springer Vieweg, 2015 (cit. on p. 189).

[BC82] Bolles, Robert C. and Cain, Ronald A.: "Recognizing and Locating Partially Visible Objects: The Local-Feature-Focus Method". In: *The International Journal of Robotics Research* 1.3 (1982), pp. 57–82. doi: 10.1177/027836498200100304 (cit. on pp. 118, 130).

[BE14] Bergström, Per and Edlund, Ove: "Robust registration of point sets using iteratively reweighted least squares". In: *Computational Optimization and Applications* 58.3 (July 2014), pp. 543–561. doi: 10.1007/s10589-014-9643-2 (cit. on pp. 157, 168).

[BE76] Bollobas, B. and Erdős, P.: "Cliques in Random Graphs". In: *Mathematical Proceedings of the Cambridge Philosophical Society* 80.3 (1976), pp. 419–427 (cit. on p. 303).

[Bec21] Beck, Johannes: "Camera Calibration with Non-Central Local Camera Models". Doctoral Dissertation. Karlsruhe, Germany: Karlsruher Institut für Technologie (KIT), 2021. 136 pp. doi: 10.5445/IR/1000131090 (cit. on p. 42).

[BGB+22] Boubakri, Anis; Gammar, Sonia METTALI; Brahim, Mohamed BEN and Filali, Fethi: "High definition map update for autonomous and connected vehicles: A survey". In: *2022 International Wireless Communications and Mobile Computing (IWCMC)*. Dubrovnik, Croatia. May 30–June 3, 2022. 2022, pp. 1148–1153. doi: 10.1109/IWCMC55113.2022.9825276 (cit. on p. 10).

[BGM+19] Behley, Jens; Garbade, Martin; Milioto, Andres; Quenzel, Jan; Behnke, Sven; Stachniss, Cyrill and Gall, Jürgen: "SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. Seoul, Korea (South). Oct. 27–Nov. 2, 2019. 2019, pp. 9296–9306. doi: 10.1109/ICCV.2019.00939 (cit. on p. 32).

[BGP19] Babin, Philippe; Giguère, Philippe and Pomerleau, François: "Analysis of Robust Functions for Registration Algorithms". In: *2019 International Conference on Robotics and Automation (ICRA)*. Montreal, QC, Canada. May 20–24, 2019. 2019, pp. 1451–1457. doi: 10.1109/ICRA.2019.8793791 (cit. on p. 71).

[BHS+23] Bieder, Frank; Hu, Haohao; Schantz, Johannes; Kirik, Oguzhan; Ries, Florian; Haueis, Martin and Stiller, Christoph: "Ein Ansatz zur automatisierten Erstellung von Trainingsdaten unter Verwendung von HD-Karten und Mehrfachbefahrungen". In: *15. Workshop Fahrerassistenz und automatisiertes Fahren*. Berkheim, Germany. Oct. 24–26, 2023. 2023, pp. 17–26 (cit. on p. 107).

[BK73] Bron, Coen and Kerbosch, Joep: "Algorithm 457: Finding All Cliques of an Undirected Graph". In: *Commun. ACM* 16.9 (Sept. 1973), pp. 575–577. doi: 10.1145/362342.362367 (cit. on p. 118).

[BKM+20] Beker, Deniz; Kato, Hiroharu; Morariu, Mihai Adrian; Ando, Takahiro; Matsuoka, Toru; Kehl, Wadim and

Gaidon, Adrien: "Monocular Differentiable Rendering for Self-supervised 3D Object Detection". In: *Computer Vision – ECCV 2020*. Virtual Conference. Aug. 23–28, 2020. Ed. by Vedaldi, Andrea; Bischof, Horst; Brox, Thomas and Frahm, Jan-Michael. Springer International Publishing, 2020, pp. 514–529 (cit. on p. 74).

[Blu53] Blumenthal, Leonard M.: Theory and Applications of Distance Geometry. Oxford, Great Britain: Clarendon Press, 1953 (cit. on pp. 131, 141, 142).

[BLZ+21] Bai, Xuyang; Luo, Zixin; Zhou, Lei; Chen, Hongkai; Li, Lei; Hu, Zeyu; Fu, Hongbo and Tai, Chiew-Lan: "PointDSC: Robust Point Cloud Registration using Deep Spatial Consistency". In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Nashville, TN, USA. June 20–25, 2021. 2021, pp. 15854–15864. doi: 10.1109/CVPR46437.2021.01560 (cit. on pp. 122, 168, 169).

[BM92] Besl, P.J. and McKay, Neil D.: "A method for registration of 3-D shapes". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14.2 (1992), pp. 239–256. doi: 10.1109/34.121791 (cit. on p. 120).

[BMN19] Barath, Daniel; Matas, Jiří and Noskova, Jana: "MAGSAC: Marginalizing Sample Consensus". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA. June 15–20, 2019. 2019, pp. 10189–10197. doi: 10.1109/CVPR.2019.01044 (cit. on pp. 120, 162).

[BN18] Blaga, Bianca-Cerasela-Zelia and Nedevschi, Sergiu: "A Method for Automatic Pole Detection from Urban Video Scenes using Stereo Vision". In: *2018 IEEE 14th International Conference on Intelligent Computer Communication and Processing (ICCP)*. Cluj-Napoca, Romania. Sept. 6–8, 2018. 2018, pp. 293–300. doi: 10.1109/ICCP.2018.8516640 (cit. on p. 35).

[BNB17] Behrendt, Karsten; Novak, Libor and Botros, Rami: "A deep learning approach to traffic lights: Detection, tracking, and classification". In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. Singapore, Singapore. May 29–June 3, 2017. 2017, pp. 1370–1377. doi: 10.1109/ICRA.2017.7989163 (cit. on p. 34).

[BNG+16] Burri, Michael; Nikolic, Janosch; Gohl, Pascal; Schneider, Thomas; Rehder, Joern; Omari, Sammy; Achtelik, Markus W and Siegwart, Roland: "The EuRoC micro aerial vehicle datasets". In: *The International Journal of Robotics Research* 35.10 (2016), pp. 1157–1163. doi: 10.1177/0278364915620033 (cit. on p. 170).

[BNI+20] Baráth, Dániel; Noskova, Jana; Ivashechkin, Maksym and Matas, Jiří: "MAGSAC++, a Fast, Reliable and Accurate Robust Estimator". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Seattle, WA, USA. June 13–19, 2020. 2020, pp. 1301–1309. doi: 10.1109/CVPR42600.2020.00138 (cit. on pp. 120, 162).

[BNM22] Barath, Daniel; Noskova, Jana and Matas, Jiri: "Marginalizing Sample Consensus". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.11 (2022), pp. 8420–8432. doi: 10.1109/TPAMI.2021.3103562 (cit. on pp. 120, 162).

[BNR+00] Bailey, T.; Nebot, E.M.; Rosenblatt, J.K. and Durrant-Whyte, H.F.: "Data association for mobile robot navigation: a graph theoretic approach". In: *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*. San Francisco, CA, USA. Apr. 24–28, 2000. Vol. 3. 2000, pp. 2512–2517. doi: 10.1109/ROBOT.2000.846406 (cit. on pp. 118, 130).

[Bol79] Bolles, Robert C.: “Robust Feature Matching Through Maximal Cliques”. In: *Imaging Applications for Automated Industrial Inspection and Assembly*. Washington, D.C., USA. Oct. 10, 1979. Vol. 0182. International Society for Optics and Photonics. SPIE, 1979, pp. 140–149. doi: 10.1117/12.957381 (cit. on pp. 118, 130).

[Bot17] Botev, Z. I.: “The Normal Law under Linear Restrictions: Simulation and Estimation via Minimax Tilting”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 79.1 (2017), pp. 125–148 (cit. on p. 162).

[BPF12] Beyerer, Jürgen; Puente Leon, Fernando and Frese, Christian: Automatische Sichtprüfung: Grundlagen, Methoden und Praxis der Bildgewinnung und Bildauswertung. Berlin, Heidelberg, Germany: Springer Verlag, 2012 (cit. on p. 30).

[Bra00] Bradski, Gary: “The OpenCV Library”. In: *Dr. Dobb’s Journal of Software Tools* 25.11 (Nov. 2000). url: https://opencv.org/ (last retrieved 2024-02-29) (cit. on p. 199).

[Bre09] Brenner, Claus: “Global Localization of Vehicles Using Local Pole Patterns”. In: *Pattern Recognition. DAGM 2009. Lecture Notes in Computer Science*. Jena, Germany. Sept. 9–11, 2009. Ed. by Denzler, Joachim; Notni, Gunther and Süsse, Herbert. Vol. 5748. Springer Berlin Heidelberg, 2009, pp. 61–70 (cit. on pp. 36, 123).

[Bru22] Brunzema, Paul: Efficient sampling from the truncated MVN. 2022. url: https://github.com/brunzema/truncated-mvn-sampler (last retrieved 2024-02-13) (cit. on p. 162).

[BS18] Beck, Johannes and Stiller, Christoph: “Generalized B-spline Camera Model”. In: *2018 IEEE Intelligent Vehicles Symposium (IV)*. Changshu, China. June 26–30, 2018. 2018, pp. 2137–2142. doi: 10.1109/IVS.2018.8500466 (cit. on p. 42).

[BSD18] Bach, Martin; Stumper, Daniel and Dietmayer, Klaus: "Deep Convolutional Traffic Light Recognition for Automated Driving". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. Maui, HI, USA. Nov. 4–7, 2018. 2018, pp. 851–858. doi: 10.1109/ITSC.2018.8569522 (cit. on p. 34).

[BVR+20] Bhavsar, Sagar Ravi; Vatavu, Andrei; Rehfeld, Timo and Krehl, Gunther: "Sensor Fusion-based Online Map Validation for Autonomous Driving". In: *2020 IEEE Intelligent Vehicles Symposium (IV)*. Las Vegas, NV, USA. Oct. 19–Nov. 13, 2020. 2020, pp. 77–82. doi: 10.1109/IV47402.2020.9304535 (cit. on p. 194).

[BWS+22] Berrio, Julie Stephany; Worrall, Stewart; Shan, Mao and Nebot, Eduardo: "Long-Term Map Maintenance Pipeline for Autonomous Vehicles". In: *IEEE Transactions on Intelligent Transportation Systems* 23.8 (2022), pp. 10427–10440. doi: 10.1109/TITS.2021.3094485 (cit. on p. 193).

[BYW+23] Burnett, Keenan; Yoon, David J; Wu, Yuchen; Li, Andrew Z; Zhang, Haowei; Lu, Shichen; Qian, Jingxing; Tseng, Wei-Kang; Lambert, Andrew; Leung, Keith YK et al.: "Boreas: A multi-season autonomous driving dataset". In: *The International Journal of Robotics Research* 42.1-2 (2023), pp. 33–42. doi: 10.1177/02783649231160195 (cit. on p. 33).

[BZS14] Bender, Philipp; Ziegler, Julius and Stiller, Christoph: "Lanelets: Efficient map representation for autonomous driving". In: *2014 IEEE Intelligent Vehicles Symposium Proceedings*. Dearborn, MI, USA. June 8–11, 2014. 2014, pp. 420–425. doi: 10.1109/IVS.2014.6856487 (cit. on p. 8).

[CBL+20] Caesar, Holger; Bankiti, Varun; Lang, Alex H.; Vora, Sourabh; Liong, Venice Erin; Xu, Qiang; Krishnan, Anush; Pan, Yu; Baldan, Giancarlo and Beijbom, Oscar: "nuScenes: A Multimodal Dataset for Autonomous Driving". In: *2020 IEEE/CVF Conference on Computer*

Vision and Pattern Recognition (CVPR). Seattle, WA, USA. June 14–19, 2020. 2020, pp. 11618–11628. DOI: 10.1109/CVPR42600.2020.01164 (cit. on pp. 41, 44).

[CBT+23] CRAMARIUC, Andrei; BERNREITER, Lukas; TSCHOPP, Florian; FEHR, Marius; REIJGWART, Victor; NIETO, Juan; SIEGWART, Roland and CADENA, Cesar: "maplab 2.0 – A Modular and Multi-Modal Mapping Framework". In: *IEEE Robotics and Automation Letters* 8.2 (2023), pp. 520–527. DOI: 10.1109/LRA.2022.3227865 (cit. on p. 125).

[CCC+16] CADENA, Cesar; CARLONE, Luca; CARRILLO, Henry; LATIF, Yasir; SCARAMUZZA, Davide; NEIRA, José; REID, Ian and LEONARD, John J.: "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age". In: *IEEE Transactions on Robotics* 32.6 (2016), pp. 1309–1332. DOI: 10.1109/TRO.2016.2624754 (cit. on p. 39).

[CER+21] CAMPOS, Carlos; ELVIRA, Richard; RODRÍGUEZ, Juan J. Gómez; M. MONTIEL, José M. and D. TARDÓS, Juan: "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM". In: *IEEE Transactions on Robotics* 37.6 (2021), pp. 1874–1890. DOI: 10.1109/TRO.2021.3075644 (cit. on p. 125).

[CLF+21] CHEN, Siheng; LIU, Baoan; FENG, Chen; VALLESPI-GONZALEZ, Carlos and WELLINGTON, Carl: "3D Point Cloud Processing and Learning for Autonomous Driving: Impacting Map Creation, Localization, and Perception". In: *IEEE Signal Processing Magazine* 38.1 (2021), pp. 68–86. DOI: 10.1109/MSP.2020.2984780 (cit. on p. 125).

[CLG+19] CHEN, Wenzheng; LING, Huan; GAO, Jun; SMITH, Edward; LEHTINEN, Jaakko; JACOBSON, Alec and FIDLER, Sanja: "Learning to Predict 3D Objects with an Interpolation-based Differentiable Renderer". In: *Advances in Neural Information Processing Systems*. Vancouver, BC, Canada. Dec. 8–14, 2019. Ed. by

Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E. and Garnett, R. Vol. 32. Red Hook, NY, USA: Curran Associates, Inc., 2019, pp. 9577–9587 (cit. on p. 74).

[CLS+19] Chang, Ming-Fang; Lambert, John; Sangkloy, Patsorn; Singh, Jagjeet; Bak, Slawomir; Hartnett, Andrew; Wang, De; Carr, Peter; Lucey, Simon; Ramanan, Deva et al.: "Argoverse: 3D Tracking and Forecasting With Rich Maps". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA. June 16–20, 2019. 2019, pp. 8740–8749. doi: 10.1109/CVPR.2019.00895 (cit. on pp. 41, 44).

[CLW+22] Cai, Yingfeng; Lu, Ziheng; Wang, Hai; Chen, Long and Li, Yicheng: "A Lightweight Feature Map Creation Method for Intelligent Vehicle Localization in Urban Road Environments". In: *IEEE Transactions on Instrumentation and Measurement* 71 (2022), pp. 1–15. doi: 10.1109/TIM.2022.3181903 (cit. on p. 126).

[CMS+22] Cheng, Bowen; Misra, Ishan; Schwing, Alexander G.; Kirillov, Alexander and Girdhar, Rohit: "Masked-attention Mask Transformer for Universal Image Segmentation". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA. June 18–24, 2022. 2022, pp. 1280–1289. doi: 10.1109/CVPR52688.2022.00135 (cit. on p. 103).

[CNF+15] Cupec, Robert; Nyarko, Emmanuel Karlo; Filko, Damir; Kitanov, Andrej and Petrović, Ivan: "Place recognition based on matching of planar surfaces and line segments". In: *The International Journal of Robotics Research* 34.4-5 (2015), pp. 674–704. doi: 10.1177/0278364914548708 (cit. on p. 123).

[COR+16] Cordts, Marius; Omran, Mohamed; Ramos, Sebastian; Rehfeld, Timo; Enzweiler, Markus; Benenson, Rodrigo; Franke, Uwe; Roth, Stefan and Schiele, Bernt: "The

Cityscapes Dataset for Semantic Urban Scene Understanding". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA. June 26–July 1, 2016. 2016, pp. 3213–3223. DOI: 10.1109/CVPR.2016.350 (cit. on pp. 32, 44).

[CPA23] Chalvatzaras, Athanasios; Pratikakis, Ioannis and Amanatiadis, Angelos A.: "A Survey on Map-Based Localization Techniques for Autonomous Vehicles". In: *IEEE Transactions on Intelligent Vehicles* 8.2 (2023), pp. 1574–1596. DOI: 10.1109/TIV.2022.3192102 (cit. on pp. 9, 11, 125).

[CPK19] Choy, Christopher; Park, Jaesik and Koltun, Vladlen: "Fully Convolutional Geometric Features". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. Seoul, Korea (South). Oct. 27–Nov. 2, 2019. 2019, pp. 8957–8965. DOI: 10.1109/ICCV.2019.00905 (cit. on pp. 122, 168).

[CRG+20] Cao, Bingyi; Ritter, Claas-Norman; Göhring, Daniel and Rojas, Raúl: "Accurate Localization of Autonomous Vehicles Based on Pattern Matching and Graph-Based Optimization in Urban Environments". In: *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. Rhodes, Greece. Sept. 20–23, 2020. 2020, pp. 1–6. DOI: 10.1109/ITSC45102.2020.9294299 (cit. on pp. 123, 125).

[CSY+22] Chen, Zhi; Sun, Kun; Yang, Fan and Tao, Wenbing: "SC2-PCR: A Second Order Spatial Compatibility for Efficient and Robust Point Cloud Registration". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA. June 18–24, 2022. 2022, pp. 13211–13221. DOI: 10.1109/CVPR52688.2022.01287 (cit. on pp. 122, 134, 162, 168, 169).

[CSY+23] Chen, Zhi; Sun, Kun; Yang, Fan; Guo, Lin and Tao, Wenbing: "$SC^2$2-PCR++: Rethinking the Generation and Selection for Efficient and Robust Point Cloud Registration". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*

45.10 (2023), pp. 12358–12376. DOI: 10.1109/TPAMI.2023.3272557 (cit. on pp. 122, 169).

[CT99] CASTELLANOS, José A. and TARDÓS, Juan D.: Mobile Robot Localization and Map Building: A Multisensor Fusion Approach. New York City, NY, USA: Springer Science+Business Media, 1999 (cit. on p. 123).

[CWL+20] CHEN, Changhao; WANG, Bing; LU, Chris Xiaoxuan; TRIGONI, Niki and MARKHAM, Andrew: A Survey on Deep Learning for Localization and Mapping: Towards the Age of Spatial Machine Intelligence. 2020. DOI: 10.48550/arXiv.2006.12567. arXiv: 2006.12567. URL: https://arxiv.org/abs/2006.12567 (last retrieved 2023-03-12) (cit. on p. 125).

[DCD+20] DUBÉ, Renaud; CRAMARIUC, Andrei; DUGAS, Daniel; SOMMER, Hannes; DYMCZYK, Marcin; NIETO, Juan; SIEGWART, Roland and CADENA, Cesar: "SegMap: Segment-based mapping and localization using data-driven descriptors". In: *The International Journal of Robotics Research* 39.2-3 (2020), pp. 339–355. DOI: 10.1177/0278364919863090 (cit. on p. 38).

[DCS+23] DONG, H.; CHEN, X.; SÄRKKÄ, S. and STACHNISS, C.: "Online pole segmentation on range images for long-term LiDAR localization in urban environments". In: *Robotics and Autonomous Systems* 159 (2023). Accessed on 2023-09-13 via arxiv at https://arxiv.org/abs/2208.07364, p. 104283. DOI: 10.1016/j.robot.2022.104283 (cit. on p. 37).

[DCS12] DIAZ-CABRERA, Moises; CERRI, Pietro and SANCHEZ-MEDINA, Javier: "Suspended traffic lights detection and distance estimation using color features". In: *2012 15th International IEEE Conference on Intelligent Transportation Systems*. Anchorage, AK, USA. Sept. 16–19, 2012. 2012, pp. 1315–1320. DOI: 10.1109/ITSC.2012.6338765 (cit. on p. 34).

