On Compositional Hierarchical Models for holistic Lane and Road Perception in Intelligent Vehicles

Zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften

der Fakultät für Maschinenbau
Karlsruher Institut für Technologie (KIT)
genehmigte

Dissertation

von

DIPL.-ING. DANIEL TÖPFER

aus Braunschweig

Tag der mündlichen Prüfung:	18. November 2014
Hauptreferent:	Prof. Dr. Christoph Stiller
Korreferent:	Prof. Dr. J. Marius Zöllner

Die Ergebnisse, Meinungen und Schlüsse dieser Dissertation sind nicht notwendigerweise die der Volkswagen AG.

Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als Doktorand in der Abteilung Fahrerassistenz und Integrierte Sicherheit der Volkswagen Konzernforschung.

Zunächst möchte ich mich herzlich bei meinem Doktorvater Herrn Prof. Christoph Stiller für die Betreuung, die Anregungen zu dieser Arbeit und für die Übernahme des Hauptreferats bedanken. Mein Dank gilt außerdem Herrn Prof. Marius Zöllner für die Übernahme des Korreferats.

Weiterhin bedanke ich mich bei Herrn Thomas Ruchatz und Dr. Jan Effertz für das in mich gesetzte Vertrauen und die mir gestatteten Freiheiten.

Einen wesentlichen Einfluss auf diese Arbeit hatten meine Kollegen, Praktikanten und Diplomanden. Vielen Dank für die wissenschaftlichen Diskussionen, die entstandenen Freundschaften und die fröhlichen Momente. Im speziellen danke ich Dr. Jens Spehr für die Einführung in die zauberhafte Welt der hierarchischen Modelle und die beispiellose Betreuung. Danke! Dr. Jan Aue danke ich neben der tollen Zusammenarbeit auch für die willkommene nicht-akademische Ablenkung.

Ganz besonderer Dank gilt meiner Familie, die mich immer gefördert hat und mir ein sorgenfreies Studium ermöglichte. Aus tiefstem Herzen danke ich meiner Frau Eri für ihr Verständnis, ihre Geduld und die Motivation, ohne die diese Arbeit nicht möglich gewesen wäre.

Braunschweig, 2014 *Daniel Töpfer*

Zusammenfassung

Die Fähigkeit die Umgebung des Fahrzeugs zur Erfassen und zu Verstehen ist eine der Schlüsseltechnologien für das automatische Fahren und moderne Fahrerassistenzsysteme. Beide dieser Systeme legen ihren Entscheidungen die Ergebnisse der sensorischen Umfelderfassung zugrunde, die sich mit der Erkennung bewegter Objekte (z.B. Pkw und Fußgänger) und der Erkennung von Fahrstreifen und Fahrbahnen beschäftigt. Hierbei stellt die sensorische Erfassung von Fahrstreifen und Fahrbahnen eine besonders große Herausforderung dar, da Geometrie und Topologie von Straßen häufig sehr komplex sind und in verschiedenen Umgebungen stark variieren. Weiterhin sind die mit Sensoren gewonnen Umfelddaten häufig verrauscht, mehrdeutig und weisen fehlende Daten aufgrund von Verdeckungen auf.

Diese Arbeit stellt einen neuartigen Ansatz zur zuverlässigen Wahrnehmung mehrspuriger Fahrbahnen in urbanen Umgebungen vor. Dies umfasst das Bestimmen der Anzahl, der Lage und der Breite von Fahrstreifen, sowie das Erkennen paralleler, sich aufspaltender und sich verengender Fahrstreifen. Ein zentraler Aspekt sind hierbei die verwendeten hierarchischen probabilistischen Modelle. Im Kontext der Fahrbahnerkennung bieten diese den Vorteil, dass sie Vorwissen über die Topologie von Fahrbahnen in den Wahrnehmungsprozess einzubeziehen ohne harte Annahmen zu treffen. Stattdessen werden diese Annahmen mittels weicher probabilistischer Beschränkungen ausgedrückt, was die explizite Modellierung räumlicher Unsicherheiten ermöglicht. Darüber hinaus erlauben hierarchische Modelle eine Zerlegung komplexe Straßenszenen in elementare Bestandteile wie Fahrbahnen, Fahrstreifen, Fahrstreifensegmente und schließlich beobachtbare Fahrstreifenmerkmale. Diese hierarchische Zerlegung erlaubt es das komplexe Wahrnehmungsproblem in einfacher zu lösende Teilprobleme zu zerlegen, was die Basis für die Echtzeitfähigkeit des vorgeschlagenen Ansatzes bildet.