[DDS+17] DUBÉ, Renaud; DUGAS, Daniel; STUMM, Elena; NIETO, Juan; SIEGWART, Roland and CADENA, Cesar: "SegMatch: Segment based place recognition in 3D point clouds". In: *2017 IEEE*

International Conference on Robotics and Automation (ICRA). Singapore, Singapore. May 29–June 3, 2017. 2017, pp. 5266–5272. DOI: 10.1109/ICRA.2017.7989618 (cit. on p. 38).

[Dem67] DEMPSTER, A. P.: "Upper and Lower Probabilities Induced by a Multivalued Mapping". In: *The Annals of Mathematical Statistics* 38.2 (1967), pp. 325–339. DOI: 10.1214/aoms/1177698950 (cit. on p. 191).

[Deu16] DEUSCH, Hendrik: "Random finite set-based localization and SLAM for highly automated vehicles". Doctoral Dissertation. Ulm, Germany: Universität Ulm, 2016. DOI: 10.18725/OPARU-4021 (cit. on p. 116).

[DFL19] DOHERTY, Kevin; FOURIE, Dehann and LEONARD, John: "Multimodal Semantic SLAM with Probabilistic Data Association". In: *2019 International Conference on Robotics and Automation (ICRA)*. Montreal, QC, Canada. May 20–24, 2019. 2019, pp. 2419–2425. DOI: 10.1109/ICRA.2019.8794244 (cit. on p. 40).

[DGP11] DEKEL, Yael; GUREL-GUREVICH, Ori and PERES, Yuval: "Finding Hidden Cliques in Linear Time with High Probability". In: *2011 Proceedings of the Workshop on Analytic Algorithmics and Combinatorics (ANALCO)*. San Francisco, CA, USA. Jan. 22, 2011. Society for Industrial and Applied Mathematics, 2011, pp. 67–75 (cit. on p. 304).

[Die17] DIESTEL, Reinhard: Graph Theory. 5th ed. Vol. 173. Graduate Texts in Mathematics. Berlin, Germany: Springer-Verlag, 2017 (cit. on p. 112).

[DJM+93] DUDEK, Gregory; JENKIN, Michael R. M.; MILIOS, Evangelos E. and WILKES, David: "Map Validation and Self-location in a Graph-like World". In: *Proceedings of the 13th International Joint Conference on Artificial Intelligence*. Chambéry, France. Aug. 28–Sept. 3, 1993. Ed. by BAJCSY, Ruzena. Morgan Kaufmann, 1993, pp. 1648–1653 (cit. on p. 192).

[DLB+15] Dymczyk, Marcin; Lynen, Simon; Bosse, Michael and Siegwart, Roland: "Keep it brief: Scalable creation of compressed localization maps". In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Hamburg, Germany. Sept. 28–Oct. 2, 2015. 2015, pp. 2536–2542. doi: 10.1109/IROS.2015.7353722 (cit. on p. 125).

[Dow72] Downs, Thomas D.: "Orientation Statistics". In: *Biometrika* 59.3 (1972), pp. 665–676 (cit. on p. 159).

[DPS20] Drost, Felix; Parolini, Luca and Schneider, Sebastian: "Siamese Networks for Online Map Validation in Autonomous Driving". In: *2020 IEEE Intelligent Vehicles Symposium (IV)*. Las Vegas, NV, USA. Oct. 19–Nov. 13, 2020. 2020, pp. 57–62. doi: 10.1109/IV47402.2020.9304642 (cit. on p. 194).

[DRD15] Deusch, Hendrik; Reuter, Stephan and Dietmayer, Klaus: "The Labeled Multi-Bernoulli SLAM Filter". In: *IEEE Signal Processing Letters* 22.10 (2015), pp. 1561–1565. doi: 10.1109/LSP.2015.2414274 (cit. on pp. 40, 116, 129, 292).

[DRS+15] Dierkes, Frank; Raaijmakers, Marvin; Schmidt, Max Theo; Bouzouraa, Mohamed Essayed; Hofmann, Ulrich and Maurer, Markus: "Towards a Multi-hypothesis Road Representation for Automated Driving". In: *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. Gran Canaria, Spain. Sept. 15–18, 2015. 2015, pp. 2497–2504. doi: 10.1109/ITSC.2015.402 (cit. on pp. 23, 195).

[DSG10] Dupuis, Marius; Strobl, Martin and Grezlikowski, Hans: "Opendrive 2010 and beyond–status and future of the de facto standard for the description of road networks". In: *Proc. of the Driving Simulation Conference Europe*. Paris, France. Sept. 9–10, 2010. 2010, pp. 231–242 (cit. on p. 9).

[Ebb17] Ebberg, Joern: A world first: Bosch creates a map that uses radar signals for automated driving. June 2017. url: https://www.bosch-presse.de/pressportal/de/en/a-world-first-bosch-

creates-a-map-that-uses-radar-signals-for-automated-driving-108544.html (last retrieved 2024-02-22) (cit. on p. 108).

[Ebb21] Ebberg, Joern: Swarm intelligence for automated driving. July 2021. url: https://www.bosch-presse.de/pressportal/de/en/swarm-intelligence-for-automated-driving-231431.html (last retrieved 2024-02-22) (cit. on p. 108).

[EFH+23] Elghazaly, Gamal; Frank, Raphaël; Harvey, Scott and Safko, Stefan: "High-Definition Maps: Comprehensive Survey, Challenges, and Future Perspectives". In: *IEEE Open Journal of Intelligent Transportation Systems* 4 (2023), pp. 527–550. doi: 10.1109/OJITS.2023.3295502 (cit. on pp. 8–11).

[EGW+04] Eren, T.; Goldenberg, O.K.; Whiteley, W.; Yang, Y.R.; Morse, A.S.; Anderson, B.D.O. and Belhumeur, P.N.: "Rigidity, computation, and randomization in network localization". In: *IEEE INFOCOM 2004*. Hong Kong, China. Mar. 7–11, 2004. Vol. 4. 2004, pp. 2673–2684. doi: 10.1109/INFCOM.2004.1354686 (cit. on pp. 131, 138, 141, 142).

[EKS+96] Ester, Martin; Kriegel, Hans-Peter; Sander, Jörg and Xu, Xiaowei: "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise". In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. Portland, OR, USA. Aug. 2–4, 1996. AAAI Press, 1996, pp. 226–231 (cit. on p. 52).

[EME+20] Esselborn, Carl; Misera, Leo; Eckert, Michael; Holzäpfel, Marc and Sax, Eric: "Map Attribute Validation using Historic Floating Car Data and Anomaly Detection Techniques". In: *Proceedings of the 6th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2020)*. Virtual Conference. May 2–4, 2020. SciTePress, 2020, pp. 504–514. doi: 10.5220/0009425905040514 (cit. on p. 192).

[Epp98] Eppstein, David: “Finding the k Shortest Paths”. In: *SIAM Journal on Computing* 28.2 (1998), pp. 652–673. doi: 10.1137/S0097539795290477 (cit. on p. 130).

[Fan21] Fang, Yu: “Localization in Possibly Outdated Semantic Urban Maps”. Master’s Thesis. Karlsruhe, Germany: Institute for Measurement and Control, Karlsruhe Institute of Technology, 2021 (cit. on pp. 30, 191).

[FB81] Fischler, Martin A. and Bolles, Robert C.: “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography”. In: *Commun. ACM* 24.6 (June 1981), pp. 381–395. doi: 10.1145/358669.358692 (cit. on pp. 120, 130).

[Feh21] Fehler, Richard: “Object-Conditioned Depth Estimation with Convolutional Neural Networks for Combining Complementary Datasets”. Master’s Thesis. Karlsruhe, Germany: Institute for Measurement and Control, Karlsruhe Institute of Technology, 2021 (cit. on pp. 33, 43).

[Fer19] Ferrari, Alberto: A Note on Sum and Difference of Correlated Chi-Squared Variables. 2019. arXiv: 1906.09982. url: https://arxiv.org/abs/1906.09982 (last retrieved 2023-06-05) (cit. on p. 311).

[FGS+17] Fatemi, Maryam; Granström, Karl; Svensson, Lennart; Ruiz, Francisco J. R. and Hammarstrand, Lars: “Poisson Multi-Bernoulli Mapping Using Gibbs Sampling”. In: *IEEE Transactions on Signal Processing* 65.11 (2017), pp. 2814–2827. doi: 10.1109/TSP.2017.2675866 (cit. on pp. 40, 129, 292).

[FGS+18] Fernández, Carlos; Guindel, Carlos; Salscheider, Niels-Ole and Stiller, Christoph: “A Deep Analysis of the Existing Datasets for Traffic Light State Recognition”. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. Maui, HI, USA. Nov. 4–7, 2018. 2018, pp. 248–254. doi: 10.1109/ITSC.2018.8569914 (cit. on pp. 32, 34).

[FKH18] Falkner, Stefan; Klein, Aaron and Hutter, Frank: "BOHB: Robust and Efficient Hyperparameter Optimization at Scale". In: *Proceedings of the 35th International Conference on Machine Learning*. Stockholm, Sweden. July 10–15, 2018. Ed. by Dy, Jennifer and Krause, Andreas. Vol. 80. Proceedings of Machine Learning Research. 2018, pp. 1437–1446. URL: https://proceedings.mlr.press/v80/falkner18a.html (last retrieved 2024-03-24) (cit. on p. 79).

[FMD17] Fregin, Andreas; Müller, Julian and Dietmayer, Klaus: "Three ways of using stereo vision for traffic light recognition". In: *2017 IEEE Intelligent Vehicles Symposium (IV)*. Los Angeles, CA, USA. June 11–14, 2017. 2017, pp. 430–436. DOI: 10.1109/IVS.2017.7995756 (cit. on p. 34).

[FMH+22] Fong, Whye Kit; Mohan, Rohit; Hurtado, Juana Valeria; Zhou, Lubing; Caesar, Holger; Beijbom, Oscar and Valada, Abhinav: "Panoptic Nuscenes: A Large-Scale Benchmark for LiDAR Panoptic Segmentation and Tracking". In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 3795–3802. DOI: 10.1109/LRA.2022.3148457 (cit. on p. 32).

[FMK+18] Fregin, Andreas; Muller, Julian; Krebel, Ulrich and Dietmayer, Klaus: "The DriveU Traffic Light Dataset: Introduction and Comparison with Existing Datasets". In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. Brisbane, QLD, Australia. May 21–25, 2018. 2018, pp. 3376–3383. DOI: 10.1109/ICRA.2018.8460737 (cit. on pp. 32, 34, 215).

[FSL+22] Fuji Tsang, Clement; Shugrina, Maria; Lafleche, Jean Francois; Takikawa, Towaki; Wang, Jiehan; Loop, Charles; Chen, Wenzheng; Jatavallabhula, Krishna Murthy; Smith, Edward; Rozantsev, Artem et al.: Kaolin: A Pytorch Library for Accelerating 3D Deep Learning Research. 2022. URL: https://github.com/NVIDIAGameWorks/kaolin (last retrieved 2023-10-11) (cit. on p. 104).

[FU11] Fairfield, Nathaniel and Urmson, Chris: "Traffic light mapping and detection". In: *2011 IEEE International Conference on Robotics and Automation*. Shanghai, China. May 9–13, 2011. 2011, pp. 5421–5426. doi: 10.1109/ICRA.2011.5980164 (cit. on p. 34).

[FVK+22] Forsgren, Brendon; Vasudevan, Ram; Kaess, Michael; McLain, Timothy W. and Mangelson, Joshua G.: "Group-k Consistent Measurement Set Maximization for Robust Outlier Detection". In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Kyoto, Japan. Oct. 23–27, 2022. 2022, pp. 4849–4856. doi: 10.1109/IROS47612.2022.9982057 (cit. on p. 119).

[GCL24] Gélis, Iris de; Corpetti, Thomas and Lefèvre, Sébastien: "Change Detection Needs Change Information: Improving Deep 3-D Point Cloud Change Detection". In: *IEEE Transactions on Geoscience and Remote Sensing* 62 (2024), pp. 1–10. doi: 10.1109/TGRS.2024.3359484 (cit. on p. 193).

[GE17] Gargoum, Suliman and El-Basyouny, Karim: "Automated extraction of road features using LiDAR data: A review of LiDAR applications in transportation". In: *2017 4th International Conference on Transportation Information and Safety (ICTIS)*. Banff, AB, Canada. Aug. 8–10, 2017. 2017, pp. 563–574. doi: 10.1109/ICTIS.2017.8047822 (cit. on p. 36).

[GEM+19] Ghallabi, Farouk; El-Haj-Shhade, Ghayath; Mittet, Marie-Anne and Nashashibi, Fawzi: "LIDAR-Based road signs detection For Vehicle Localization in an HD Map". In: *2019 IEEE Intelligent Vehicles Symposium (IV)*. Paris, France. June 9–12, 2019. 2019, pp. 1484–1490. doi: 10.1109/IVS.2019.8814029 (cit. on pp. 36, 50).

[GeoServer] Open Source Geospatial Foundation: GeoServer. url: http://geoserver.org/ (last retrieved 2023-03-24) (cit. on p. 17).

[GES+17] Gargoum, Suliman; El-Basyouny, Karim; Sabbagh, Joseph and Froese, Kenneth: "Automated Highway Sign Extraction Using Lidar Data". In: *Transportation Research Record* 2643.1 (2017), pp. 1–8. doi: 10.3141/2643-01 (cit. on pp. 36, 50).

[GEY+18] Geneva, Patrick; Eckenhoff, Kevin; Yang, Yulin and Huang, Guoquan: "LIPS: LiDAR-Inertial 3D Plane SLAM". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Madrid, Spain. Oct. 1–5, 2018. 2018, pp. 123–130. doi: 10.1109/IROS.2018.8594463 (cit. on p. 124).

[GLS+13] Geiger, A; Lenz, P; Stiller, C and Urtasun, R: "Vision meets robotics: The KITTI dataset". In: *The International Journal of Robotics Research* 32.11 (2013), pp. 1231–1237. doi: 10.1177/0278364913491297 (cit. on pp. 126, 152, 161, 168, 170, 218).

[GLU12] Geiger, Andreas; Lenz, Philip and Urtasun, Raquel: "Are we ready for autonomous driving? The KITTI vision benchmark suite". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. Providence, RI, USA. June 16–21, 2012. 2012, pp. 3354–3361. doi: 10.1109/CVPR.2012.6248074 (cit. on pp. 126, 152, 161, 168, 170, 218).

[GPS+17] Guo, Chuan; Pleiss, Geoff; Sun, Yu and Weinberger, Kilian Q.: "On Calibration of Modern Neural Networks". In: *Proceedings of the 34th International Conference on Machine Learning*. Sydney, NSW, Australia. Aug. 6–11, 2017. Ed. by Precup, Doina and Teh, Yee Whye. Vol. 70. Proceedings of Machine Learning Research. 2017, pp. 1321–1330. url: https://proceedings.mlr.press/v70/guo17a.html (last retrieved 2023-03-25) (cit. on p. 68).

[GRA+11] Gonzalez-Jorge, Higinio; Riveiro, Belen; Armesto, Julia and Arias, P.: "Geometric Evaluation of Road Signs Using Radiometric Information from Laser Scanning Data". In: *28th International Symposium on Automation and Robotics in Construction (ISARC 2011)*. Seoul, Korea (South). June 29–July 2,

2011. 2011, pp. 1007–1012. DOI: 10.22260/ISARC2011/0186 (cit. on p. 36).

[Gri90] GRIMSON, W. Eric L.: Object Recognition by Computer: The Role of Geometric Constraints. Cambridge, MA, USA: MIT Press, 1990 (cit. on pp. 112, 116, 118, 123, 130, 151).

[Gur22] GUROBI OPTIMIZATION, LLC: Gurobi Optimizer. Version 9.5.2. 2022. URL: https://www.gurobi.com (last retrieved 2024-03-24) (cit. on p. 147).

[Gut84] GUTTMAN, Antonin: "R-Trees: A Dynamic Index Structure for Spatial Searching". In: *SIGMOD Rec.* 14.2 (June 1984), pp. 47–57. DOI: 10.1145/971697.602266 (cit. on p. 48).

[GWG+18] GARCÍA-FERNÁNDEZ, Ángel F.; WILLIAMS, Jason L.; GRANSTRÖM, Karl and SVENSSON, Lennart: "Poisson Multi-Bernoulli Mixture Filter: Direct Derivation and Implementation". In: *IEEE Transactions on Aerospace and Electronic Systems* 54.4 (2018), pp. 1883–1901. DOI: 10.1109/TAES.2018.2805153 (cit. on p. 65).

[HA04] HOLM, Henrik and ALOUINI, Mohamed-Slim: "Sum and Difference of Two Squared Correlated Nakagami Variates in Connection with the McKay Distribution". In: *IEEE Transactions on Communications* 52.8 (2004), pp. 1367–1376 (cit. on pp. 307–309).

[HER17] HERE TECHNOLOGIES: HERE HD Live Map, The most intelligent sensor for autonomous driving. Tech. rep. Original URL accessed on 2021-07-19, last accessed state from 2021-12-14 via the Internet Archive on 2023-07-28 at https://web.archive.org/web/20211214133013/https://www.here.com/sites/g/files/odxslz166/files/2018-11/HERE_HD_Live_Map_one_pager.pdf. Eindhoven, Netherlands: HERE Technologies, 2017. URL: https://www.here.com/sites/g/files/odxslz166/files/2018-11/HERE_HD_Live_Map_one_pager.pdf (cit. on p. 9).

[Her18] Herrtwich, Ralf: The evolution of the HERE HD Live Map at Daimler. Feb. 2018. URL: https://www.here.com/learn/blog/the-evolution-of-the-hd-live-map (last retrieved 2023-09-28) (cit. on p. 8).

[HFE+16] Hauberg, Søren; Feragen, Aasa; Enficiaud, Raffi and Black, Michael J.: "Scalable Robust Principal Component Analysis Using Grassmann Averages". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.11 (2016), pp. 2298–2311. DOI: 10.1109/TPAMI.2015.2511743 (cit. on p. 55).

[HGD+17] He, Kaiming; Gkioxari, Georgia; Dollár, Piotr and Girshick, Ross: "Mask R-CNN". In: *2017 IEEE International Conference on Computer Vision (ICCV)*. Venice, Italy. Oct. 22–29, 2017. 2017, pp. 2980–2988. DOI: 10.1109/ICCV.2017.322 (cit. on p. 94).

[HGS+14a] Hartmann, Oliver; Gabb, Michael; Schüle, Florian; Schweiger, Roland and Dietmayer, Klaus: "Robust and real-time multi-cue map verification for the road ahead". In: *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Qingdao, China. Oct. 8–11, 2014. 2014, pp. 894–899. DOI: 10.1109/ITSC.2014.6957802 (cit. on p. 194).

[HGS+14b] Hartmann, Oliver; Gabb, Michael; Schweiger, Roland and Dietmayer, Klaus: "Towards autonomous self-assessment of digital maps". In: *2014 IEEE Intelligent Vehicles Symposium Proceedings*. Dearborn, MI, USA. June 8–11, 2014. 2014, pp. 89–95. DOI: 10.1109/IVS.2014.6856564 (cit. on p. 194).

[HGU+21] Huang, Shengyu; Gojcic, Zan; Usvyatsov, Mikhail; Wieser, Andreas and Schindler, Konrad: "PREDATOR: Registration of 3D Point Clouds with Low Overlap". In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Nashville, TN, USA. June 20–25, 2021. 2021,

pp. 4265–4274. DOI: 10.1109/CVPR46437.2021.00425 (cit. on pp. 121, 122, 168).

[HJL+22] He, Lei; Jiang, Shengjie; Liang, Xiaoqing; Wang, Ning and Song, Shiyu: "Diff-Net: Image Feature Difference Based High-Definition Map Change Detection for Autonomous Driving". In: *2022 International Conference on Robotics and Automation (ICRA)*. Philadelphia, PA, USA. May 23–27, 2022. 2022, pp. 2635–2641. DOI: 10.1109/ICRA46639.2022.9811573 (cit. on p. 194).

[HK19] Hsiao, Ming and Kaess, Michael: "MH-iSAM2: Multi-hypothesis iSAM using Bayes Tree and Hypo-tree". In: *2019 International Conference on Robotics and Automation (ICRA)*. Montreal, QC, Canada. May 20–24, 2019. 2019, pp. 1274–1280. DOI: 10.1109/ICRA.2019.8793854 (cit. on p. 40).

[HKK20] Heo, Minhyeok; Kim, Jiwon and Kim, Sujung: "HD Map Change Detection with Cross-Domain Deep Metric Learning". In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Las Vegas, NV, USA. Oct. 24, 2020–Jan. 24, 2021. 2020, pp. 10218–10224. DOI: 10.1109/IROS45743.2020.9340757 (cit. on p. 194).

[HMC] HERE Global B.V.: See how the map changes over one and a half years. URL: http://mapchanges.navigation.com/ (last retrieved 2023-09-27) (cit. on p. 10).

[HOL+23] Ha, Taeoh; Oh, Jeongwoo; Lee, Gunmin; Heo, Jaeseok; Kim, Do Hyung; Park, Byungkyu; Lee, Chang-Gun and Oh, Songhwai: "RIANet++: Road Graph and Image Attention Networks for Robust Urban Autonomous Driving Under Road Changes". In: *IEEE Robotics and Automation Letters* 8.11 (2023), pp. 7815–7822. DOI: 10.1109/LRA.2023.3320491 (cit. on p. 192).

[Hor87] Horn, Berthold K. P.: “Closed-form solution of absolute orientation using unit quaternions”. In: *Journal of the Optical Society of America A* 4.4 (Apr. 1987), pp. 629–642. doi: 10.1364/JOSAA.4.000629 (cit. on pp. 107, 155).

[HSR+20] Hofstetter, Isabell; Sprunk, Michael; Ries, Florian and Haueis, Martin: “Reliable Data Association for Feature-Based Vehicle Localization using Geometric Hashing Methods”. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Paris, France. May 31–Aug. 31, 2020. 2020, pp. 1322–1328. doi: 10.1109/ICRA40945.2020.9196601 (cit. on p. 108).

[HSS+19] Hofstetter, Isabell; Sprunk, Michael; Schuster, Frank; Ries, Florian and Haueis, Martin: “On Ambiguities in Feature-Based Vehicle Localization and their A Priori Detection in Maps”. In: *2019 IEEE Intelligent Vehicles Symposium (IV)*. Paris, France. June 9–12, 2019. 2019, pp. 1192–1198. doi: 10.1109/IVS.2019.8813978 (cit. on pp. 108, 113).

[HSS+22] Hvarfner, Carl; Stoll, Danny; Souza, Artur; Nardi, Luigi; Lindauer, Marius and Hutter, Frank: “πBO: Augmenting Acquisition Functions with User Beliefs for Bayesian Optimization”. In: *International Conference on Learning Representations*. Virtual Conference. Apr. 25–29, 2022. 2022. url: https://openreview.net/forum?id=MMAeCXIa89 (last retrieved 2024-03-24) (cit. on p. 80).

[HW77] Holland, Paul W. and Welsch, Roy E.: “Robust regression using iteratively reweighted least-squares”. In: *Communications in Statistics - Theory and Methods* 6.9 (1977), pp. 813–827. doi: 10.1080/03610927708827533 (cit. on pp. 157, 168).

[HZ04] Hartley, Richard and Zisserman, Andrew: Multiple View Geometry in Computer Vision. 2nd ed. Cambridge, Great Britain: Cambridge University Press, 2004. doi: 10.1017/CBO9780511811685 (cit. on p. 30).

[Jan22] Janosovits, Johannes: "Cityscapes TL++: Semantic Traffic Light Annotations for the Cityscapes Dataset". In: *2022 International Conference on Robotics and Automation (ICRA)*. Philadelphia, PA, USA. May 23–27, 2022. 2022, pp. 2569–2575. doi: 10.1109/ICRA46639.2022.9812144 (cit. on pp. 32, 34).

[JGB+20] Janai, Joel; Güney, Fatma; Behl, Aseem and Geiger, Andreas: "Computer Vision for Autonomous Vehicles: Problems, Datasets and State of the Art". In: *Foundations and Trends® in Computer Graphics and Vision* 12.1–3 (2020). Accessed on 2023-09-01 via arxiv at https://arxiv.org/abs/1704.05519, pp. 1–308. doi: 10.1561/0600000079 (cit. on pp. 32, 35, 215).

[JKS18] Jo, Kichun; Kim, Chansoo and Sunwoo, Myoungho: "Simultaneous Localization and Map Change Update for the High Definition Map-Based Autonomous Driving Car". In: *Sensors* 18.9 (2018). doi: 10.3390/s18093145 (cit. on pp. 10, 14, 190, 195, 196).

[JLC+23] Jain, Jitesh; Li, Jiachen; Chiu, Mang Tik; Hassani, Ali; Orlov, Nikita and Shi, Humphrey: "OneFormer: One Transformer To Rule Universal Image Segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Vancouver, BC, Canada. June 18–22, 2023. 2023, pp. 2989–2998 (cit. on p. 94).

[Jom20] Jomrich, Florian: "Dynamic Maps for Highly Automated Driving - Generation, Distribution and Provision". Doctoral Dissertation. Darmstadt, Germany: Technische Universität Darmstadt, Jan. 2020. doi: 10.25534/tuprints-00009702 (cit. on pp. 10, 195).

[JOSM] The Openstreetmap Project: JOSM. url: https://josm.openstreetmap.de/ (last retrieved 2023-03-24) (cit. on p. 17).

[JPM+16] Jensen, Morten Bornø; Philipsen, Mark Philip; Møgelmose, Andreas; Moeslund, Thomas Baltzer and Trivedi, Mohan

Manubhai: "Vision for Looking at Traffic Lights: Issues, Survey, and Perspectives". In: *IEEE Transactions on Intelligent Transportation Systems* 17.7 (2016), pp. 1800–1815. DOI: 10.1109/TITS.2015.2509509 (cit. on p. 34).

[JV87] JONKER, R. and VOLGENANT, A.: "A Shortest Augmenting Path Algorithm for Dense and Sparse Linear Assignment Problems". In: *Computing* 38.4 (1987), pp. 325–340 (cit. on pp. 64, 107, 130).

[Kar14] KARNEY, Charles: Implement Quaternion Fitting from the Todo List. 2014. URL: https://eigen.tuxfamily.org/bz/show%5C_bug.cgi?id=771 (last retrieved 2024-02-14) (cit. on p. 155).

[Kar72] KARP, Richard M.: "Reducibility among Combinatorial Problems". In: *Complexity of Computer Computations*. Yorktown Heights, NY, USA. Mar. 20–22, 1972. Ed. by MILLER, Raymond E.; THATCHER, James W. and BOHLINGER, Jean D. Boston, MA: Springer US, 1972, pp. 85–103. DOI: 10.1007/978-1-4684-2001-2_9 (cit. on p. 146).

[KBM+20] KATO, Hiroharu; BEKER, Deniz; MORARIU, Mihai; ANDO, Takahiro; MATSUOKA, Toru; KEHL, Wadim and GAIDON, Adrien: Differentiable Rendering: A Survey. 2020. DOI: 10.48550/arXiv.2006.12057. arXiv: 2006.12057. URL: https://arxiv.org/abs/2006.12057 (last retrieved 2023-10-11) (cit. on p. 104).

[KCS+21] KIM, Chansoo; CHO, Sungjin; SUNWOO, Myoungho; RESENDE, Paulo; BRADAÏ, Benazouz and JO, Kichun: "Updating Point Cloud Layer of High Definition (HD) Map Based on Crowd-Sourcing of Multiple Vehicles Installed LiDAR". In: *IEEE Access* 9 (2021), pp. 8028–8046. DOI: 10.1109/ACCESS.2021.3049482 (cit. on pp. 193, 194).

[KD19] Kraus, Florian and Dietmayer, Klaus: "Uncertainty Estimation in One-Stage Object Detection". In: *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. Auckland, New Zealand. Oct. 27–30, 2019. 2019, pp. 53–60. doi: 10.1109/ITSC.2019.8917494 (cit. on p. 68).

[KHK+22] Küppers, Fabian; Haselhoff, Anselm; Kronenberger, Jan and Schneider, Jonas: "Confidence Calibration for Object Detection and Segmentation". In: *Deep Neural Networks and Data for Automated Driving: Robustness, Uncertainty Quantification, and Insights Towards Safety*. Ed. by Fingscheidt, Tim; Gottschalk, Hanno and Houben, Sebastian. Cham, Switzerland: Springer International Publishing, 2022, pp. 225–250 (cit. on p. 68).

[KK19] Kühner, Tilman and Kümmerle, Julius: "Extrinsic Multi Sensor Calibration under Uncertainties". In: *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. Auckland, New Zealand. Oct. 27–30, 2019. 2019, pp. 3921–3927. doi: 10.1109/ITSC.2019.8917319 (cit. on p. 42).

[KK20] Kümmerle, Julius and Kühner, Tilman: "Unified Intrinsic and Extrinsic Camera and LiDAR Calibration under Uncertainties". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Paris, France. May 31–Aug. 31, 2020. 2020, pp. 6028–6034. doi: 10.1109/ICRA40945.2020.9197496 (cit. on p. 42).

[KKL18] Kümmerle, Julius; Kühner, Tilman and Lauer, Martin: "Automatic Calibration of Multiple Cameras and Depth Sensors with a Spherical Target". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Madrid, Spain. Oct. 1–5, 2018. 2018, pp. 1–8. doi: 10.1109/IROS.2018.8593955 (cit. on p. 42).

[KKS+20] Küppers, Fabian; Kronenberger, Jan; Shantia, Amirhossein and Haselhoff, Anselm: "Multivariate Confidence Calibration for Object Detection". In: *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. Virtual Conference. June 14–19, 2020. 2020, pp. 326–327 (cit. on p. 68).

[Kla15] Klar, Bernhard: "A Note on Gamma Difference Distributions". In: *Journal of Statistical Computation and Simulation* 85.18 (2015), pp. 3708–3715 (cit. on pp. 307, 308).

[KM77] Khatri, C. G. and Mardia, K. V.: "The Von Mises-Fisher Matrix Distribution in Orientation Statistics". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 39.1 (1977), pp. 95–106 (cit. on p. 159).

[KSP+19] Kümmerle, Julius; Sons, Marc; Poggenhans, Fabian; Kühner, Tilman; Lauer, Martin and Stiller, Christoph: "Accurate and Efficient Self-Localization on Roads using Basic Geometric Primitives". In: *2019 International Conference on Robotics and Automation (ICRA)*. Montreal, QC, Canada. May 20–24, 2019. 2019, pp. 5965–5971. doi: 10.1109/ICRA.2019.8793497 (cit. on p. 38).

[Kuh55] Kuhn, Harold W.: "The Hungarian Method for the Assignment Problem". In: *Naval Research Logistics Quarterly* 2.1-2 (May 1955), pp. 83–97. doi: 10.1002/nav.3800020109 (cit. on pp. 64, 74).

[Küm20] Kümmerle, Julius Valentin: "Multimodal Sensor Calibration with a Spherical Calibration Target". Doctoral Dissertation. Karlsruhe, Germany: Karlsruher Institut für Technologie (KIT), 2020. 185 pp. doi: 10.5445/IR/1000124721 (cit. on p. 42).

[Küp23] Küppers, Fabian: Uncertainty Calibration and its Application to Object Detection. 2023. doi: 10.48550/arXiv.2302.02622. arXiv: 2302.02622. url: https://arxiv.org/abs/2302.02622v1 (last retrieved 2023-03-19) (cit. on p. 68).

[LAD+11] Levinson, Jesse; Askeland, Jake; Dolson, Jennifer and Thrun, Sebastian: "Traffic light mapping, localization, and state detection for autonomous vehicles". In: *2011 IEEE International Conference on Robotics and Automation*. Shanghai, China. May 9–13, 2011. 2011, pp. 5784–5791. doi: 10.1109/ICRA.2011.5979714 (cit. on p. 34).

[LEF+22] Lindauer, Marius; Eggensperger, Katharina; Feurer, Matthias; Biedenkapp, André; Deng, Difan; Benjamins, Carolin; Ruhkopf, Tim; Sass, René and Hutter, Frank: "SMAC3: A Versatile Bayesian Optimization Package for Hyperparameter Optimization". In: *Journal of Machine Learning Research* 23.54 (2022), pp. 1–9. url: http://jmlr.org/papers/v23/21-0888.html (last retrieved 2024-03-24) (cit. on pp. 78, 313).

[LFH21] Lusk, Parker C.; Fathian, Kaveh and How, Jonathan P.: "CLIPPER: A Graph-Theoretic Framework for Robust Data Association". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. Xi'an, China. May 30–June 5, 2021. 2021, pp. 13828–13834. doi: 10.1109/ICRA48506.2021.9561069 (cit. on pp. 119, 131, 151, 162).

[LGU15] Lenz, Philip; Geiger, Andreas and Urtasun, Raquel: "FollowMe: Efficient Online Min-Cost Flow Tracking with Bounded Memory and Computation". In: *2015 IEEE International Conference on Computer Vision (ICCV)*. Santiago, Chile. Dec. 7–13, 2015. IEEE, 2015, pp. 4364–4372. doi: 10.1109/ICCV.2015.496 (cit. on p. 65).

[LH05] Leordeanu, M. and Hebert, M.: "A spectral technique for correspondence problems using pairwise constraints". In: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Beijing, China. Oct. 17–21, 2005. Vol. 2. 2005, pp. 1482–1489. doi: 10.1109/ICCV.2005.20 (cit. on pp. 119, 131, 151).

[LH21] Lambert, John and Hays, James: "Trust, but Verify: Cross-Modality Fusion for HD Map Change Detection". In: *Advances in Neural Information Processing Systems Track on Datasets and Benchmarks*. New Orleans, LA, USA. Nov. 28–Dec. 9, 2022. 2021. url: https://openreview.net/forum?id=cXCZnLjDm4s (last retrieved 2024-03-24) (cit. on p. 194).

[LH22] Lusk, Parker C. and How, Jonathan P.: "Global Data Association for SLAM with 3D Grassmannian Manifold Objects". In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Kyoto, Japan. Oct. 23–27, 2022. 2022, pp. 4463–4470. doi: 10.1109/IROS47612.2022.9981075 (cit. on pp. 39, 112, 124, 152, 153, 156).

[LJD+18] Li, Lisha; Jamieson, Kevin; DeSalvo, Giulia; Rostamizadeh, Afshin and Talwalkar, Ameet: "Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization". In: *Journal of Machine Learning Research* 18.185 (2018), pp. 1–52. url: http://jmlr.org/papers/v18/16-558.html (last retrieved 2024-03-24) (cit. on p. 79).

[LKC+21] Lee, Junha; Kim, Seungwook; Cho, Minsu and Park, Jaesik: "Deep Hough Voting for Robust Global Registration". In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. Montreal, QC, Canada. Oct. 10–17, 2021. 2021, pp. 15994–16003. doi: 10.1109/ICCV48922.2021.01569 (cit. on p. 169).

[LLC+19] Liu, Chunsheng; Li, Shuang; Chang, Faliang and Wang, Yinhai: "Machine Vision Based Traffic Sign Detection Methods: Review, Analyses and Perspectives". In: *IEEE Access* 7 (2019), pp. 86578–86596. doi: 10.1109/ACCESS.2019.2924947 (cit. on pp. 32, 35, 215).

[LMB+14] Lin, Tsung-Yi; Maire, Michael; Belongie, Serge J.; Hays, James; Perona, Pietro; Ramanan, Deva; Dollár, Piotr and Zitnick, C. Lawrence: "Microsoft COCO: Common Objects in Context". In: *Computer Vision - ECCV 2014 - 13th European*

Conference. Zurich, Switzerland. Sept. 6–12, 2014. Ed. by Fleet, David J.; Pajdla, Tomás; Schiele, Bernt and Tuytelaars, Tinne. Vol. 8693. Lecture Notes in Computer Science. Springer International Publishing, 2014, pp. 740–755. doi: 10.1007/978-3-319-10602-1_48 (cit. on p. 32).

[LMK22] Lin, Muyuan; Murali, Varun and Karaman, Sertac: "A Planted Clique Perspective on Hypothesis Pruning". In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 5167–5174. doi: 10.1109/LRA.2022.3155198 (cit. on p. 303).

[LNT10] Li, Hao; Nashashibi, Fawzi and Toulminet, Gwenaelle: "Localization for intelligent vehicle by fusing mono-camera, low-cost GPS and map data". In: *13th International IEEE Conference on Intelligent Transportation Systems*. Funchal, Portugal. Sept. 19–22, 2010. 2010, pp. 1657–1662. doi: 10.1109/ITSC.2010.5625240 (cit. on p. 35).

[LO12] Li, Yangming and Olson, Edwin B.: "IPJC: The Incremental Posterior Joint Compatibility test for fast feature cloud matching". In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Vilamoura-Algarve, Portugal. Oct. 7–12, 2012. 2012, pp. 3467–3474. doi: 10.1109/IROS.2012.6385470 (cit. on p. 116).

[LPH23] Lusk, Parker C.; Parikh, Devarth and How, Jonathan P.: "GraffMatch: Global Matching of 3D Lines and Planes for Wide Baseline LiDAR Registration". In: *IEEE Robotics and Automation Letters* 8.2 (2023), pp. 632–639. doi: 10.1109/LRA.2022.3229224 (cit. on pp. 112, 124, 126, 152, 153).

[LS14] Lategahn, Henning and Stiller, Christoph: "Vision-Only Localization". In: *IEEE Transactions on Intelligent Transportation Systems* 15.3 (2014), pp. 1246–1257. doi: 10.1109/TITS.2014.2298492 (cit. on p. 125).

[LWZ20] Liu, Rong; Wang, Jinling and Zhang, Bingqi: "High Definition Map for Automated Driving: Overview and Analysis".

In: *The Journal of Navigation* 73.2 (2020), pp. 324–341. DOI: 10.1017/S0373463319000638 (cit. on pp. 8, 9, 11).

[LXG23] Liao, Y.; Xie, J. and Geiger, A.: "KITTI-360: A Novel Dataset and Benchmarks for Urban Scene Understanding in 2D and 3D". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.3 (2023), pp. 3292–3310. DOI: 10.1109/TPAMI.2022.3179507 (cit. on p. 32).

[LYL+23] Li, Liang; Yang, Ming; Li, Hao; Wang, Chunxiang and Wang, Bing: "Robust Localization for Intelligent Vehicles Based on Compressed Road Scene Map in Urban Environments". In: *IEEE Transactions on Intelligent Vehicles* 8.1 (2023), pp. 250–262. DOI: 10.1109/TIV.2022.3162845 (cit. on p. 125).

[LZW+19] Lu, Weixin; Zhou, Yao; Wan, Guowei; Hou, Shenhua and Song, Shiyu: "L3-Net: Towards Learning Based LiDAR Localization for Autonomous Driving". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA. June 15–20, 2019. 2019, pp. 6382–6391. DOI: 10.1109/CVPR.2019.00655 (cit. on p. 125).

[Mah07] Mahler, Ronald P. S.: Statistical Multisource-Multitarget Information Fusion. Norwood, MA, USA: Artech House, Inc., 2007 (cit. on pp. 112, 128, 129, 134, 135).

[Mah14] Mahler, Ronald P. S.: Advances in Statistical Multisource-Multitarget Information Fusion. Norwood, MA, USA: Artech House, Inc., 2014 (cit. on pp. 112, 134).

[Mat76] Matula, David W.: The Largest Clique Size in a Random Graph. Tech. rep. Department of Computer Science, Southern Methodist University Dallas, TX, USA, 1976 (cit. on p. 303).

[Mat93] Mathai, Arakaparampil M.: "On Noncentral Generalized Laplacianness of Quadratic Forms in Normal Variables". In: *Journal of Multivariate Analysis* 45.2 (1993), pp. 239–246 (cit. on pp. 307, 308).