Am wichtigsten ist allerdings, dass diese Arbeit die erste Anwendung hybrider hierarchischer Modelle für die Erkennung von Kreuzungsanfahrten vorstellt. Diese hybriden hierarchischen Modelle ermöglichen es neben der Topologie von Kreuzungsanfahrten auch die Lage von Haltelinien und die Abbiegerichtung einzelner Fahrstreifen zu bestimmen. Grundlage für diese komplexe Wahrnehmungsaufgabe ist die Fusion verschiedener Umfeldmerkmale, zu denen Fahrstreifenmarkierungen, Fahrbahnränder, Haltelinien und Abbiegepfeile zählen.

Abstract

The ability of sensing and understanding the vehicle environment is a key technology for autonomous driving and driver assistance systems. Both of these systems base their decisions on the environment perception results obtained by on-board sensing. Generally, the environment perception aims to detect dynamic objects (e.g., vehicles and pedestrians) as well as lanes and roads. In this context, the perception of lanes and roads is one of the major challenges, since the geometry and topology of roads is often complex and may vary significantly depending on the environment. Moreover, the observations obtained by on-board sensing are likely to be noisy, ambiguous and to suffer from occlusions.

This thesis presents a novel approach for the reliable perception of multi-lane roads in urban environments. This includes the estimate of the number, the location and the width of lanes as well as the detection of parallel, splitting and merging lanes. A key aspect of the proposed approach is the application of hierarchical probabilistic models. These models have the advantage, that they allows to incorporate prior knowledge on the road topology in the perception process without imposing hard constraints. Instead, prior expectations on the lane topology are expressed through weak probabilistic constraints which allow to account for spatial uncertainties. Furthermore, the proposed hierarchical models allow for the decomposition of complex road-scenes into fundamental elements, such as multi-lane roads, lanes, lane segments and finally observable lane cues. This hierarchical decomposition allows to divide the complex perception task into sub-tasks that are easier to solve. This, in turn, is the basis for the real-time application of the proposed approach.

Most importantly, however, this thesis presents the first application of hybrid hierarchical models for holistic lane and road perception at urban intersections. Thereby, the hybrid hierarchical models provide the possibility to simultaneously estimate the topology of intersection roads as well as the location of stoplines and the turn direction of individual lanes. The basis for this challenging perception task is the fusion of multiple lane cues, including lane markings, road edges, stoplines and turn arrows.

Contents

Notation and Symbols

This chapter introduces the abbreviations, 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.

Abbrevations

e.g.,	for example *(latin : exempli gratia)*
i.e.,	that is *(latin : id est)*
ADAS	Advanced Driver Assistance System
BEV	Bird's Eye View
BN	Bayesian network
CHM	Compositional Hierarchical Model
CPT	Conditional Probability Table
(D)GPS	(Differential) Global Positioning System
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
IMU	Inertial Measurement Unit
KDE	Kernel Density Estimation
LIDAR	Light Detection And Ranging
(N)BP	(Nonparametric) Belief Propagation
MAP	Maximum a Posteriori Probability
MCMC	Markov Chain Monte Carlo
MRF	Markov Random Field
PDF	Probability Density Function
PF	Particle Filter
PR	Precision-Recall
RANSAC	Random Sampling Consensus

General Notation

Scalars	Regular (greek) lower case: a, b, c, σ, λ

Vectors	Bold (greek) lower case: $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{\sigma}$, $\boldsymbol{\lambda}$
Matrices	Bold upper case: $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{\Sigma}$, $\boldsymbol{\Lambda}$
Sets	Calligraphic upper case/Bold lower case: $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{z}$
Distributions	Calligraphic upper case: $\mathcal{N}(\cdot), \mathcal{M}(\cdot), \mathcal{U}(\cdot)$
Numbers	Blackboard uppercase: $\mathbb{N}$, $\mathbb{R}$