[MBA23] Maierhofer, Sebastian; Ballnath, Yannick and Althoff, Matthias: "Map Verification and Repairing Using Formalized Map Specifications". In: *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*. Bilbao, Spain. Sept. 24–28, 2023. 2023, pp. 1277–1284. doi: 10.1109/ITSC57777.2023.10422044 (cit. on p. 195).

[MBB+16] Mühlfellner, Peter; Bürki, Mathias; Bosse, Michael; Derendarz, Wojciech; Philippsen, Roland and Furgale, Paul: "Summary Maps for Lifelong Visual Localization". In: *Journal of Field Robotics* 33.5 (2016), pp. 561–590. doi: 10.1002/rob.21595 (cit. on p. 125).

[MCC+07] Markley, F. Landis; Cheng, Yang; Crassidis, John L. and Oshman, Yaakov: "Averaging Quaternions". In: *Journal of Guidance, Control, and Dynamics* 30.4 (2007), pp. 1193–1197. doi: 10.2514/1.28949 (cit. on p. 72).

[McK32] McKay, A. T.: "A Bessel Function Distribution". In: *Biometrika* 24.1/2 (1932), pp. 39–44 (cit. on p. 307).

[MD18] Müller, Julian and Dietmayer, Klaus: "Detecting Traffic Lights by Single Shot Detection". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. Maui, HI, USA. Nov. 4–7, 2018. 2018, pp. 266–273. doi: 10.1109/ITSC.2018.8569683 (cit. on p. 34).

[MDE+18] Mangelson, Joshua G.; Dominic, Derrick; Eustice, Ryan M. and Vasudevan, Ram: "Pairwise Consistent Measurement Set Maximization for Robust Multi-Robot Map Merging". In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. Brisbane, QLD, Australia. May 21–25, 2018. 2018, pp. 2916–2923. doi: 10.1109/ICRA.2018.8460217 (cit. on pp. 118, 130).

[Mey23] Meyer, Annika: "Echtzeitfähige Schätzung generischer Linienzüge für das kartenlose automatisierte Fahren mittels Deep

Learning". Doctoral Dissertation. Karlsruhe, Germany: Karlsruher Institut für Technologie (KIT), 2023. 129 pp. DOI: 10.5445/IR/1000161444 (cit. on p. 103).

[MFD17] MÜLLER, Julian; FREGIN, Andreas and DIETMAYER, Klaus: "Multi-camera system for traffic light detection: About camera setup and mapping of detections". In: *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. Yokohama, Japan. Oct. 16–19, 2017. 2017, pp. 165–172. DOI: 10.1109/ITSC.2017.8317946 (cit. on p. 34).

[MJ00] MARDIA, Kantilal V. and JUPP, Peter E.: Directional Statistics. New ed. Wiley series in probability and statistics. Chichester, Great Britain: Wiley, 2000. DOI: 10.1002/9780470316979 (cit. on pp. 159, 160).

[MKM08] MEUTER, Mirko; KUMMERT, Anton and MULLER-SCHNEIDERS, Stefan: "3D Traffic Sign Tracking Using a Particle Filter". In: *2008 11th International IEEE Conference on Intelligent Transportation Systems*. Beijing, China. Oct. 12–15, 2008. 2008, pp. 168–173. DOI: 10.1109/ITSC.2008.4732525 (cit. on p. 35).

[MLP+16] MU, Beipeng; LIU, Shih-Yuan; PAULL, Liam; LEONARD, John and HOW, Jonathan P.: "SLAM with objects using a nonparametric pose graph". In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Daejeon, Korea (South). Oct. 9–14, 2016. 2016, pp. 4602–4609. DOI: 10.1109/IROS.2016.7759677 (cit. on pp. 39, 40).

[MP92] MATHAI, Arakaparampil M. and PROVOST, Serge B.: Quadratic Forms in Random Variables: Theory and Applications. Vol. 126. Statistics, Textbooks and Monographs. New York City, NY, USA; Basel, Switzerland; Hong Kong, China: Marcel Dekker, 1992 (cit. on p. 306).

[MPH+22a] MUÑOZ-BAÑÓN, Miguel Ángel; PAULS, Jan-Hendrik; HU, Haohao and STILLER, Christoph: "DA-LMR: A Robust Lane

Marking Representation for Data Association". In: *2022 International Conference on Robotics and Automation (ICRA)*. Philadelphia, PA, USA. May 23–27, 2022. 2022, pp. 2193–2199. DOI: 10.1109/ICRA46639.2022.9812271 (cit. on pp. 112, 121, 124, 186).

[MPH+22b] Muñoz-Bañón, Miguel Ángel; Pauls, Jan-Hendrik; Hu, Haohao; Stiller, Christoph; Candelas, Francisco A. and Torres, Fernando: "Robust Self-Tuning Data Association for Geo-Referencing Using Lane Markings". In: *IEEE Robotics and Automation Letters* 7.4 (2022), pp. 12339–12346. DOI: 10.1109/LRA.2022.3216991 (cit. on pp. 112, 124, 186).

[MPL+17] Maddern, Will; Pascoe, Geoffrey; Linegar, Chris and Newman, Paul: "1 year, 1000 km: The Oxford RobotCar dataset". In: *The International Journal of Robotics Research* 36.1 (2017), pp. 3–15. DOI: 10.1177/0278364916679498 (cit. on p. 33).

[MSP+21] Meyer, Annika; Skudlik, Philipp; Pauls, Jan-Hendrik and Stiller, Christoph: "YOLinO: Generic Single Shot Polyline Detection in Real Time". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*. Virtual Conference. Oct. 11–17, 2021. 2021, pp. 2916–2925. DOI: 10.1109/ICCVW54120.2021.00326 (cit. on p. 103).

[MT03] Montemerlo, Michael and Thrun, Sebastian: "Simultaneous localization and mapping with unknown data association using FastSLAM". In: *2003 IEEE International Conference on Robotics and Automation*. Taipei, Taiwan. Sept. 14–19, 2003. Vol. 2. 2003, 1985–1991 vol.2. DOI: 10.1109/ROBOT.2003.1241885 (cit. on p. 115).

[MTK+02] Montemerlo, Michael; Thrun, Sebastian; Koller, Daphne and Wegbreit, Ben: "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem". In: *Eighteenth National Conference on Artificial Intelligence*. Edmonton, AB, Canada. July 28–Aug. 1, 2002. American

Association for Artificial Intelligence, 2002, pp. 593–598 (cit. on p. 115).

[Mun57] Munkres, James: "Algorithms for the Assignment and Transportation Problems". In: *Journal of the Society for Industrial and Applied Mathematics* 5.1 (1957), pp. 32–38 (cit. on pp. 64, 107, 130).

[Mur68] Murty, Katta G.: "Letter to the Editor—An Algorithm for Ranking all the Assignments in Order of Increasing Cost". In: *Operations Research* 16.3 (1968), pp. 682–687. doi: 10.1287/opre.16.3.682 (cit. on p. 130).

[MVA+11] Mullane, John; Vo, Ba-Ngu; Adams, Martin D. and Vo, Ba-Tuong: "A Random-Finite-Set Approach to Bayesian SLAM". In: *IEEE Transactions on Robotics* 27.2 (2011), pp. 268–282. doi: 10.1109/TRO.2010.2101370 (cit. on pp. 40, 116, 129, 292).

[NBW19] Naujoks, Benjamin; Burger, Patrick and Wuensche, Hans-Joachim: "Combining Deep Learning and Model-Based Methods for Robust Real-Time Semantic Landmark Detection". In: *2019 22th International Conference on Information Fusion (FUSION)*. Ottawa, ON, Canada. July 2–5, 2019. 2019, pp. 1–8. doi: 10.23919/FUSION43075.2019.9011403 (cit. on pp. 38, 40).

[NGZ09] Nienhuser, Dennis; Gumpp, Thomas and Zollner, J. Marius: "A Situation Context Aware Dempster-Shafer Fusion of Digital Maps and a Road Sign Recognition System". In: *2009 IEEE Intelligent Vehicles Symposium*. Xi'an, China. June 3–5, 2009. 2009, pp. 1401–1406 (cit. on p. 22).

[NMS19] Nicholson, Lachlan; Milford, Michael and Sünderhauf, Niko: "QuadricSLAM: Dual Quadrics From Object Detections as Landmarks in Object-Oriented SLAM". In: *IEEE Robotics and Automation Letters* 4.1 (2019), pp. 1–8. doi: 10.1109/LRA.2018.2866205 (cit. on p. 39).

[NOB+17] Neuhold, Gerhard; Ollmann, Tobias; Bulò, Samuel Rota and Kontschieder, Peter: "The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes". In: *2017 IEEE International Conference on Computer Vision (ICCV)*. Venice, Italy. Oct. 22–29, 2017. 2017, pp. 5000–5009. doi: 10.1109/ICCV.2017.534 (cit. on pp. 32, 43, 94).

[NSU+16] Nguyen, Tran Tuan; Spehr, Jens; Uhlemann, Matthias; Zug, Sebastian and Kruse, Rudolf: "Learning of lane information reliability for intelligent vehicles". In: *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. Baden-Baden, Germany. Sept. 19–21, 2016. 2016, pp. 142–147. doi: 10.1109/MFI.2016.7849480 (cit. on p. 195).

[NSV+18] Nguyen, Tran Tuan; Spehr, Jens; Vock, Dominik; Baum, Marcus; Zug, Sebastian and Kruse, Rudolf: "A General Reliability-Aware Fusion Concept Using DST and Supervised Learning with Its Applications in Multi-Source Road Estimation". In: *2018 IEEE Intelligent Vehicles Symposium (IV)*. Changshu, China. June 26–30, 2018. 2018, pp. 597–604. doi: 10.1109/IVS.2018.8500713 (cit. on p. 195).

[NT01] Neira, J. and Tardos, J.D.: "Data association in stochastic mapping using the joint compatibility test". In: *IEEE Transactions on Robotics and Automation* 17.6 (2001), pp. 890–897. doi: 10.1109/70.976019 (cit. on p. 116).

[NTC03] Neira, J.; Tardos, J.D. and Castellanos, J.A.: "Linear time vehicle relocation in SLAM". In: *2003 IEEE International Conference on Robotics and Automation*. Taipei, Taiwan. Sept. 14–19, 2003. Vol. 1. 2003, pp. 427–433. doi: 10.1109/ROBOT.2003.1241632 (cit. on pp. 118, 120, 130, 131, 151).

[OLF+19] Ok, Kyel; Liu, Katherine; Frey, Kris; How, Jonathan P. and Roy, Nicholas: "Robust Object-based SLAM for High-speed Autonomous Navigation". In: *2019 International Conference on Robotics and Automation (ICRA)*. Montreal, QC, Canada.

May 20–24, 2019. 2019, pp. 669–675. DOI: 10.1109/ICRA.2019.8794344 (cit. on p. 39).

[PBC+19] Porzi, Lorenzo; Bulò, Samuel Rota; Colovic, Aleksander and Kontschieder, Peter: "Seamless Scene Segmentation". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA. June 16–20, 2019. 2019, pp. 8269–8278. DOI: 10.1109/CVPR.2019.00847 (cit. on pp. 43, 74, 94).

[PBV+10] Pathak, Kaustubh; Birk, Andreas; Vaškevičius, Narunas and Poppinga, Jann: "Fast Registration Based on Noisy Planes With Unknown Correspondences for 3-D Mapping". In: *IEEE Transactions on Robotics* 26.3 (2010), pp. 424–441 (cit. on p. 123).

[PFL+22] Pauls, Jan-Hendrik; Fehler, Richard; Lauer, Martin and Stiller, Christoph: "Combining 2D and 3D Datasets with Object-Conditioned Depth Estimation". In: *2022 IEEE Intelligent Vehicles Symposium (IV)*. Aachen, Germany. June 4–9, 2022. 2022, pp. 1194–1200. DOI: 10.1109/IV51971.2022.9827425 (cit. on p. 33).

[PGC+19] Possatti, Lucas C.; Guidolini, Rânik; Cardoso, Vinicius B.; Berriel, Rodrigo F.; Paixão, Thiago M.; Badue, Claudine; De Souza, Alberto F. and Oliveira-Santos, Thiago: "Traffic Light Recognition Using Deep Learning and Prior Maps for Autonomous Cars". In: *2019 International Joint Conference on Neural Networks (IJCNN)*. Budapest, Hungary. July 14–19, 2019. 2019, pp. 1–8. DOI: 10.1109/IJCNN.2019.8851927 (cit. on pp. 37, 38).

[PLB+23] Pavlitska, Svetlana; Lambing, Nico; Bangaru, Ashok Kumar and Zöllner, J Marius: "Traffic Light Recognition using Convolutional Neural Networks: A Survey". In: *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*. Bilbao, Spain. Sept. 24–28, 2023. 2023, pp. 2790–2796 (cit. on p. 34).

[PLB19] Pannen, David; Liebner, Martin and Burgard, Wolfram: "HD Map Change Detection with a Boosted Particle Filter". In: *2019 International Conference on Robotics and Automation (ICRA)*. Montreal, QC, Canada. May 20–24, 2019. 2019, pp. 2561–2567. doi: 10.1109/ICRA.2019.8794329 (cit. on p. 193).

[PLH+20] Pannen, David; Liebner, Martin; Hempel, Wolfgang and Burgard, Wolfram: "How to Keep HD Maps for Automated Driving Up To Date". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Paris, France (virtual). May 31–Aug. 31, 2020. 2020, pp. 2288–2294. doi: 10.1109/ICRA40945.2020.9197419 (cit. on p. 193).

[PMF+20] Plachetka, Christopher; Maier, Niels; Fricke, Jenny; Termöhlen, Jan-Aike and Fingscheidt, Tim: "Terminology and Analysis of Map Deviations in Urban Domains: Towards Dependability for HD Maps in Automated Vehicles". In: *2020 IEEE Intelligent Vehicles Symposium (IV)*. Las Vegas, NV, USA. Oct. 19–Nov. 13, 2020. 2020, pp. 63–70. doi: 10.1109/IV47402.2020.9304580 (cit. on pp. 9, 10, 13, 15).

[Pog19] Poggenhans, Fabian: "Generierung hochdetaillierter Karten für das automatisierte Fahren". Doctoral Dissertation. Karlsruhe, Germany: Karlsruher Institut für Technologie (KIT), 2019. 107 pp. doi: 10.5445/IR/1000100719 (cit. on p. 213).

[PP18] Pfeifer, Tim and Protzel, Peter: "Robust Sensor Fusion with Self-Tuning Mixture Models". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Madrid, Spain. Oct. 1–5, 2018. 2018, pp. 3678–3685. doi: 10.1109/IROS.2018.8594459 (cit. on p. 40).

[PPJ+18] Poggenhans, Fabian; Pauls, Jan-Hendrik; Janosovits, Johannes; Orf, Stefan; Naumann, Maximilian; Kuhnt, Florian and Mayr, Matthias: "Lanelet2: A high-definition map framework for the future of automated driving". In: *2018 21st International Conference on Intelligent Transportation Systems*

(ITSC). Maui, HI, USA. Nov. 4–7, 2018. 2018, pp. 1672–1679. DOI: 10.1109/ITSC.2018.8569929 (cit. on pp. 8, 9, 11, 12, 190, 191, 213, 219).

[PPP+20] PAULS, Jan-Hendrik; PETEK, Kürsat; POGGENHANS, Fabian and STILLER, Christoph: "Monocular Localization in HD Maps by Combining Semantic Segmentation and Distance Transform". In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Virtual Conference. Oct. 24, 2020–Jan. 24, 2021. 2020, pp. 4595–4601. DOI: 10.1109/IROS45743.2020.9341003 (cit. on p. 9).

[PS22] PAULS, Jan-Hendrik and STILLER, Christoph: "Aktualitätsverifikation semantischer HD-Karten für das urbane automatisierte Fahren". In: *14. Workshop Fahrerassistenz und automatisiertes Fahren*. Berkheim, Germany. May 9–11, 2022. 2022, pp. 47–56 (cit. on pp. 30, 112, 114, 118, 191).

[PSC+18] PHAN, Buu; SALAY, Rick; CZARNECKI, Krzysztof; ABDELZAD, Vahdat; DENOUDEN, Taylor and VERNEKAR, Sachin: Calibrating Uncertainties in Object Localization Task. Dec. 2018. DOI: 10.48550/arXiv.1811.11210. arXiv: 1811.11210. URL: http://arxiv.org/abs/1811.11210 (last retrieved 2023-03-19) (cit. on p. 68).

[PSF+22] PLACHETKA, Christopher; SERTOLLI, Benjamin; FRICKE, Jenny; KLINGNER, Marvin and FINGSCHEIDT, Tim: "3DHD CityScenes: High-Definition Maps in High-Density Point Clouds". In: *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. Macau, China. Oct. 8–12, 2022. 2022, pp. 627–634. DOI: 10.1109/ITSC55140.2022.9921866 (cit. on pp. 32, 36, 37, 39, 102, 214).

[PSF+23] PLACHETKA, Christopher; SERTOLLI, Benjamin; FRICKE, Jenny; KLINGNER, Marvin and FINGSCHEIDT, Tim: "DNN-Based Map Deviation Detection in LiDAR Point Clouds". In: *IEEE Open Journal of Intelligent Transportation Systems* 4 (2023),

pp. 580–601. DOI: 10.1109/OJITS.2023.3293911 (cit. on pp. 10, 36, 37, 39, 101, 102, 195, 200, 214).

[PSH+18] Pauls, Jan-Hendrik; Strauss, Tobias; Hasberg, Carsten; Lauer, Martin and Stiller, Christoph: "Can We Trust Our Maps? An Evaluation of Road Changes and a Dataset for Map Validation". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. Maui, HI, USA. Nov. 4–7, 2018. 2018, pp. 2639–2644. DOI: 10.1109/ITSC.2018.8569249 (cit. on pp. 8, 24).

[PSH+20a] Pauls, Jan-Hendrik; Strauss, Tobias; Hasberg, Carsten; Lauer, Martin and Stiller, Christoph: "HD Map Verification Without Accurate Localization Prior Using Spatio-Semantic 1D Signals". In: *2020 IEEE Intelligent Vehicles Symposium (IV)*. Las Vegas, NV, USA. Oct. 19–Nov. 13, 2020. 2020, pp. 680–686. DOI: 10.1109/IV47402.2020.9304716 (cit. on pp. 24, 109, 192).

[PSH+20b] Pauls, Jan-Hendrik; Strauss, Tobias; Hasberg, Carsten and Stiller, Christoph: "Verifikation von HD-Karten mittels räumlich und semantisch separierbarer 1D-Signale". In: *13. Workshop Fahrerassistenz und automatisiertes Fahren*. Virtual Workshop. July 16–17, 2020. 2020, pp. 84–92 (cit. on pp. 24, 109, 192).

[PSH+21] Pauls, Jan-Hendrik; Strauss, Tobias; Hasberg, Carsten and Stiller, Christoph: "Boosted Classifiers on 1D Signals and Mutual Evaluation of Independently Aligned Spatio-Semantic Feature Groups for HD Map Change Detection". In: *2021 IEEE Intelligent Vehicles Symposium (IV)*. Nagoya, Japan. July 11–17, 2021. 2021, pp. 961–966. DOI: 10.1109/IV48863.2021.9575778 (cit. on pp. 24, 109, 192).

[PSS21] Pauls, Jan-Hendrik; Schmidt, Benjamin and Stiller, Christoph: "Automatic Mapping of Tailored Landmark Representations for Automated Driving and Map Learning". In: *2021 IEEE International Conference on Robotics and*

Automation (ICRA). Xi'an, China. May 30–June 5, 2021. 2021, pp. 6725–6731. DOI: 10.1109/ICRA48506.2021.9561432 (cit. on pp. 30, 33, 38, 40, 51, 64, 71, 101, 102).

[PVG+11] PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V. et al.: "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830. URL: http://jmlr.org/papers/v12/pedregosa11a.html (last retrieved 2023-03-25) (cit. on p. 291).

[QY20] QUAN, Siwen and YANG, Jiaqi: "Compatibility-Guided Sampling Consensus for 3-D Point Cloud Registration". In: *IEEE Transactions on Geoscience and Remote Sensing* 58.10 (2020), pp. 7380–7392. DOI: 10.1109/TGRS.2020.2982221 (cit. on pp. 121, 169).

[QYW+22] QIN, Zheng; YU, Hao; WANG, Changiian; GUO, Yulan; PENG, Yuxing and XU, Kai: "Geometric Transformer for Fast and Robust Point Cloud Registration". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA. June 18–24, 2022. 2022, pp. 11133–11142. DOI: 10.1109/CVPR52688.2022.01086 (cit. on p. 122).

[QYW+23] QIN, Zheng; YU, Hao; WANG, Changjian; GUO, Yulan; PENG, Yuxing; ILIC, Slobodan; HU, Dewen and XU, Kai: "GeoTransformer: Fast and Robust Point Cloud Registration With Geometric Transformer". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.8 (Aug. 2023), pp. 9806–9821. DOI: 10.1109/TPAMI.2023.3259038 (cit. on p. 122).

[RA12] RAUH, André and ARCE, Gonzalo R.: "Optimal Pivot Selection in Fast Weighted Median Search". In: *IEEE Transactions on Signal Processing* 60.8 (2012), pp. 4108–4117. DOI: 10.1109/TSP.2012.2197394 (cit. on p. 53).

[Raa17] RAAIJMAKERS, M.: "Towards environment perception for highly automated driving: with a case study on roundabouts". Proefschrift. Doctoral Dissertation. Eindhoven, Netherlands:

Technische Universiteit Eindhoven, June 2017 (cit. on pp. 10, 22, 194, 195).

[Ran23] Rangwala, Sabbir: "Lidar Miniaturization". In: *ADAS & Autonomous Vehicle International* April 2023 (2023), pp. 34–38 (cit. on p. 40).

[RB14] Raaijmakers, Marvin and Bouzouraa, Mohamed Essayed: "Circle Detection in Single-Layer Laser Scans for Roundabout Perception". In: *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Qingdao, China. Oct. 8–11, 2014. 2014, pp. 2636–2643 (cit. on pp. 22, 193).

[RB15] Raaijmakers, Marvin and Bouzouraa, Mohamed Essayed: "In-Vehicle Roundabout Perception Supported by A Priori Map Data". In: *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. Gran Canaria, Spain. Sept. 15–18, 2015. 2015, pp. 437–443 (cit. on pp. 22, 193).

[RBB09] Rusu, Radu Bogdan; Blodow, Nico and Beetz, Michael: "Fast Point Feature Histograms (FPFH) for 3D registration". In: *2009 IEEE International Conference on Robotics and Automation*. Kobe, Japan. May 12–17, 2009. 2009, pp. 3212–3217. doi: 10.1109/ROBOT.2009.5152473 (cit. on pp. 122, 168).

[RBP+17] Ruhhammer, Christian; Baumann, Michael; Protschky, Valentin; Kloeden, Horst; Klanner, Felix and Stiller, Christoph: "Automated Intersection Mapping From Crowd Trajectory Data". In: *IEEE Transactions on Intelligent Transportation Systems* 18.3 (2017), pp. 666–677. doi: 10.1109/TITS.2016.2585518 (cit. on p. 192).

[RCD18] Rubino, Cosimo; Crocco, Marco and Del Bue, Alessio: "3D Object Localisation from Multi-View Image Detections". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.6 (2018), pp. 1281–1294. doi: 10.1109/TPAMI.2017.2701373 (cit. on p. 39).

[RCG22] Roriz, Ricardo; Cabral, Jorge and Gomes, Tiago: "Automotive LiDAR Technology: A Survey". In: *IEEE Transactions on Intelligent Transportation Systems* 23.7 (2022), pp. 6282–6297. doi: 10.1109/TITS.2021.3086804 (cit. on p. 35).

[RDC+16] Riveiro, Belén; Díaz-Vilariño, Lucía; Conde-Carnero, Borja; Soilán, Mario and Arias, Pedro: "Automatic Segmentation and Shape-Based Classification of Retro-Reflective Traffic Signs from Mobile LiDAR Data". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 9.1 (2016), pp. 295–303. doi: 10.1109/JSTARS.2015.2461680 (cit. on pp. 36, 48).

[RDG18] Roynard, Xavier; Deschaud, Jean-Emmanuel and Goulette, François: "Paris-Lille-3D: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification". In: *The International Journal of Robotics Research* 37.6 (2018), pp. 545–557. doi: 10.1177/0278364918767506 (cit. on p. 31).

[RDT+21] Rosen, David M.; Doherty, Kevin J.; Terán Espinoza, Antonio and Leonard, John J.: "Advances in Inference and Representation for Simultaneous Localization and Mapping". In: *Annual Review of Control, Robotics, and Autonomous Systems* 4.1 (2021), pp. 215–242. doi: 10.1146/annurev-control-072720-082553 (cit. on p. 39).

[Reu14] Reuter, Stephan: "Multi-object tracking using random finite sets". Doctoral Dissertation. Ulm, Germany: Universität Ulm, 2014. doi: 10.18725/OPARU-3204 (cit. on pp. 128, 129, 134).

[RF18] Redmon, Joseph and Farhadi, Ali: Yolov3: An Incremental Improvement. Apr. 2018. arXiv: 1804.02767. url: https://arxiv.org/abs/1804.02767 (last retrieved 2023-03-01) (cit. on p. 43).

[RGG+14] Rossi, Ryan A.; Gleich, David F.; Gebremedhin, Assefaw H. and Patwary, Md. Mostofa Ali: "Fast maximum clique algorithms for large graphs". In: *Proceedings of the 23rd International Conference on World Wide Web*. Seoul, Korea (South). Apr. 7–11, 2014. Association for Computing Machinery, 2014, pp. 365–366. doi: 10.1145/2567948.2577283 (cit. on p. 162).

[Rou87] Rousseeuw, Peter J.: "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis". In: *Journal of Computational and Applied Mathematics* 20 (1987), pp. 53–65. doi: 10.1016/0377-0427(87)90125-7 (cit. on p. 291).

[RVV+14] Reuter, Stephan; Vo, Ba-Tuong; Vo, Ba-Ngu and Dietmayer, Klaus: "The Labeled Multi-Bernoulli Filter". In: *IEEE Transactions on Signal Processing* 62.12 (2014), pp. 3246–3260. doi: 10.1109/TSP.2014.2323064 (cit. on pp. 65, 129).

[SAE21] SAE J3016: Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. Tech. rep. Warrendale, PA, USA: SAE International, Apr. 2021 (cit. on p. 11).

[SAR18] Shen, Xiaotong; Ang, Marcelo H. and Rus, Daniela: "Conditional Compatibility Branch and Bound for Feature Cloud Matching". In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. Brisbane, QLD, Australia. May 21–25, 2018. 2018, pp. 5965–5970. doi: 10.1109/ICRA.2018.8460711 (cit. on p. 117).

[SB14] Schlichting, Alexander and Brenner, Claus: "Localization using automotive laser scanners and local pattern matching". In: *2014 IEEE Intelligent Vehicles Symposium Proceedings*. Dearborn, MI, USA. June 8–11, 2014. 2014, pp. 414–419. doi: 10.1109/IVS.2014.6856460 (cit. on pp. 36, 123).

[SBV+19] Schaefer, Alexander; Büscher, Daniel; Vertens, Johan; Luft, Lukas and Burgard, Wolfram: "Long-Term Urban Vehicle Localization Using Pole Landmarks Extracted from 3-D Lidar Scans". In: *2019 European Conference on Mobile*

Robots (ECMR). Prague, Czech Republic. Sept. 4–6, 2019. 2019, pp. 1–7. DOI: 10.1109/ECMR.2019.8870928 (cit. on pp. 37, 125).

[SCC23] SHAN, Xiaoyu; CABANI, Adnane and CHAFOUK, Houcine: "A Survey of Vehicle Localization: Performance Analysis and Challenges". In: *IEEE Access* 11 (2023), pp. 107085–107107. DOI: 10.1109/ACCESS.2023.3318885 (cit. on p. 125).

[Sch20] SCHMIDT, Benjamin: "Semantic Urban Maps via Instance Segmentation Networks and LiDAR ". Bachelor's Thesis. Karlsruhe, Germany: Institute for Measurement and Control, Karlsruhe Institute of Technology, 2020 (cit. on pp. 30, 45, 51, 63, 64, 71).

[Sch87] SCHUIRMANN, Donald J.: "A Comparison of the Two One-Sided Tests Procedure and the Power Approach for Assessing the Equivalence of Average Bioavailability". In: *Journal of Pharmacokinetics and Biopharmaceutics* 15.6 (1987), pp. 657–680. DOI: 10.1007/BF01068419 (cit. on p. 15).

[SDF+18] SCHNEIDER, Thomas; DYMCZYK, Marcin; FEHR, Marius; EGGER, Kevin; LYNEN, Simon; GILITSCHENSKI, Igor and SIEGWART, Roland: "Maplab: An Open Framework for Research in Visual-Inertial Mapping and Localization". In: *IEEE Robotics and Automation Letters* 3.3 (2018), pp. 1418–1425. DOI: 10.1109/LRA.2018.2800113 (cit. on p. 125).

[SDS+17] SEFATI, M.; DAUM, M.; SONDERMANN, B.; KREISKÖTHER, K. D. and KAMPKER, A.: "Improving vehicle localization using semantic and pole-like landmarks". In: *2017 IEEE Intelligent Vehicles Symposium (IV)*. Los Angeles, CA, USA. June 11–14, 2017. 2017, pp. 13–19. DOI: 10.1109/IVS.2017.7995692 (cit. on pp. 36, 37).

[Sel05] SELIG, J. M.: Geometric Fundamentals of Robotics. 2nd ed. Monographs in Computer Science. New York City, NY, USA: Springer Verlag, 2005 (cit. on p. 30).

[SFR+16] Shen, Xiaotong; Frazzoli, Emilio; Rus, Daniela and Ang, Marcelo H.: "Fast Joint Compatibility Branch and Bound for feature cloud matching". In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Daejeon, Korea (South). Oct. 9–14, 2016. 2016, pp. 1757–1764. doi: 10.1109/IROS.2016.7759281 (cit. on pp. 116, 117).

[SGD+18] Schubert, David; Goll, Thore; Demmel, Nikolaus; Usenko, Vladyslav; Stückler, Jörg and Cremers, Daniel: "The TUM VI Benchmark for Evaluating Visual-Inertial Odometry". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Madrid, Spain. Oct. 1–5, 2018. 2018, pp. 1680–1687. doi: 10.1109/IROS.2018.8593419 (cit. on p. 170).

[SGR16] Spangenberg, Robert; Goehring, Daniel and Rojas, Raúl: "Pole-based localization for autonomous vehicles in urban scenarios". In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Daejeon, Korea (South). Oct. 9–14, 2016. 2016, pp. 2161–2166. doi: 10.1109/IROS.2016.7759339 (cit. on p. 35).

[Sha22] Shashua, Amnon: Mobileye and Zeekr OTA Update Opens a New Chapter in Advanced Driver Assist. July 2022. url: https://www.mobileye.com/blog/mobileye-supervision-zeekr-ota-update/ (last retrieved 2023-08-28) (cit. on p. 40).

[Sha76] Shafer, Glenn: A Mathematical Theory of Evidence. Princeton, NJ, USA: Princeton University Press, 1976. doi: 10.1515/9780691214696 (cit. on p. 191).

[SHT09] Segal, A.; Haehnel, D. and Thrun, S.: "Generalized-ICP". In: *Proceedings of Robotics: Science and Systems*. Seattle, USA. June 28–July 1, 2009. 2009. doi: 10.15607/RSS.2009.V.021 (cit. on p. 120).

[SKF13] Schreiber, Markus; Knöppel, Carsten and Franke, Uwe: "LaneLoc: Lane marking based localization using highly accurate maps". In: *2013 IEEE Intelligent Vehicles Symposium*

(IV). Gold Coast, QLD, Australia. June 23–26, 2013. 2013, pp. 449–454. DOI: 10.1109/IVS.2013.6629509 (cit. on p. 33).

[SLK+17] SONS, Marc; LAUER, Martin; KELLER, Christoph G. and STILLER, Christoph: "Mapping and localization using surround view". In: *2017 IEEE Intelligent Vehicles Symposium (IV)*. Los Angeles, CA, USA. June 11–14, 2017. 2017, pp. 1158–1163. DOI: 10.1109/IVS.2017.7995869 (cit. on p. 125).

[SNS+13] SALAS-MORENO, Renato F.; NEWCOMBE, Richard A.; STRASDAT, Hauke; KELLY, Paul H.J. and DAVISON, Andrew J.: "SLAM++: Simultaneous Localisation and Mapping at the Level of Objects". In: *2013 IEEE Conference on Computer Vision and Pattern Recognition*. Portland, OR, USA. June 23–28, 2013. 2013, pp. 1352–1359. DOI: 10.1109/CVPR.2013.178 (cit. on p. 39).

[SP13] SÜNDERHAUF, Niko and PROTZEL, Peter: "Switchable constraints vs. max-mixture models vs. RRR - A comparison of three approaches to robust pose graph SLAM". In: *2013 IEEE International Conference on Robotics and Automation*. Karlsruhe, Germany. May 6–10, 2013. 2013, pp. 5198–5203. DOI: 10.1109/ICRA.2013.6631320 (cit. on p. 40).

[SPL+17] SÜNDERHAUF, Niko; PHAM, Trung T.; LATIF, Yasir; MILFORD, Michael and REID, Ian: "Meaningful maps with object-oriented semantic mapping". In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Vancouver, BC, Canada. Sept. 24–28, 2017. 2017, pp. 5079–5085. DOI: 10.1109/IROS.2017.8206392 (cit. on p. 39).

[SPV13] SOHEILIAN, Bahman; PAPARODITIS, Nicolas and VALLET, Bruno: "Detection and 3D reconstruction of traffic signs from multiple view color images". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 77 (2013), pp. 1–20. DOI: 10.1016/j.isprsjprs.2012.11.009 (cit. on pp. 35, 38).

[Sra11] Sra, Suvrit: "A short note on parameter approximation for von Mises-Fisher distributions: and a fast implementation of Is (x)". In: *Computational Statistics* 27.1 (Feb. 2011), pp. 177–190. doi: 10.1007/s00180-011-0232-x (cit. on p. 160).

[SRD17a] Stübler, Manuel; Reuter, Stephan and Dietmayer, Klaus: "A continuously learning feature-based map using a bernoulli filtering approach". In: *2017 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*. Bonn, Germany. Oct. 10–12, 2017. 2017, pp. 1–6. doi: 10.1109/SDF.2017.8126353 (cit. on p. 116).

[SRD17b] Stübler, Manuel; Reuter, Stephan and Dietmayer, Klaus: "Consistency of feature-based random-set Monte-Carlo localization". In: *2017 European Conference on Mobile Robots (ECMR)*. Paris, France. Sept. 6–8, 2017. 2017, pp. 1–6. doi: 10.1109/ECMR.2017.8098674 (cit. on pp. 108, 116).

[SS18] Sons, Marc and Stiller, Christoph: "Efficient Multi-Drive Map Optimization towards Life-long Localization using Surround View". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. Maui, HI, USA. Nov. 4–7, 2018. 2018, pp. 2671–2677. doi: 10.1109/ITSC.2018.8570011 (cit. on p. 125).

[SSE+17] Schubert, Erich; Sander, Jörg; Ester, Martin; Kriegel, Hans Peter and Xu, Xiaowei: "DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN". In: 42.3 (July 2017). doi: 10.1145/3068335 (cit. on p. 52).

[SSO06] Szarvas, M.; Sakai, U. and Ogata, Jun: "Real-time Pedestrian Detection Using LIDAR and Convolutional Neural Networks". In: *2006 IEEE Intelligent Vehicles Symposium*. Meguro-Ku, Japan. June 13–15, 2006. 2006, pp. 213–218. doi: 10.1109/IVS.2006.1689630 (cit. on p. 37).

[Str15] Strauss, Tobias: "Kalibrierung von Multi-Kamera-Systemen - Kombinierte Schätzung von intrinsischem Abbildungsverhalten der einzelnen Kameras und deren relativer Lage zueinander

ohne Erfordernis sich überlappender Sichtbereiche". Doctoral Dissertation. Karlsruhe, Germany: Karlsruher Institut für Technologie (KIT), 2015. DOI: 10.5445/IR/1000051877 (cit. on p. 42).

[Stü18] Stübler, Manuel: "Self-assessing localization and long-term mapping using random finite sets". Doctoral Dissertation. Ulm, Germany: Universität Ulm, 2018. DOI: 10.18725/OPARU-10683 (cit. on pp. 108, 116).

[SWD20] Sucar, Edgar; Wada, Kentaro and Davison, Andrew: "NodeSLAM: Neural Object Descriptors for Multi-View Shape Reconstruction". In: *2020 International Conference on 3D Vision (3DV)*. Fukuoka, Japan. Nov. 25–28, 2020. 2020, pp. 949–958. DOI: 10.1109/3DV50981.2020.00105 (cit. on p. 38).

[SYC21] Shi, Jingnan; Yang, Heng and Carlone, Luca: "ROBIN: a Graph-Theoretic Approach to Reject Outliers in Robust Estimation using Invariants". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. Xi'an, China. May 30–June 5, 2021. 2021, pp. 13820–13827. DOI: 10.1109/ICRA48506.2021.9562007 (cit. on pp. 119, 131).

[SYM19] Shimizu, Satoshi; Yamaguchi, Kazuaki and Masuda, Sumio: "A Branch-and-Bound Based Exact Algorithm for the Maximum Edge-Weight Clique Problem". In: *Computational Science/Intelligence & Applied Informatics*. Ed. by Lee, Roger. Vol. 787. Studies in Computational Intelligence. Cham, Switzerland: Springer International Publishing, 2019, pp. 27–47 (cit. on p. 147).

[SYM20] Shimizu, Satoshi; Yamaguchi, Kazuaki and Masuda, Sumio: "A Maximum Edge-Weight Clique Extraction Algorithm Based on Branch-and-Bound". In: *Discrete Optimization* 37 (2020), p. 100583 (cit. on p. 147).

[SZB14] Strauss, Tobias; Ziegler, Julius and Beck, Johannes: "Calibrating multiple cameras with non-overlapping views using coded checkerboard targets". In: *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Qingdao, China. Oct. 8–11, 2014. 2014, pp. 2623–2628. doi: 10.1109/ITSC.2014.6958110 (cit. on p. 42).

[TBF05] Thrun, Sebastian; Burgard, Wolfram and Fox, Dieter: Probabilistic Robotics. Cambridge, MA, USA: MIT Press, 2005 (cit. on pp. 30, 112, 115).

[TFC+14] Tombari, Federico; Fioraio, Nicola; Cavallari, Tommaso; Salti, Samuele; Petrelli, Alioscia and Di Stefano, Luigi: "Automatic detection of pole-like structures in 3D urban environments". In: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Chicago, IL, USA. Sept. 14–18, 2014. 2014, pp. 4922–4929. doi: 10.1109/IROS.2014.6943262 (cit. on p. 36).

[TJR+13] Taguchi, Yuichi; Jian, Yong-Dian; Ramalingam, Srikumar and Feng, Chen: "Point-plane SLAM for hand-held 3D sensors". In: *2013 IEEE International Conference on Robotics and Automation*. Karlsruhe, Germany. May 6–10, 2013. 2013, pp. 5182–5189. doi: 10.1109/ICRA.2013.6631318 (cit. on p. 123).

[TL94] Turk, Greg and Levoy, Marc: "Zippered Polygon Meshes from Range Images". In: *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*. Orlando, FL, USA. July 24–29, 1994. Association for Computing Machinery, 1994, pp. 311–318. doi: 10.1145/192161.192241 (cit. on pp. 139, 151, 162, 163, 165, 167).