Numbers

$\mathbb{N}$	Natural numbers
$\mathbb{R}$	Real numbers

Subscripts and Superscripts

i	First-order index $i \in \mathbb{N}$
j	Second-order index $j \in \mathbb{N}$
f	Superscript denoting a feature variable
p	Superscript denoting a patch variable
l	Superscript denoting a lane variable
r	Superscript denoting a road variable

Symbols and Geometry

$\propto$	Proportional
$\mathcal{O}(\cdot)$	Runtime complexity
$\boldsymbol{x}^{\mathrm{T}}$	Transposed of the vector $\boldsymbol{x}$
$\boldsymbol{X}^{-1}$	Inverse of the matrix $\boldsymbol{X}$
d	Euclidean distance $d \in \mathbb{R}^+$
l	Length $l \in \mathbb{R}^+$
w	Width $w \in \mathbb{R}^+$
ν	Constant $\nu \in \mathbb{R}^+$
ν_{l_p}	Patch length $\nu_{l_p} \in \mathbb{R}^+$
ν_{w_p}	Patch width $\nu_{w_p} \in \mathbb{R}^+$
(x, y)	2D position in the vehicle coordinate system $(x, y) \in \mathbb{R}^2$
ϑ	Angle in the vehicles coordinate system $\vartheta \in [0; 2\pi)$
$\boldsymbol{r}_{ij}$	Relative configuration vector $\boldsymbol{r}_{ij} \in \mathbb{R}^n$

k^{l_c}	Location of a center feature along the lane centerline $k^{l_s} \in \mathbb{R}^+$
k^{r_c}	Location of a center feature along the road centerline $k^{r_s} \in \mathbb{R}^+$
$\boldsymbol{\lambda}_i$	Vector defining the offset between the centerline of a road and its lane i-th lane

Graph Theory

$\mathcal{G}$	Graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$
$\mathcal{E}$	Set of edges in a graph $\mathcal{G}$
$\mathcal{V}$	Set of nodes in a graph $\mathcal{G}$
$\mathcal{F}$	Set of factor nodes in a factor graph $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{F})$
$\Upsilon(i)$	Neighbors of node $i \in \mathcal{V}$
$\Xi(i)$	Children of node $i \in \mathcal{V}$
$\Gamma(i)$	Parents of node $i \in \mathcal{V}$

Sensory Evidence

$\boldsymbol{b}$	Set of boundary features $\boldsymbol{b} = \{\boldsymbol{m}, \boldsymbol{r}\}$
$\boldsymbol{m}$	Set of lane marking features $\boldsymbol{m} = \{\boldsymbol{m}_1, \ldots, \boldsymbol{m}_{N_m}\}$
$\boldsymbol{r}$	Set of road edge features $\boldsymbol{r} = \{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{N_r}\}$
$\boldsymbol{t}_b$	Discrete boundary feature types
$\boldsymbol{c}$	Set of center features $\boldsymbol{c} = \{\boldsymbol{a}, \boldsymbol{s}\}$
$\boldsymbol{a}$	Set of arrow features $\boldsymbol{a} = \{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{N_a}\}$
$\boldsymbol{s}$	Set of stopline features $\boldsymbol{s} = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_{N_s}\}$
$\boldsymbol{t}_c$	Discrete center feature types

Compositional Hierarchical Model

$p(\cdot)$	Probability
$\log\ p(\cdot)$	Logarithmic probability
$p(\cdot\|\cdot)$	Conditional probability
$b_i(\cdot)$	Belief
$b_i^-(\cdot)$	Bottom up belief
Z	Normalization constant $Z \in \mathbb{R}$
$\boldsymbol{\mathcal{G}}$	Set of hierarchies $\boldsymbol{\mathcal{G}} = \{\mathcal{G}_1, \ldots, \mathcal{G}_{N_h}\}$

$\mathcal{G}$	Compositional hierarchical model
$\mathcal{L}_n$	n-th level of a compositional hierarchical model
$\mathcal{I}$	Set of clique indexes
$\boldsymbol{x}_i$	Continues random variable associated to node i
$\boldsymbol{x}$	Set of continues random variables $\boldsymbol{x} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N_x}\}$
$\boldsymbol{d}_i$	Discrete random variable associated to node i
$\boldsymbol{d}$	Set of discrete random variables $\boldsymbol{d} = \{\boldsymbol{d}_1, \ldots, \boldsymbol{d}_{N_d}\}$
$\boldsymbol{z}_i$	Hybrid random variable associated to node i
$\boldsymbol{z}$	Set of hybrid random variables $\boldsymbol{z} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{N_z}\}$
$\phi_i(\cdot)$	Observation potential at node i
λ^0	Outlier probability
$\boldsymbol{\Sigma}_0$	Covariance Matrix of outlier process
$\psi_{i,j}(\cdot)$	Spatial constraint between nodes i and j
$S_{i,j}(\cdot)$	Function predicting the mean of node j given the state of node i
$\eta_{i,j}(\cdot)$	Type constraint between nodes i and j
$F_{i,j}(\cdot)$	Function predicting the mean of node j given the state of node i
$\varphi_{ij}(\cdot)$	Marginal influence of node i on node j
$m_{i,j}(\cdot)$	Message send from node i to node j
$\mu, \boldsymbol{\mu}$	Mean, mean vector
$\sigma^2, \boldsymbol{\Sigma}$	Variance, covariance matrix
$\mathcal{U}(\cdot)$	Uniform distribution
$\mathcal{N}(\cdot)$	Gaussian distribution
$\mathcal{M}(\cdot)$	Von Misses Fisher distribution
$I_0(\cdot)$	Modified Bessel function of first kind and order zero
τ	Concentration parameter
γ	Periodic variable
$\delta(\cdot)$	Dirac delta function
$\boldsymbol{s}_i^{(k)}$	Sample
$\pi_i^{(k)}$	Importance weight
$\Omega(\boldsymbol{s}_k^{(i)})$	Area associated to a sample $\boldsymbol{s}_k^{(i)}$
$\kappa(\cdot)$	Kernel
$h, \boldsymbol{\Lambda}$	Bandwidth

1 Introduction

This chapter introduces and motivates the thesis and outlines the key ideas and contributions. The chapter also introduces the problem of lane and road perception. Challenges in this problem are discussed along with motivations for solving them. Further, the chapter gives an overview of the overall thesis structure.

1.1 Environment Perception in Intelligent Vehicles

Advanced Driver Assistance Systems (ADAS), which aim to either alert the driver in certain situations or support the driver while driving, are available in most new vehicles. Thereby, these systems not only aim to increase the driving comfort but to reduce accident rates and to increase the traffic flow.

Examples for the first generation of commercial systems are the adaptive cruise control [Mar01], lane keeping assist and parking assist, all aiming to support the driver to increase the driving comfort. A typical characteristic of these systems is that they support the driver, while he is controlling the vehicle. More recent systems perform more complex tasks, such as autonomous lane changes or autonomous parking [Kab08]. These examples show the recent shift from assisting systems to semi-autonomous [1] systems that take over the vehicle control for specific driving maneuvers. During the next decade, this trend is assumed to continue and therefore ADAS are expected to grow more complex towards fully autonomy.

Although the step towards fully autonomous driving seems small, the impression that this step is easy is misleading. In fact, building a systems meeting the enormous reliability demands of commercial applications is a large-scale research and development effort. The probably biggest challenge in the development of such systems is the environment perception problem.

Environment perception describes a very wide concept, but two main tasks can be identified with respect to autonomous driving. First, the perception of lanes and roads. Second, the detection of obstacles, such as vehicles and pedestrians. This thesis addresses the first.

Road and lane perception systems for semi and fully autonomous vehicles aim to infer the same amount of knowledge about lanes and roads as human drivers. As a consequence, such systems are expected to rely on the same perceptual cues

[1] Semi-autonomous ADAS are systems that perform autonomous driving maneuvers but rely on the supervision of a human driver.

as human drivers, i.e., road color, road texture, road boundaries, and lane markings. In principal, different infrastructure could be used for human drivers and vehicles, such as lane markings for humans and some kind of car-to-infrastructure communication for autonomous vehicles. This approach, however, is unrealistic since it requires huge investments to be made for both the construction and the maintenance of such redundant infrastructure. Therefore, on-board lane and road perception based on the traditional cues remains the most likely solution for autonomous driving.