[TNS+21] Tschopp, Florian; Nieto, Juan; Siegwart, Roland and Cadena, Cesar: Superquadric Object Representation for Optimization-based Semantic SLAM. 2021. doi: 10.3929/ETHZ-B-000487527. arXiv: 2109.09627.

URL: https://arxiv.org/abs/2109.09627 (last retrieved 2023-09-15) (cit. on p. 39).

[Tom18] TomTom International BV: How we create our Real-Time Maps | TomTom Automotive. Original URL accessed on 2018-08-16, last accessed state from 2018-10-27 via the Internet Archive on 2023-03-06 at https://web.archive.org/web/20181027154245/https://www.tomtom.com/automotive/products-services/real-time-maps/how-we-create-best-real-time-maps/. 2018. URL: https://www.tomtom.com/automotive/products-services/real-time-maps/how-we-create-best-real-time-maps/ (cit. on p. 20).

[TRC12] Trevor, Alexander J. B.; Rogers, John G. and Christensen, Henrik I.: "Planar surface SLAM with 3D and 2D sensors". In: *2012 IEEE International Conference on Robotics and Automation*. Saint Paul, MN, USA. May 14–18, 2012. 2012, pp. 3041–3048. DOI: 10.1109/ICRA.2012.6225287 (cit. on p. 123).

[TZ00] Torr, P.H.S. and Zisserman, A.: "MLESAC: A New Robust Estimator with Application to Estimating Image Geometry". In: *Computer Vision and Image Understanding* 78.1 (2000), pp. 138–156. DOI: 10.1006/cviu.1999.0832 (cit. on p. 120).

[TZV09] Timofte, Radu; Zimmermann, Karel and Van Gool, Luc: "Multi-view traffic sign detection, recognition, and 3D localisation". In: *2009 Workshop on Applications of Computer Vision (WACV)*. Snowbird, UT, USA. Dec. 7–9, 2009. 2009, pp. 1–8. DOI: 10.1109/WACV.2009.5403121 (cit. on p. 35).

[UGB+13] Underwood, J. P.; Gillsjö, D.; Bailey, T. and Vlaskine, V.: "Explicit 3D change detection using ray-tracing in spherical coordinates". In: *2013 IEEE International Conference on Robotics and Automation*. Karlsruhe, Germany. May 6–10, 2013. 2013, pp. 4735–4741. DOI: 10.1109/ICRA.2013.6631251 (cit. on pp. 194, 198).

[Ume91] Umeyama, S.: "Least-squares estimation of transformation parameters between two point patterns". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13.4 (1991), pp. 376–380. doi: 10.1109/34.88573 (cit. on pp. 107, 139, 155, 156, 163).

[VGM+23] Vizzo, Ignacio; Guadagnino, Tiziano; Mersch, Benedikt; Wiesmann, Louis; Behley, Jens and Stachniss, Cyrill: "KISS-ICP: In Defense of Point-to-Point ICP – Simple, Accurate, and Robust Registration If Done the Right Way". In: *IEEE Robotics and Automation Letters (RA-L)* 8.2 (2023), pp. 1–8. doi: 10.1109/LRA.2023.3236571 (cit. on pp. 45, 101, 171, 172).

[VYF+13] Vu, Anh; Yang, Qichi; Farrell, Jay A. and Barth, Matthew: "Traffic sign detection, state estimation, and identification using onboard sensors". In: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. The Hague, Netherlands. Oct. 6–9, 2013. 2013, pp. 875–880. doi: 10.1109/ITSC.2013.6728342 (cit. on pp. 36–38, 48).

[Wan19] Wang, Congchao; Wang, Yizhi; Wang, Yinxue; Wu, Chiung-Ting and Yu, Guoqiang: "muSSP: Efficient Min-Cost Flow Algorithm for Multi-Object Tracking". In: *Advances in Neural Information Processing Systems*. Vancouver, BC, Canada. Dec. 8–14, 2019. Ed. by Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E. and Garnett, R. Vol. 32. Red Hook, NY, USA: Curran Associates, Inc., 2020, pp. 425–434. doi: 10.5555/3454287.3454326 (cit. on pp. 10, 40, 65).

[Way22] Waymo LLC: Waymo Open Dataset – 3D Semantic Segmentation. 2022. url: https://waymo.com/open/challenges/2022/3d-semantic-segmentation/ (last retrieved 2023-08-31) (cit. on p. 32).

[WE18] Wang, Jinkun and Englot, Brendan: "Robust Exploration with Multiple Hypothesis Data Association". In: *2018*

IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Madrid, Spain. Oct. 1–5, 2018. 2018, pp. 3537–3544. DOI: 10.1109/IROS.2018.8593753 (cit. on pp. 40, 117).

[Wei37] Weiszfeld, E.: "Sur le point pour lequel la Somme des distances de *n* points donnés est minimum". In: *Tohoku Mathematical Journal, First Series* 43 (1937), pp. 355–386 (cit. on p. 72).

[WGK21] Wong, Kelvin; Gu, Yanlei and Kamijo, Shunsuke: "Mapping for Autonomous Driving: Opportunities and Challenges". In: *IEEE Intelligent Transportation Systems Magazine* 13.1 (2021), pp. 91–106. DOI: 10.1109/MITS.2020.3014152 (cit. on p. 9).

[WGV+23] Wiesmann, Louis; Guadagnino, Tiziano; Vizzo, Ignacio; Zimmerman, Nicky; Pan, Yue; Kuang, Haofei; Behley, Jens and Stachniss, Cyrill: "LocNDF: Neural Distance Field Mapping for Robot Localization". In: *IEEE Robotics and Automation Letters* 8.8 (2023), pp. 4999–5006. DOI: 10.1109/LRA.2023.3291274 (cit. on p. 125).

[WHJ+19] Wang, Chunxiang; Huang, Hairu; Ji, Yang; Wang, Bing and Yang, Ming: "Vehicle Localization at an Intersection Using a Traffic Light Map". In: *IEEE Transactions on Intelligent Transportation Systems* 20.4 (2019), pp. 1432–1441. DOI: 10.1109/TITS.2018.2851788 (cit. on p. 34).

[WK19] Wang, Yali and Kuhn, Steffen: Reliable and safe maps for automated driving. Tech. rep. Erlangen, Germany: Elektrobit Automotive GmbH, 2019. URL: https://www.elektrobit.com/tech-corner/reliable-and-safe-maps-for-automated-driving/ (last retrieved 2023-09-27) (cit. on p. 10).

[WKM+19] Wu, Yuxin; Kirillov, Alexander; Massa, Francisco; Lo, Wan-Yen and Girshick, Ross: Detectron2. 2019. URL: https://github.com/facebookresearch/detectron2 (last retrieved 2023-10-14) (cit. on p. 94).

[WLL+23] Wang, Huijie; Li, Tianyu; Li, Yang; Chen, Li; Sima, Chonghao; Liu, Zhenbo; Wang, Bangjun; Jia, Peijin; Wang, Yuting; Jiang, Shengyin et al.: "OpenLane-V2: A Topology Reasoning Benchmark for Unified 3D HD Mapping". In: *Advances in Neural Information Processing Systems*. Vancouver, BC, Canada. Dec. 10–16, 2023. Ed. by Oh, A.; Neumann, T.; Globerson, A.; Saenko, K.; Hardt, M. and Levine, S. Vol. 36. Curran Associates, Inc., 2023, pp. 18873–18884 (cit. on p. 32).

[WQA+21] Wilson, Benjamin; Qi, William; Agarwal, Tanmay; Lambert, John; Singh, Jagjeet; Khandelwal, Siddhesh; Pan, Bowen; Kumar, Ratnesh; Hartnett, Andrew; Kaesemodel Pontes, Jhony et al.: "Argoverse 2: Next Generation Datasets for Self-Driving Perception and Forecasting". In: *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*. Virtual Conference. Dec. 6–14, 2021. Ed. by Vanschoren, J. and Yeung, S. Vol. 1. Curran, 2021 (cit. on p. 41).

[WRW15] Welzel, Andre; Reisdorf, Pierre and Wanielik, Gerd: "Improving Urban Vehicle Localization with Traffic Sign Recognition". In: *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. Gran Canaria, Spain. Sept. 12–15, 2015. 2015, pp. 2728–2732. doi: 10.1109/ITSC.2015.438 (cit. on p. 35).

[WS05] Weingarten, J. and Siegwart, R.: "EKF-based 3D SLAM for structured environment reconstruction". In: *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Edmonton, AB, Canada. Aug. 2–6, 2005. 2005, pp. 3834–3839. doi: 10.1109/IROS.2005.1545285 (cit. on p. 123).

[WS06] Weingarten, Jan and Siegwart, Roland: "3D SLAM using planar segments". In: *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Beijing, China. Oct. 9–15, 2006. 2006, pp. 3062–3067 (cit. on p. 123).

[WWZ16] Weber, Michael; Wolf, Peter and Zöllner, J. Marius: "DeepTLR: A single deep convolutional network for detection and classification of traffic lights". In: *2016 IEEE Intelligent Vehicles Symposium (IV)*. Gothenburg, Sweden. June 19–22, 2016. 2016, pp. 342–348. doi: 10.1109/IVS.2016.7535408 (cit. on p. 34).

[XZC+19] Xu, Yingkun; Zhou, Xiaolong; Chen, Shengyong and Li, Fenfen: "Deep learning for multiple object tracking: a survey". In: *IET Computer Vision* 13.4 (2019), pp. 355–368. doi: 10.1049/iet-cvi.2018.5598 (cit. on p. 65).

[YC19] Yang, Heng and Carlone, Luca: "A Polynomial-time Solution for Robust Registration with Extreme Outlier Rates". In: *Proceedings of Robotics: Science and Systems*. Freiburg im Breisgau, Germany. June 22–26, 2019. 2019. doi: 10.15607/RSS.2019.XV.003 (cit. on pp. 121, 130, 162).

[YC20] Yang, Heng and Carlone, Luca: "One Ring to Rule Them All: Certifiably Robust Geometric Perception with Outliers". In: *Advances in Neural Information Processing Systems*. Virtual Conference. Dec. 6–12, 2020. Ed. by Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.F. and Lin, H. Vol. 33. Red Hook, NY, USA: Curran Associates, Inc., 2020, pp. 18846–18859 (cit. on pp. 114, 121).

[YC23] Yang, Heng and Carlone, Luca: "Certifiably Optimal Outlier-Robust Geometric Perception: Semidefinite Relaxations and Scalable Global Optimization". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.3 (2023), pp. 2816–2834. doi: 10.1109/TPAMI.2022.3179463 (cit. on pp. 114, 121).

[Yen71] Yen, Jin Y.: "Finding the K Shortest Loopless Paths in a Network". In: *Management Science* 17.11 (1971), pp. 712–716 (cit. on p. 130).

[YHQ+22] Yang, Jiaqi; Huang, Zhiqiang; Quan, Siwen; Qi, Zhaoshuai and Zhang, Yanning: "SAC-COT: Sample Consensus by Sampling Compatibility Triangles in Graphs for 3-D Point Cloud Registration". In: *IEEE Transactions on Geoscience and Remote Sensing* 60 (2022), pp. 1–15. doi: 10.1109/TGRS.2021.3058552 (cit. on pp. 121, 134).

[YLC+16] Yang, Jiaolong; Li, Hongdong; Campbell, Dylan and Jia, Yunde: "Go-ICP: A Globally Optimal Solution to 3D ICP Point-Set Registration". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.11 (2016), pp. 2241–2254. doi: 10.1109/TPAMI.2015.2513405 (cit. on pp. 120, 130, 162).

[YLJ13] Yang, Jiaolong; Li, Hongdong and Jia, Yunde: "Go-ICP: Solving 3D Registration Efficiently and Globally Optimally". In: *2013 IEEE International Conference on Computer Vision*. Sydney, NSW, Australia. Dec. 1–8, 2013. 2013, pp. 1457–1464. doi: 10.1109/ICCV.2013.184 (cit. on pp. 120, 130, 162).

[YS19] Yang, Shichao and Scherer, Sebastian: "CubeSLAM: Monocular 3-D Object SLAM". In: *IEEE Transactions on Robotics* 35.4 (2019), pp. 925–938. doi: 10.1109/TRO.2019.2909168 (cit. on p. 39).

[YSC21] Yang, Heng; Shi, Jingnan and Carlone, Luca: "TEASER: Fast and Certifiable Point Cloud Registration". In: *IEEE Transactions on Robotics* 37.2 (2021), pp. 314–333. doi: 10.1109/TRO.2020.3033695 (cit. on pp. 121, 130, 162).

[YWW+19] You, Changbin; Wen, Chenglu; Wang, Cheng; Li, Jonathan and Habib, Ayman: "Joint 2-D–3-D Traffic Sign Landmark Data Set for Geo-Localization Using Mobile Laser Scanning Data". In: *IEEE Transactions on Intelligent Transportation Systems* 20.7 (2019), pp. 2550–2565. doi: 10.1109/TITS.2018.2868168 (cit. on p. 37).

[YZF+23] Yang, Jiaqi; Zhang, Xiyu; Fan, Shichao; Ren, Chunlin and Zhang, Yanning: "Mutual Voting for Ranking 3D Correspondences". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023). (Early Access), pp. 1–18. doi: 10.1109/TPAMI.2023.3268297 (cit. on p. 122).

[ZBI12a] Zinoune, Clément; Bonnifait, Philippe and Ibañez-Guzmán, Javier: "A Sequential Test for Autonomous Localisation of Map Errors for Driving Assistance Systems". In: *2012 15th International IEEE Conference on Intelligent Transportation Systems*. Anchorage, AK, USA. Sept. 16–16, 2012. 2012, pp. 1377–1382 (cit. on p. 192).

[ZBI12b] Zinoune, Clément; Bonnifait, Philippe and Ibañez-Guzmán, Javier: "Detection of Missing Roundabouts in Maps for Driving Assistance Systems". In: *2012 IEEE Intelligent Vehicles Symposium*. Madrid, Spain. June 3–7, 2012. 2012, pp. 123–128 (cit. on pp. 22, 192).

[ZBI16] Zinoune, Clément; Bonnifait, Philippe and Ibañez-Guzmán, Javier: "Sequential FDIA for Autonomous Integrity Monitoring of Navigation Maps on Board Vehicles". In: *IEEE Transactions on Intelligent Transportation Systems* 17.1 (2016), pp. 143–155. doi: 10.1109/TITS.2015.2474145 (cit. on p. 192).

[ZBS+14] Ziegler, Julius; Bender, Philipp; Schreiber, Markus; Lategahn, Henning; Strauss, Tobias; Stiller, Christoph; Dang, Thao; Franke, Uwe; Appenrodt, Nils; Keller, Christoph G. et al.: "Making Bertha Drive—An Autonomous Journey on a Historic Route". In: *IEEE Intelligent Transportation Systems Magazine* 6.2 (2014), pp. 8–20. doi: 10.1109/MITS.2014.2306552 (cit. on pp. 8, 33).

[ZD14] Zhou, Lipu and Deng, Zhidong: "LIDAR and vision-based real-time traffic sign detection and recognition algorithm for intelligent vehicle". In: *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Qingdao, China.

Oct. 8–11, 2014. 2014, pp. 578–583. DOI: 10.1109/ITSC.2014.6957752 (cit. on p. 37).

[Zha23] ZHANG, Xiyu: Three questions regarding the code. Also available via the Internet Archive at https://web.archive.org/web/20231130103525/https://github.com/zhangxy0517/3D-Registration-with-Maximal-Cliques/issues/3. May 2023. URL: https://github.com/zhangxy0517/3D-Registration-with-Maximal-Cliques/issues/3#issuecomment-1564268834 (last retrieved 2023-11-30) (cit. on p. 123).

[ZKB+20] ZAKHAROV, Sergey; KEHL, Wadim; BHARGAVA, Arjun and GAIDON, Adrien: "Autolabeling 3D Objects With Differentiable Rendering of SDF Shape Priors". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Seattle, WA, USA. June 13–19, 2020. 2020, pp. 12221–12230. DOI: 10.1109/CVPR42600.2020.01224 (cit. on p. 73).

[ZLN08] ZHANG, Li; LI, Yuan and NEVATIA, Ramakant: "Global Data Association for Multi-Object Tracking Using Network Flows". In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. Anchorage, AK, USA. June 23–28, 2008. IEEE, 2008, pp. 1–8. DOI: 10.1109/CVPR.2008.4587584 (cit. on pp. 65, 67, 68).

[ZPK16] ZHOU, Qian-Yi; PARK, Jaesik and KOLTUN, Vladlen: "Fast Global Registration". In: *Computer Vision – ECCV 2016*. Amsterdam, Netherlands. Oct. 11–14, 2016. Ed. by LEIBE, Bastian; MATAS, Jiri; SEBE, Nicu and WELLING, Max. Lecture Notes in Computer Science. Cham, Switzerland: Springer International Publishing, 2016, pp. 766–782 (cit. on p. 169).

[ZS14] ZHANG, Ji and SINGH, Sanjiv: "LOAM: Lidar Odometry and Mapping in Real-time". In: *Proceedings of Robotics: Science and Systems*. Berkeley, USA. July 12–16, 2014. Ed. by FOX, Dieter; KAVRAKI, Lydia E. and KURNIAWATI, Hanna. 2014 (cit. on pp. 38, 123).

[ZSN+17] Zeng, Andy; Song, Shuran; Niessner, Matthias; Fisher, Matthew; Xiao, Jianxiong and Funkhouser, Thomas: "3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI, USA. July 21–26, 2017. 2017, pp. 199–208. doi: 10.1109/CVPR.2017.29 (cit. on p. 121).

[ZWK21] Zhou, Lipu; Wang, Shengze and Kaess, Michael: "π-LSAM: LiDAR Smoothing and Mapping With Planes". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. Xi'an, China. May 30–June 5, 2021. 2021, pp. 5751–5757. doi: 10.1109/ICRA48506.2021.9561933 (cit. on p. 124).

[ZYZ+23] Zhang, Xiyu; Yang, Jiaqi; Zhang, Shikun and Zhang, Yanning: "3D Registration With Maximal Cliques". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Vancouver, BC, Canada. June 18–22, 2023. 2023, pp. 17745–17754 (cit. on pp. 122, 169).