This thesis considers the task of on-board lane and road perception for autonomous driving, which includes detecting the topology and semantic of lanes and roads. In this context, topology refers to the extend of the road, the number and position of lanes, merging, splitting and ending lanes and roads [Hil12]. In contrast, the semantics of roads refers to e.g., stopline locations on lanes and lane turn directions. While this level of scene understanding is natural to human drivers, it is beyond the reach of current perception systems (cf. Chapter 2).

Before the proposed approach is detailed, the following section describes open research gaps and details the challenges addressed in this thesis. Thereafter, a more detailed overview of the proposed approach is given.

1.2 Challenges

Lane and road perception, at least in its basic setting, seams to be an easy task. In this setting, the task is simply to detect the host lane for a relatively short distance ahead. In fact, this basic task has been solved in the late eighties by Dickmanns et al. [Dic88]. While similar perception approaches are still used in commercial applications, such as lane keeping assist [Bar12b], those systems make strong assumptions on the structure of roads, i.e., a smooth and continuous curvature, parallel lanes and well-defined lane markings, limiting their applicability to highways and highway-like scenarios.

Significant research effort was also devoted to applications targeting more complex urban scenarios [Urm08, Kna10a]. A recent research project targeting such urban scenarios is the project INTERSAFE-2. The goal of this project was the development of ADAS for urban intersection, such as a stopline assistant and a left turn assistant. Both of these systems require a high level of scene understanding. For example, the stopline assistant requires to detect the host lane and the location of the stopline along this lane. More challenging, however, is the left turn assistant that requires to detect multiple lane with non-linear lane topologies (i.e., splitting lanes) as well as to identify turn lane directions. As these systems were already realized, one may get the impression that the corresponding environment perception problems are solved. This, however, is not the case if it comes to lane and road

perception. The reason for this is that the lack of coast constraints, together with the availability of highly accurate digital-maps, led to solutions with very limited or even no forms of on-board lane and road perception.

A typical vehicle in the INTERSAFE-2 project carries a stereo-vision system, multiple Light Detection and Ranging (LIDAR) and radar sensors, a highly accurate Differential Global Positioning System (DGPS), an Inertial Measurement Unit (IMU) and a precise digital-map. The digital-map contains detail information on the road network, such as the number and position of lanes, the location of stoplines and the turn direction of lanes [Kna10b, Hom11]. The combination of map data and the positioning equipment (DGPS+IMU) provide information with an accuracy in the sub-meter range. Therefore, the combination of these two information sources makes any kind of lane and road perception unnecessary.

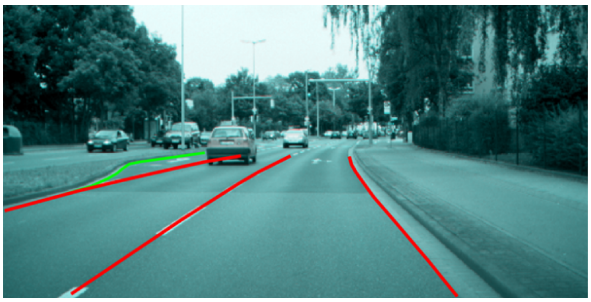
In contrast to the global positioning based solutions developed during the INTERSAFE-2 project, lane and road perception for commercial vehicles has to be accomplished with affordable sensors. Consequently, for most current vehicles the sensor modalities are limited to vision, LIDAR, GPS and different radar types. These considerations as well as the reliability issues of systems based on global positing discussed in Section 2.1.5, imply that lane and road understanding remains a challenge that should be solved by on-board sensing.

In recent years, considerable progress has been made in the field of on-board lane and road perception. However, many fundamental questions are still unanswered. The main research gaps remaining are the detection of multiple lanes, the estimation of complex non-linear lane and road topologies (e.g., splitting and merging lanes) and the identification of lane semantics (e.g., turn lane directions and stopline locations). The latter of which is particularly important to understand intersections.