List of Figures

1.1 Overview of the Map Verification Framework 5

2.1 HD Map Layers 9
2.2 Lanelet2 Map Layers 13
2.3 Example of Geo-referenced Aerial Imagery 18
2.4 Illustration of the observed map changes 19

3.1 Overview of the Parametric Mapping Module 29
3.2 Sensor Setups 42
3.3 Example Detections of the DNN 44
3.4 Illustration of Motion Parallax Effect 46
3.5 Illustration of Static Parallax Effect 48
3.6 Illustration of Lidar Beam Divergence 49
3.7 Parallax Compensation 49
3.8 Lidar Point Weighting 51
3.9 Lidar Point Distance Histogram 52
3.10 Pole Estimation 56
3.11 Traffic Light Estimation 57
3.12 Traffic Sign Estimation 59
3.13 Measurement Duplicates 61
3.14 Weighting Function Examples 62
3.15 Mapping Example 63
3.16 Acausal Processing 64
3.17 Illustration of muSSP 66
3.18 Data Association for Traffic Light Mapping 69
3.19 Data Association for Traffic Sign Mapping 70
3.20 Weighting Function Examples 72

3.21 Detection Rendering Instance IoU 77
3.22 Parameter Optimization Scheme 79
3.23 Sequence Routes of the Dataset 81
3.24 Measurement Errors over Detection Distance (2020) 86
3.25 Measurement Errors over Detection Distance (2023) 87
3.26 1/5/10-Recall over Detection Distance 88
3.27 Qualitative Examples of Parametric Detections (2020) 90
3.28 Qualitative Examples of Parametric Detections (2020) 91
3.29 Qualitative Examples of Parametric Detections (2023) 92
3.30 Qualitative Examples of Parametric Detections (2023) 93
3.31 Runtime Analysis 95
3.32 Qualitative Examples of the HAD Map (2020) 97
3.33 Qualitative Examples of the HAD Map (2020) 98
3.34 Qualitative Examples of the HAD Map (2023) 99
3.35 Qualitative Examples of the HAD Map (2023) 100

4.1 Probabilistic Correspondence Graph Toy Example 111
4.2 Binary Correspondence Graph 132
4.3 Optimality in Residual and Correspondence Space 140
4.4 Exemplary Probabilistic Correspondence Graph 143
4.5 Simulated Euclidean Distance Differences 152
4.6 Comparison of JEADDs and Affine Grassmannian Distances . . 154
4.7 Data Association Example 163
4.8 Simulation Results for Unknown Correspondences 165
4.9 Simulation Results for Outlier Rejection 167
4.10 Comparison of Localization References 172
4.11 DPE Histograms for Localization in Up-to-date Maps 177
4.12 Localization Results in Up-to-date Maps 179
4.13 Qualitative Examples of Localization in an Outdated Map . . 181
4.14 Qualitative Examples of Localization in an Outdated Map . . 182
4.15 Localization Results in Outdated Maps 183
4.16 Computation Times for Localization in Up-to-date Maps . . . 184

5.1 Evidential Map Verification Approach 197
5.2 Lidar Ray Casting Example 198

5.3 Verification over Range 206
5.4 Qualitative Examples of Verifying an Outdated Map 210
5.5 Qualitative Examples of Verifying an Outdated Map 211
5.6 Exemplary Limitations of the Verification Approach 212

List of Tables

2.1 Map Changes over Years 20

3.1 RIIoU Metric for Parametric Detections 89
3.2 RIIoU Metric for the HAD Map 96
3.3 MAE Reported by Pauls et al. 102
3.4 Measurement Errors Reported by Plachetka et al. 102

4.1 Related Works for Data Association 114
4.2 Pearson Correlation of Correspondence Space 141
4.3 Spearman Correlation of Correspondence Space 141
4.4 Point Cloud Registration on KITTI 169
4.5 Comparison of APE and ADPE 176
4.6 Comparison of MPE and MDPE 176
4.7 Localization Results in Up-to-date Maps 177
4.8 Localization Results in Outdated Maps 180
4.9 Storage Requirements 185

5.1 Random Forest Classifier Performance 200
5.2 Shares of Verified Visible Landmarks 207
5.3 Eventual Classification Performance 208

B.1 Sequences from 2020 293
B.2 Sequences from 2023 294
B.3 2020 Scenarios 295
B.4 2023 Scenarios 297

C.1 Precision of Parametric Detections (2020) 300
C.2 Precision of Parametric Detections (2023) 301

Acronyms

ADAS	advanced driver assistance system
ADPE	average delta pose error
AGI	artificial general intelligence
AP	average precision
APE	average (absolute) pose error
API	application programming interface
AR	average recall
BBA	basic belief assignment (also referred to as basic mass assignment)
BEV	bird's eye view
CCBB	conditional compatibility branch and bound
cdf	cumulative density function
CPU	central processing unit
CS	correspondence space
DBSCAN	density-based spacial clustering of applications with noise
DNN	deep neural network

DPE	delta pose error
DT	distance transform
EDD	Euclidean distance difference
EKF	extended Kalman filter
ER	Erdős-Rényi
FCGF	fully convolutional geometric feature
FISST	finite set statistics
FMCW	frequency modulated continuous wave
FoV	field of view
FPFH	fast point feature histogram
GNC	graduated nonconvexity
GNSS	global navigation satellite system
GPU	graphics processing unit
GSD	ground sample distance
HAD	highly automated driving
HD	high definition
HDR	high dynamic range
IC	individual compatibility (also referred to as individual consistency)
ICP	iterative closest point
INS	inertial navigation system

IoU	intersection over union
IPJC	incremental posterior joint compatibility
JC	joint compatibility (also referred to as joint consistency)
JCBB	joint compatibility branch and bound
JEADD	joint Euclidean angular distance difference
KLD	Kullback-Leibler divergence
LZMA	Lempel-Ziv-Markov chain algorithm
MAE	mean absolute error
MB	multi-Bernoulli
MCS	maximum clique search
MDPE	maximum delta pose error
mIoU	mean intersection over union
ML	machine learning
MLE	maximum likelihood estimation
MPE	maximum (absolute) pose error
mRIIoU	mean RIIoU (Rendering Instance IoU)
MRT	Institute of Measurement and Control Systems
NIS	normalized innovation squared
PC	pairwise (geometric) compatibility (also referred to as pairwise (geometric) consistency)
PCA	principal component analysis

PCB printed circuit board

PCG probabilistic correspondence graph

pdf probability density function

RANSAC random sample consensus

RE rotation error

RFS random finite set

RIIoU Rendering Instance IoU (Intersection over Union)

RMSE root mean squared error

ROI region of interest

RR registration recall

RS residual space

RTK real-time kinematic

SCNN sequential compatibility nearest neighbor

SD standard definition

SDP semidefinite programming

SEDD squared Euclidean distance difference

SLAM simultaneous localization and mapping

SMAC sequential model-based algorithm configuration

SVD singular value decomposition

SVM support vector machine

TE translation error

TIM transformation invariant measurement

TLS truncated least squares

ToF time of flight

TOST two one sided tests

VG variance gamma

A Inferior Metrics

This appendix describes two self-supervised metrics that turned out not to work as well as the proposed mRIIoU metric.

Silhouette Coefficient

The first is to see mapping as clustering of the parametric detections. While there are many metrics to assess the outcome of clustering methods, the silhouette coefficient [Rou87] offers a good performance at high robustness to edge cases [AGM+13]. An implementation is available in Scikit-learn [PVG+11].

One can adapt its definition to a measurement $\boldsymbol{d}_i$ belonging to a map element $\boldsymbol{\ell}_k$, denoted as $\boldsymbol{d}_i \to \boldsymbol{\ell}_k$, where $|\boldsymbol{\ell}_k|$ denotes the total number of measurements associated to $\boldsymbol{\ell}_k$. The silhouette coefficient $sc(\boldsymbol{d}_i)$ for each measurement $\boldsymbol{d}_i$ can then be defined by

$$a(\boldsymbol{d}_i) := \frac{1}{|\boldsymbol{\ell}_k|} \sum_{\substack{\boldsymbol{d}_j \to \boldsymbol{\ell}_k \\ \boldsymbol{d}_j \neq \boldsymbol{d}_i}} \|\boldsymbol{d}_i - \boldsymbol{d}_j\|, \tag{A.1}$$

$$b(\boldsymbol{d}_i) := \min_{\boldsymbol{\ell}_l \neq \boldsymbol{\ell}_k} \frac{1}{|\boldsymbol{\ell}_l|} \sum_{\boldsymbol{d}_j \to \boldsymbol{\ell}_l} \|\boldsymbol{d}_i - \boldsymbol{d}_j\|, \tag{A.2}$$

$$sc(\boldsymbol{d}_i) := \frac{b(\boldsymbol{d}_i) - a(\boldsymbol{d}_i)}{\max(a(\boldsymbol{d}_i), b(\boldsymbol{d}_i))}. \tag{A.3}$$

The metric for evaluating a map could then be the average silhouette coefficient taken over all measurements.

$$p(\mathcal{M} \mid \mathcal{D}_{t_i}, i \in \{1, \ldots, N\}) \approx \sum_{\boldsymbol{d}_i \in \mathcal{D}} sc(\boldsymbol{d}_i) \tag{A.4}$$

The problem is that the silhouette coefficient, as most clustering metrics, assumes a uniform distribution of clusters, *i.e.* map elements. This is not true since some traffic lights are closely together when mounted at the same pole while some are far apart when being mounted at different poles. Consequently, the silhouette coefficient was discarded as suitable metric.

Probabilistic Map Metric

The second idea is to view the map as outcome of a RFS estimation problem. This view is inspired by (multi-hypothesis) filter-based SLAM [MVA+11, DRD15, FGS+17]. One can model the assignment of detections with a joint stochastic process combining the likelihoods of landmark existence, detections, clutter, and spatial distribution of detections around landmarks.

The problem with this theoretically very appealing metric is to obtain initial estimates for the necessary parameters to properly model clutter, detections and their respective spatial distributions probabilistically. An iterative approach could extract the parameters from an initial map, optimize the mapping parameters with this metric, refine the metric parameters with an improved map *etc*. Eventually, this approach was discarded as it bears the risk of a self-fulfilling prophecy, *i.e.* converging to intrinsically consistent parameters that are inconsistent with external evidence, and results were subjectively inferior to the RIIoU metric.

B Sequences and Scenarios

Tables B.1 and B.2 list the sequences used for this work.

Table B.1: Sequences from 2020 with date, clock time at start, length in (non-standstill) seconds, weather and road conditions.

Sequence	Date	Time	Length	Weather	Road
Adenauer 01	2020-07-28	10.23	515.2 s	overcast	dry
Adenauer 02	2020-07-28	10.35	270.8 s	overcast	dry
Adenauer 04	2020-07-28	11.04	271.4 s	overcast	dry
Moltke Big 01	2020-07-27	13.40	598.4 s	sunny	dry
Moltke Big 02	2020-07-27	14.10	447.9 s	sunny	dry
Moltke Big 03	2020-07-27	14.46	441.6 s	sunny with cloudy intervals	dry
Moltke Big 04	2020-07-28	7.20	422.5 s	sunny	dry
Moltke Small 01	2020-07-27	11.54	405.8 s	sunny	dry
Moltke Small 02	2020-07-27	12.18	324.6 s	sunny	dry
Moltke Small 03	2020-07-27	12.27	272.8 s	sunny	dry
Moltke Small 04	2020-07-28	7.08	280.1 s	sunny	dry
Ostring 01	2020-07-24	15.51	636.0 s	sunny with cloudy intervals	dry
Ostring 02	2020-07-24	16.03	353.6 s	sunny with cloudy intervals	dry
Ostring 03	2020-07-27	15.13	376.9 s	cloudy with sunny intervals	dry
Ostring 04	2020-07-28	8.01	364.8 s	sunny	dry

Table B.2: Sequences from 2023 with date, clock time at start, length in (non-standstill) seconds, weather and road conditions.

Sequence	Date	Time	Length	Weather	Road
Adenauer 01	2023-05-19	13.24	583.4 s	cloudy with slightly sunny intervals	dry
Adenauer 02	2023-05-19	13.37	544.4 s	cloudy with slightly sunny intervals	dry
Moltke Big 01	2023-05-19	14.52	836.0 s	overcast	dry
Moltke Big 02	2023-05-19	15.14	644.5 s	overcast	dry
Moltke Small 01	2023-05-19	15.40	529.4 s	overcast	dry
Moltke Small 02	2023-05-19	15.54	573.5 s	overcast	dry
Ostring 01	2023-05-19	13.53	581.8 s	overcast	dry
Ostring 02	2023-05-19	14.07	561.0 s	overcast	dry

Table B.3 lists the scenarios from 2020, *i.e.* parts from the sequences listed in Table B.1, that were used for hyperparameter optimization.

Table B.3: Scenarios from 2020 used for hyperparameter optimization with their length in (non-standstill) seconds and a brief description.

Sequence	Section	Length	Description
Ostring 01	Fire Brigade	10.0 s	Intersection with challenging lighting conditions and a sharp right turn.
Ostring 01	Gottesauer	10.0 s	Passing by a complex intersection.
Ostring 01	Todeskreisel	30.0 s	Complex multi-lane roundabout-like intersection with a tram crossing.
Ostring 01	Intersection	10.0 s	Complex multi-lane intersection.
Ostring 01	Ostring Street Lights	10.0 s	Straight two-lane road with very few landmarks, and mostly poles
Ostring 01	Tulla South	10.0 s	Small urban intersection that includes a tram going along.
Ostring 01	Tulla Mid	10.0 s	Small urban intersection that includes a tram going along.
Ostring 01	Tulla North	20.0 s	Typical urban intersection with tram crossing.
Ostring 03	Rintheimer	20.0 s	Multi-lane intersection with tram crossing.
Ostring 03	Ostring Curve	20.0 s	Curved two-lane road with few landmarks.
Ostring 04	Rintheimer	20.0 s	Multi-lane intersection with tram crossing.
Ostring 04	Durlacher	25.0 s	Single lane with challenging lighting conditions.
Moltke Small 01	Reinhold Frank	25.0 s	Road with few landmarks, mostly poles.
Moltke Small 01	Kaiser Wilhelm	20.0 s	Complex multi-lane intersection.

continued on the next page

Sequence	Section	Length	Description
Moltke Small 01	Mühlburger Tor	20.0 s	Many traffic lights as trams cross the road.
Moltke Small 01	Schiller	20.0 s	Two-lane road with few landmarks.
Moltke Small 01	Aral / Kaiser	30.0 s	Construction site reducing two lanes to one.
Moltke Small 01	Moltke West	20.0 s	Road with few landmarks.
Moltke Small 01	Moltke Mid	10.0 s	Road with few landmarks.
Moltke Small 01	Moltke East	30.0 s	Typical intersection and a tram crossing.
Moltke Small 04	Moltke West	20.0 s	Road with few landmarks at challenging lighting conditions.
Moltke Small 04	Moltke Mid	15.0 s	Road with few landmarks at challenging lighting conditions.
Moltke Small 04	Moltke East	40.0 s	Typical intersection and a tram crossing. Challenging lighting conditions.
Moltke Small 04	Reinhold Frank	20.0 s	Road with few landmarks, mostly poles.
Moltke Small 04	Kaiser Wilhelm	20.0 s	Complex multi-lane intersection at challenging lighting conditions.
Moltke Small 04	Rathaus West	15.0 s	Two-lane road with mainly traffic lights and poles.
Moltke Small 04	Kaiser	15.0 s	Two-lane road with few landmarks, mostly poles.

Table B.4 lists the scenarios from 2023, *i.e.* parts from the sequences listed in Table B.2, that were used for hyperparameter optimization.

Table B.4: Scenarios from 2023 used for hyperparameter optimization with their length in (non-standstill) seconds and a brief description.

Sequence	Section	Length	Description
Ostring 01	Start	15.0 s	Straight two-lane road with few landmarks.
Ostring 01	Rintheimer	20.0 s	Multi-lane intersection with tram crossing.
Ostring 01	Ostring Curve	20.0 s	Curved two-lane road with few landmarks.
Ostring 01	Small Intersection	15.0 s	Multi-lane intersection.
Ostring 01	Large Intersection	25.0 s	Complex multi-lane intersection.
Ostring 01	Ostring Street Lights	20.0 s	Straight two-lane road with very few landmarks, and mostly poles.
Ostring 01	Fire Brigade	20.0 s	Multi-lane intersection with a sharp right turn.
Ostring 01	Todeskreisel	25.0 s	Complex multi-lane roundabout-like intersection with a tram crossing.
Ostring 01	Wolfartsweierer	10.0 s	Road with few landmarks.
Ostring 01	Gottesauer	20.0 s	Passing by a complex intersection.
Ostring 01	Durlacher	15.0 s	Single lane in very wide road.
Ostring 01	Schlachthof	25.0 s	Multi-lane intersection including a tram crossing and a small construction site.
Ostring 01	Tulla South	25.0 s	Two small urban intersections that include a tram going along.
Ostring 01	Tulla Mid	30.0 s	Two medium urban intersections that include a tram going along.
Ostring 02	Rintheimer	20.0 s	Multi-lane intersection with tram crossing.

continued on the next page

Sequence	Section	Length	Description
Ostring 02	Gottesauer	25.0 s	Passing by a complex intersection.
Ostring 02	Tulla South	30.0 s	Small urban intersection that include a tram going along.
Ostring 02	Tulla North	30.0 s	Typical urban intersection with tram crossing.
Moltke Small 01	Erzberger	20.0 s	Typical intersection including a tram crossing.
Moltke Small 01	Moltke East	25.0 s	Intersection with a right turn.
Moltke Small 01	Reinhold Frank	25.0 s	Road with few landmarks, mostly poles.
Moltke Small 01	Kaiser Wilhelm	25.0 s	Complex multi-lane intersection.
Moltke Small 01	Rathaus West	15.0 s	Two-lane road with a tram crossing regulated by traffic lights.
Moltke Small 01	Kaiser	20.0 s	Two-lane road with few landmarks, mostly poles.
Moltke Small 01	Aral / Kaiser	15.0 s	Two-lane road approaching a tram station.
Moltke Small 01	Moltke West	15.0 s	Road with few landmarks.
Moltke Small 01	Moltke Mid	20.0 s	Road with a pedestrian crossing.

C Additional Evaluation Results

C.1 Parametric Detection Precision

Table C.1: Precision (root mean squared error (RMSE)) of the parametric detections against the associated map elements measured in meters and degrees, respectively. Evaluated on the validation sequences and using the 2020 sensor setup.

		(0, 20]	(20, 40]	(40, 60]	(60, 80]	(80, 100]	(100, 120]	(120, 140]	(140, 160]	(160, 180]	(180, 200]
$e_{\mathbf{c}}^{3D}$	Poles	0.39	0.33	0.40	0.53	0.43	0.52	0.55	-	-	-
	Traffic Lights	0.08	0.09	0.11	0.15	0.21	0.29	0.30	0.36	0.10	0.05
	Traffic Signs	0.07	0.06	0.08	0.09	0.12	0.16	0.15	0.13	0.10	-
$e_{\mathbf{c}}^{xy}$	Poles	0.05	0.06	0.08	0.08	0.09	0.09	0.08	-	-	-
	Traffic Lights	0.08	0.09	0.11	0.14	0.20	0.28	0.27	0.25	0.06	0.04
	Traffic Signs	0.04	0.04	0.05	0.06	0.06	0.08	0.07	0.06	0.05	-
e_w	Poles	0.02	0.03	0.03	0.04	0.05	0.08	0.08	-	-	-
	Traffic Lights	0.04	0.04	0.05	0.05	0.06	0.08	0.11	0.12	0.07	0.07
	Traffic Signs	0.07	0.08	0.09	0.11	0.14	0.16	0.13	0.10	0.08	-
e_h	Poles	0.74	0.66	0.78	1.10	0.88	0.95	1.03	-	-	-
	Traffic Lights	0.05	0.06	0.07	0.08	0.10	0.13	0.14	0.12	0.08	0.09
	Traffic Signs	0.12	0.09	0.11	0.15	0.20	0.25	0.28	0.22	0.15	-
$e_{\boldsymbol{o}}$	Poles	4.08	2.91	2.38	2.07	1.56	1.86	1.47	-	-	-
	Traffic Lights	-	-	-	-	-	-	-	-	-	-
	Traffic Signs	4.07	4.35	5.93	8.24	9.96	10.52	8.73	8.26	8.63	-

Table C.2: Precision (root mean squared error (RMSE)) of the parametric detections against the associated map elements measured in meters and degrees, respectively. Evaluated on the validation sequences and using the 2023 sensor setup.

		(0, 20]	(20, 40]	(40, 60]	(60, 80]	(80, 100]	(100, 120]	(120, 140]	(140, 160]	(160, 180]	(180, 200]	(200, 220]
$e_{\mathbf{c}}^{3D}$	Poles	0.67	0.61	0.61	0.62	0.60	0.56	0.52	0.50	0.50	0.19	-
	Traffic Lights	0.07	0.07	0.10	0.10	0.12	0.18	0.16	0.14	0.09	0.11	-
	Traffic Signs	0.07	0.05	0.07	0.08	0.09	0.10	0.10	0.12	0.14	0.19	0.26
$e_{\mathbf{c}}^{xy}$	Poles	0.06	0.04	0.05	0.05	0.05	0.06	0.06	0.05	0.07	0.08	-
	Traffic Lights	0.06	0.05	0.09	0.09	0.10	0.17	0.13	0.10	0.08	0.09	-
	Traffic Signs	0.05	0.04	0.05	0.05	0.07	0.07	0.07	0.07	0.09	0.16	0.17
e_w	Poles	0.02	0.02	0.02	0.03	0.04	0.04	0.09	0.05	0.05	0.04	-
	Traffic Lights	0.04	0.04	0.05	0.05	0.05	0.10	0.14	0.19	0.08	0.09	-
	Traffic Signs	0.18	0.09	0.08	0.09	0.10	0.12	0.12	0.10	0.12	0.20	0.15
e_h	Poles	1.19	1.18	1.19	1.25	1.26	1.06	1.04	1.08	0.93	0.45	-
	Traffic Lights	0.07	0.07	0.06	0.07	0.08	0.11	0.17	0.13	0.10	0.06	-
	Traffic Signs	0.09	0.07	0.08	0.11	0.11	0.14	0.15	0.20	0.25	0.23	0.20
$e_{\boldsymbol{o}}$	Poles	1.63	1.55	1.35	1.32	1.61	1.40	0.96	1.09	1.04	1.28	-
	Traffic Lights	-	-	-	-	-	-	-	-	-	-	-
	Traffic Signs	7.40	4.54	5.95	5.57	6.98	8.46	8.90	8.85	7.35	8.65	5.56

D On the Binarization of Correspondence Graphs

The connection between random graphs and sparse (probabilistic or binary) correspondence graphs allows to reason about the expected behavior of inlier retrieval.