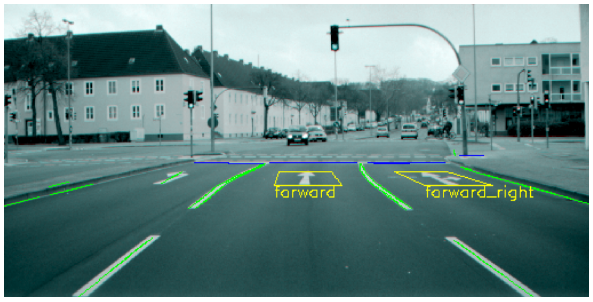
While these tasks are already an enormous challenge, commercial lane and road perception systems are required to handle diverse scenarios (e.g., well structured highways as well as semi structured rural and urban roads) and have to allow for real-time computation. Especially, the diversity of target scenarios is an enormous challenge in developing a general lane and road perception approach, since roads often differ in their topology as well as in their appearance. This is illustrated in Figure 1.1, showing typical road topologies and their appearance diversity on the top. The appearance diversity is particularly challenging from a perception point of view, because a change in appearance also affects the available lane and road cues.

For example, while lane markings are a strong cue for structured highways, lane marking cues are expected to be less reliable in less structured urban or rural scenarios. Therefore, a combination of lane markings with other lane or road cues, such as curbstones or guardrails is required to achieve reliable results. Using multi-

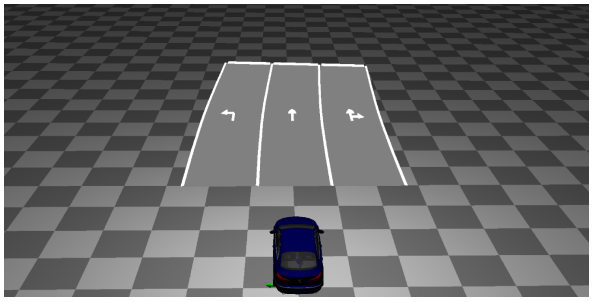
(a) Highway



(b) Urban Road



(c) Low-Level Input Features



(d) Estimated Road Topology and Semantic

Figure 1.1: Challenges in lane and road perception. (top) Roads are inherently structured as highlighted in red. However, they often differ in geometry and topology as highlighted in green. (bottom) Estimating high-level road scene knowledge from low-level cues is a key challenge in lane an road perception.

ple cues is not only required to increase the reliability of road topology estimation, but to recognize semantic road properties. For example, roads at intersections often comprise additional markings in the center of lanes, such as stoplines and turn arrows, where the latter of which have different classes, i.e., forward, left, right, forward-left and forward-right. In order to capture the semantic meaning of these features a holistic lane and road perception approach, as proposed in this thesis, has to incorporate both lane boundary features and lane center features to allow for the joint estimation of the topology and semantic of roads.

The probably hardest challenge in lane and road perception, however, is to develop an approach that accounts for the erroneous sensory evidence. In fact, each lane and road cue has to be expected to suffer from clutter, measurement noise and classification errors, as shown in Figure 1.1. Hence, any on-board lane and road perception approach is required to take into account that the sensory knowledge about the vehicles environment is imperfect.

This thesis addresses exactly the above challenges and thereby presents a novel unified probabilistic framework for holistic lane and road perception, as detailed in the next section.

1.3 Thesis Contributions

This thesis makes contributions to the problem of lane and road perception to solve the problems of detecting scene topologies and semantics in varying environments (i.e., highways, rural roads and urban roads). Towards this goal, this thesis presents a novel compositional hierarchical framework that outperforms existing algorithms (in terms of computational efficiency and scalability) for reliable lane and road perception in real-time and in the presence of uncertainty (including partial occlusion and erroneous sensory evidence). The main contributions of this thesis are:

- A novel compositional hierarchical model of multi-lane roads is introduced. In contrast to existing approaches, this model is compositional and generic in the sense that it does not impose any hard constraints on the lane geometry as imposed by e.g., clothoids or splines. Instead, a priori expectations on the lane geometry are expressed through weak probabilistic constraints, and lanes are assembled from a large number of lane patches.