Formally, one can view a PCG filtered with threshold τ as Erdős-Rényi (ER) random graph. Analogously, this holds for binary correspondence graphs with a hard threshold on the TIM Δ: $\mathcal{E}_\tau = \{e \in \mathcal{E} : \Delta(e) < \tau\}$.

An ER random graph $\mathcal{G}(n, p)$ is defined by n known vertices which are pairwise connected with probability p. This allows to reason about the asymptotic probability of retrieving the correct association as maximum clique. The probably most interesting finding for ER random graphs is that the size of a maximum clique is usually concentrated on one of two integers around $2\log_{\frac{1}{p}}(n)$ [BE76, Mat76].

As is the given problem two random graphs, one inlier graph $\mathcal{G}(k,q)$ and one outlier graph $\mathcal{G}(n-k,p)$, are combined, one can assess that for $2\log_{\frac{1}{q}}(k) \gg 2\log_{\frac{1}{p}}(n-k)$, at least the largest part of the correct association will be retrieved with the largest clique. For the exact computation of expected retrieval metrics, like precision and recall, though, the boundary between inlier and outlier subgraphs is actually most interesting. Unfortunately, to the best of the author's knowledge, computing such retrieval metrics for random graphs is an open problem – in particular when the graphs are inhomogeneous.

The inlier subgraph retrieval in such a combined random graph is also known as (uncertain) *planted clique* problem. The author disagrees with [LMK22] and argues that, given imperfect recall *e.g.* due to thresholding, the planted

inlier clique should not be considered certain. However, it can be assumed that in real cases the outlier connection probability is much lower than $p = \frac{1}{2}$, making the problem significantly easier [DGP11]. Unfortunately, the author found no literature that is for general inhomogeneous ER graphs $\mathcal{G}(n, p, k, q)$ and provides more than the probabilistic guarantees stated in [DGP11].

E Analytical Derivation of the Squared Euclidean Distance Difference

This appendix sketches how SEDDs could be derived analytically to show how TIMs can work mathematically, in particular how inliers, outliers and noise interact. For this, the data sets are assumed normally distributed with zero-mean Gaussian noise.

Squared Euclidean distance difference are defined as

$$\Delta\delta_{ijkl}^2 = \|\delta\boldsymbol{x}_{ij}\|_2^2 - \|\delta\boldsymbol{y}_{kl}\|_2^2 \tag{E.1}$$

$$= \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2 - \|\boldsymbol{y}_k - \boldsymbol{y}_l\|_2^2. \tag{E.2}$$

The sketch can be split in four parts. Appendix E.1 discusses that the resulting distribution of squared distances within each data set is a gamma mixture distribution. Appendix E.2 describes the distribution of differences between correlated and uncorrelated gamma distributions. In Appendix E.3 observations on correlation under Gaussian noise are stated. Finally, Appendix E.4 puts all parts together to derive the distribution of SEDDs. In Appendix E.5 alternatives which are closer to real problems, but lack analytical derivations, are discussed.

As none of the formulas from this appendix is used in the main part of this thesis, this appendix has differences in notation that make it closer to related works. It also allows to re-use notationally clearer symbols that have already been used previously with different meanings.

E.1 Distribution of Squared Distances

The gamma distribution arises when squaring normally distributed random variables. In this work, gamma distributions are preferred over χ^2 distributions as the latter are only valid for squares of *standard* normal distributions. Gamma distributions are parameterized by shape α and scale β, leading to the following probability density function (pdf)

$$f_{\Gamma(\alpha,\beta)}(x) = \frac{x^{\alpha-1}e^{-\frac{x}{\beta}}}{\beta^{\alpha}\Gamma(\alpha)}. \tag{E.3}$$

Gamma distributions are related to univariate zero mean normal distributions, $X \sim \mathcal{N}(0,\sigma^2)$, via the χ^2 distribution and scaling:

$$\begin{aligned} \frac{X}{\sigma} \sim \mathcal{N}(0,1) &\Rightarrow \left(\frac{X}{\sigma}\right)^2 \sim \chi_1^2 \\ &\Rightarrow X^2 \sim \sigma^2\chi_1^2 = \Gamma\left(\frac{1}{2}, 2\sigma^2\right). \end{aligned} \tag{E.4}$$

Using χ^2 distributions as TIMs is briefly discussed in Appendix E.5. Non-zero mean normal distributions lead to non-central χ^2 distributions which are not relevant for this work.

For the multivariate case, the generalized square of a random variable leads to a quadratic form of $X \sim \mathcal{N}_k(\mu,\Sigma)$ depending on the weight matrix A

$$Q(X) = X^T A X. \tag{E.5}$$

Mathai and Provost [MP92, p. 29ff.] show that this quadratic form – in the central case based on zero mean normal distributions – can be expressed as linear combination of independent random variables that each follow a χ_1^2 distribution. This linear combination is weighted with the eigenvalues λ_j of

$\Sigma^{\frac{1}{2}} A \Sigma^{\frac{1}{2}}$, yielding

$$Q(X) = U^T \text{diag}(\lambda_j) U = \sum_j \lambda_j U_j^2 = \sum_j V_j \tag{E.6}$$

where $U_j^2 \sim \chi_1^2$ is the j-th component of the diagonalization of X, $U = P^T \Sigma^{\frac{1}{2}} X$. P is the corresponding matrix of eigenvectors of $\Sigma^{\frac{1}{2}} A \Sigma^{\frac{1}{2}}$. By scaling, the quadratic form can be transformed into a sum of independent gamma random variables $V_j \sim \Gamma\left(\frac{1}{2}, 2\lambda_j\right)$.

For the unweighted quadratic form, i.e. $A = I$, this yields

$$Q(X) = X^T X = \|X\|_2^2 = \sum_j V_j \tag{E.7}$$

with $V_j \sim \Gamma\left(\frac{1}{2}, 2\lambda_j\right)$ where λ_j are the eigenvalues of Σ.

Finally, in the case of an uncorrelated underlying normal distribution with zero mean, $X \sim \mathcal{N}_k\left(0, \text{diag}\left(\sigma_i^2\right)\right)$, this is $V_j \sim \Gamma\left(\frac{1}{2}, 2\sigma_i^2\right)$.

E.2 Gamma Difference Distributions

In general, *i.e.* for false / outlier correspondences, the difference of two i.i.d. random variables from a gamma distribution follows a so-called *gamma difference distribution* [Mat93, Kla15].

The difference of two *i.i.d.* random variables from a gamma distribution can be shown to be a special case of the variance gamma (VG) distribution [Kla15].

For identical shape α, but possibly different scale parameters β_i, the difference of two *independent* gamma distributed random variables $X_i \sim \Gamma(\alpha, \beta_i)$ follows so-called *Bessel function distribution* [McK32], also called *Type II McKay distribution* [HA04], $\Delta X = X_1 - X_2 \sim \Delta\Gamma(\alpha, \beta_1, \beta_2)$, with probability

density function (pdf)

$$f_{\Delta\Gamma(\alpha,\beta_1,\beta_2)}(x) = \frac{\sqrt{\frac{1}{\beta_1}+\frac{1}{\beta_2}}|x|^{\alpha-\frac{1}{2}}\mathrm{e}^{\frac{1}{2}\left(\frac{1}{\beta_1}-\frac{1}{\beta_2}\right)|x|}}{\sqrt{\pi}\Gamma(\alpha)(\beta_1+\beta_2)^{\alpha}} \cdot K_{\alpha-\frac{1}{2}}\left(\frac{|z|}{2}\left(\frac{1}{\beta_1}+\frac{1}{\beta_2}\right)\right) \tag{E.8}$$

for $x \neq 0$ [Mat93, HA04]. Here, $K_\alpha(\cdot)$ is the modified Bessel function of second kind.

The difference of gamma distributed random variables that differ in both shape and scale is examined in [Mat93, Kla15], but not relevant for this work. The interested reader might want to note the fact that the author was not able to confirm the pdf stated in [Kla15], but found agreement between [Mat93] and [HA04] for the special case of identical shape parameters.

The case for inlier correspondences is more complicated since this leads to *correlated* variables that only differ by the additive noise (*cf*. Appendix E.3). The lack of similar results for *e.g*. Nakagami distributions is in fact the reason that only the derivation for exact distributions for *squared* EDDs is stated in this thesis. As explained in Appendix E.4, they follow correlated gamma distributions for which the difference is known.

Holm and Alouini [HA04] describe the case of two *correlated* gamma random variables $\overline{X}_{1,2}$ with identical shape, but possibly different scale parameters is considered. The resulting random variable $\overline{\Delta X} = \overline{X}_1 - \overline{X}_2$ additionally depends

on the correlation coefficient ρ, leading to the notation $\overline{\Delta\Gamma}(\alpha,\beta_1,\beta_2,\rho)$ with pdf

$$\begin{aligned} f_{\overline{\Delta\Gamma}(\alpha,\beta_1,\beta_2,\rho)}(x) = & \frac{|x|^{\alpha-\frac{1}{2}}}{\Gamma(\alpha)\sqrt{\pi}\sqrt{\beta_1\beta_2(1-\rho)}} \\ & \cdot\left(\frac{1}{(\beta_1+\beta 2)^2-4\beta_1\beta_2\rho}\right)^{\frac{2\alpha-1}{4}} \\ & \cdot\exp\left(\frac{x}{2(1-\rho)}\left(\frac{1}{\beta_2}-\frac{1}{\beta_1}\right)\right) \\ & \cdot K_{\alpha-\frac{1}{2}}\left(|x|\frac{\sqrt{(\beta_1+\beta 2)^2-4\beta_1\beta_2\rho}}{2\beta_1\beta_2(1-\rho)}\right) \end{aligned} \tag{E.9}$$

for $x \neq 0$ [HA04].

E.3 Correlation under Gaussian Noise

One may assume both data sets that should be associated to be from a normal distribution, *i.e.* $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}_X, \Sigma_X)$. Hence, the difference of two points within the same set is from a zero-mean normal distribution with doubled covariance, *i.e.* $\delta\boldsymbol{x}_{ij} := \boldsymbol{x}_i - \boldsymbol{x}_j \in\sim \mathcal{N}(0, 2\Sigma_X)$.

By the assumptions that regards one set as true and the other as noisy with noise magnitude Σ_N, one can then derive that

$$\delta\boldsymbol{y} := \delta\boldsymbol{x} + \boldsymbol{\varepsilon}, \boldsymbol{\varepsilon} \sim \mathcal{N}(0, 2\Sigma_N). \tag{E.10}$$

Now, when neglecting rotations, one can observe empirically that the correlation coefficient asymptotically follows

$$\rho(\delta\boldsymbol{x}, \delta\boldsymbol{y}) \approx \sqrt{\frac{\Sigma_X}{\Sigma_X + \Sigma_N}}. \tag{E.11}$$

Analogously, the correlation coefficient for the squared distances asymptotically follows

$$\rho\Big(\|\delta\boldsymbol{x}\|_2^2, \|\delta\boldsymbol{y}\|_2^2\Big) \approx \frac{\Sigma_X}{\Sigma_X + \Sigma_N}. \tag{E.12}$$

E.4 Distribution of *Squared* EDDs $\Delta\delta_{ijkl}^2$

Now it can all be put together to show that under these assumptions the squared distances follow a gamma mixture distribution and that the $\Delta\delta_{ijkl}^2$ follow either a $\overline{\Delta\Gamma}$ or a $\Delta\Gamma$ distribution.

With the knowledge established in Appendix E.1, one can now formulate the distribution for $\|\delta\boldsymbol{x}_{ij}\|_2^2$ which depends on the eigenvalues λ_p^X of $2\Sigma_X$

$$\|\delta\boldsymbol{x}_{ij}\|_2^2 = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2 \sim \sum_p \Gamma\left(\frac{1}{2}, 2\lambda_p^X\right). \tag{E.13}$$

Similarly, for a transformation $T = (R, \boldsymbol{t})$, the distances between noisy detections follow a gamma mixture distribution ruled by the eigenvalues λ_p^Y of $2\Sigma_Y = 2R(\Sigma_X + \Sigma_N)R^T$

$$\|\delta\boldsymbol{y}_{kl}\|_2^2 = \|\boldsymbol{y}_k - \boldsymbol{y}_l\|_2^2 \sim \sum_p \Gamma\left(\frac{1}{2}, 2\lambda_p^Y\right). \tag{E.14}$$

If both (i,k) and (j,l) are true / inlier correspondences, this makes $\Delta\delta_{ijkl}^2$ follow a mixture of *correlated* gamma difference distributions

$$\begin{aligned}\Delta\delta_{ijkl}^2 &= \|\delta\boldsymbol{x}_{ij}\|_2^2 - \|\delta\boldsymbol{y}_{kl}\|_2^2 \\ &\sim \sum_p \overline{\Delta\Gamma}\left(\frac{1}{2}, 2\lambda_p^X, 2\lambda_p^Y, \rho\right)\end{aligned} \tag{E.15}$$

with $\rho \gg 0$ as derived in Equation (E.12).

In contrast, if either (i,k) or (j,l) is a false / outlier correspondence, they exhibit uncorrelated distances that follow a mixture of *independent* gamma difference distributions

$$\begin{aligned} \Delta\delta_{ijkl}^2 &= \|\delta\boldsymbol{x}_{ij}\|_2^2 - \|\delta\boldsymbol{y}_{kl}\|_2^2 \\ &\sim \sum_p \Delta\Gamma\left(\frac{1}{2}, 2\lambda_p^X, 2\lambda_p^Y\right). \end{aligned} \tag{E.16}$$

E.5 Alternative Distributions

Due to exact results about differences of correlated distributions, only the derivation for *squared* EDDs was sketched. Empirically, original, *i.e.* unsquared, EDDs $\Delta\delta$ were found to provide better separability between inliers and outliers, so an exact formulation of their cdf and pdf depending on Σ_X and Σ_N would be desirable.

Normalization would allow to subtract correlated χ^2 distributions whose difference has been found exactly [Fer19]. But, again, for real-world problems, normalization was found to hurt separability for non-isotropic noise.

Ideally, one could imagine a transformation invariant distribution which (provably) optimally separates inliers and outliers, but also has its cdf and pdf exactly known for both independent and correlated terms.

F Hyperparameter Optimization for Data Association

This appendix describes how the hyperparameters of PCG for evaluation on point cloud registration and data association in HAD maps are optimized. The general idea, *i.e.* using SMAC [LEF+22] as state-of-the-art framework for hyperparameter tuning, is presented in Section 3.8.

F.1 Point Cloud Registration

For point cloud registration, the hyperparameters comprise the scale of inlier and outlier distributions, width of the similarity distribution, weight of correspondences, inlier outlier ratio, number of sampled correspondences, and the threshold for edges to exist in a PCG. To optimize them, three loss terms are combined in one loss function

$$J_{PCR} = \begin{cases} 1 - \overline{F_1} + 555^2 - RR \cdot B^2 & \text{if } \overline{t_{eval}} > 0.3\,\mathrm{s} \\ 2\big(1 - \overline{F}_1 + 555^2 - RR \cdot B^2\big) & \text{otherwise} \end{cases}. \tag{F.1}$$

Using the multi-fidelity budget $B \in [1, 555]$, *i.e.* the number of lidar scan pairs used for the current evaluation, this puts the overall registration recall (RR) as primary goal. Basically, the more lidar scan pairs are registered successfully, the lower the loss. Mean correspondence inlier F_1 score, $\overline{F_1}$, acts as secondary goal that improves both translational and rotational error. To penalize excessively slow parametrizations, the loss is doubled if the mean evaluation time, $\overline{t_{eval}}$, is larger than $0.3\,\mathrm{s}$.

Unfortunately, the optimization procedure comes with two restrictions. First, since there is no training/test split, the optimization used the same data that is used for evaluation. Additionally, to obtain deterministic results, the random seed was kept fixed for both optimization and evaluation. While both restrictions definitely exaggerate the performance on unseen data, details like losses and fixed random seeds in open source code of other state-of-the-art approaches as well as the evaluation of other approaches by the author hint that these restrictions are the only possibility to obtain numbers even close to reported state-of-the-art results.

F.2 Data Association in HAD Maps

The result of the data association is used for two tasks at once, localization in HAD maps and their verification. This motivates a joint loss function which optimizes RIIoU, pose error, and the evidence of verification. Together, the RIIoU of the map given the localization result and the pose error evaluate the performance for the localization task.

By simulating changes, the localization can be made robust to work reliably even in partially outdated maps. At the same time, the map verification performance can be measured by comparing the evidence each landmark to be changed or unchanged with ground truth. This motivates a loss for localization and verification, J_{LV}, with four components that makes it possible to optimize one common hyperparameter set for both localization and map verification

$$J_{LV} = \frac{1}{N} \sum_{i=1}^{N} J_{\mathrm{mRIIoU},i} + J_{\mathrm{DPE},i} + J_{\mathrm{ev},i} + J_{\mathrm{time},i}. \tag{F.2}$$

The components are defined as by

$$J_{\mathrm{mRIIoU},i} = 1 - \mathrm{mRIIoU}(\mathtt{R}(\mathcal{M}, \mathfrak{i}_i, T_i), \mathfrak{M}_{\mathfrak{i}_i}) \tag{F.3}$$

$$J_{\mathrm{DPE},i} = \beta_{\mathrm{DPE}} \left(\delta e^{xy}_{i-1,i} + \beta_\varphi \delta e^{\varphi}_{i-1,i}\right)^2 \tag{F.4}$$

$$J_{\mathrm{ev},i} = \frac{1}{\mathcal{M}_i} \sum_{\boldsymbol{\ell} \in \mathcal{M}_i} \begin{cases} \beta_{\mathrm{nev}}\, m_{\mathrm{VMA}} & \text{if } \boldsymbol{\ell} \text{ changed} \\ m_{\mathrm{VMA}} & \text{if } \boldsymbol{\ell} \text{ unchanged} \end{cases} \tag{F.5}$$

$$J_{\mathrm{time},i} = \beta_{\mathrm{time}}\, t_{eval,i}. \tag{F.6}$$

The prefactor $\beta_\varphi = 10$ brings yaw and 2D translational error in the same order of magnitude. Scalars $\beta_{\mathrm{DPE}} = 25$ and $\beta_{\mathrm{time}} = 0.1$ are chosen manually to balance the loss terms. Using $\beta_{\mathrm{nev}} = 100$ punishes false verification significantly higher than its opposite.

If localization is not successful due to self-assessment, $J_{\mathrm{mRIIoU},i}$ is computed by dead reckoning using lidar odometry and $J_{\mathrm{ev},i}$ is assumed zero. This allows to judge performance for applications where localization is complemented with odometry, making a correctly estimated orientation more important than position accuracy. $J_{\mathrm{DPE},i}$ gets assigned a default value which is very sensitive but crucial since it trades localization accuracy against availability. If chosen larger, the self-assessment is relaxed since avoiding localization is more expensive and vice versa. $J_{\mathrm{time},i}$ is independent from the availability of localization.

The restriction of $J_{\mathrm{DPE},i}$ to planar coordinates is that these are most important for automated driving and, in contrast to z position and pitch angle, less prone to bad observability.