- A novel hybrid (discrete/continuous) compositional hierarchical model of intersection roads is presented, which in contrast to previous approaches, is able to detect both complex road topologies and scene semantics, i.e., the turn direction of lanes and the positions of stoplines along lanes.

- Compared to existing approaches, the presented approach is applicable to varying road topologies by representing them in a set of compositional hierarchical road models. Thereby, part-sharing [Zhu10, Spe11] is employed which allows to account for similarities of different road topologies and thus to avoid redundant computations.

- In contrast to previous approaches, the presented model combines multiple features (e.g., lane markings, road edges, stoplines or turn-arrows) for increased performance and robustness. Thereby, it takes into account different turn arrow types (e.g., forward, left or right) and lane boundary types (e.g., solid, dashed or curbstone).

- Building on the depth-first message passing algorithm presented in [Spe13], a novel real-time lane detection algorithm [Töp13] is presented. This algorithm dynamically initiates inference in region of strong belief and performs message passing in several sequential sweeps. Empirical results show that this inference algorithm requires significantly lower computation for a performance comparable to classic message passing.

- A novel expectation based message passing algorithm is introduced that accounts for the performance diversity of different feature extraction ap-

proaches. This algorithm adopts the order in which features are incorporated into the detection process for increased efficiency and performance.

Most important, this thesis presents the first application of hybrid compositional hierarchical models to holistic lane and road perception in urban intersections.

For all the above aspects, it is shown that the proposed hierarchical framework obtains reliable results both qualitatively and quantitatively, while achieving real-time performance.

1.4 Thesis Outline

The following chapters are organized as follows:

Chapter 2: Related Work This chapter presents and discusses recent works in the field of lane and road perception. It provides an overview on the sensor modalities and the cues used for the task of lane and road perception. Further, it outlines typical model assumptions regarding the longitudinal and lateral road structure.

Chapter 3: Graphical Models and Inference This chapter briefly summarizes the formalism and the statistical methods used in this thesis. First a brief introduction of probability theory and graph theory is provided. Then, probabilistic graphical models [Jor04, Jor01] are introduced, including undirected, directed and factor graphs. Moreover, the probability distributions underlying the nodes of the graphical models used in this thesis are presented. Towards this goal the properties of parametric and nonparametric probability distributions are detailed. Finally, common inference methods for continuous and discrete state graphical models, along with approximate inference algorithms are introduced.

Chapter 4: Compositional Hierarchical Models of Multi-Lane Roads This chapter presents the key contribution of the proposed compositional hierarchical model of multi-lane roads [Töp13]. Further, it details how part-sharing and depth-first message passing allow for real-time applications.

Chapter 5: Hierarchical Lane and Road Perception for heterogeneous Road Scenes This chapter introduces the notion of extending the hierarchical framework to scenarios with varying road topologies using a set of hierarchies and part-sharing [Töp13]. Further, it provides the formulation of the novel hybrid hierarchical representation of lanes and roads.

Chapter 6: Hierarchical Approach for Holistic Lane and Road Perception at Urban Intersections This chapter presents the first application of hybrid hierarchical compositional models to intersection scenes. The proposed hybrid framework allows to infer both complex non-linear lane topologies and lane semantics, such as the turn direction of lanes and the location of stoplines along lanes.

Chapter 7: Experimental Evaluation This chapter presents a set of quantitative and qualitative results, demonstrating the reliability of the presented approach in various real world scenarios. As part of this effort, a novel evaluation framework for lane and road perception is proposed. The key benefit of this evaluation framework is that it allows to evaluate the results of the proposed approach without exhaustive manual labeling of the visual input.

Chapter 8: Conclusion This chapter summarizes the contributions of this thesis, discusses open issues and outlines promising future lines of research.

2 Related Work

This chapter discusses related work in the field of lane and road perception as well as scene understanding approaches in the context of intelligent vehicles. First, Section 2.1 introduces common sensor modalities used for the task of lane and road perception. The following sections outline lane and road detection approaches, which typically aim to detect the topology of roads, including the number of lanes, the position of lanes and the extend of the road.

In general, lane and road detection approaches can be decomposed in functional modules, i.e., image pre-processing, feature extraction, model fitting and time integration [McC06, Hil12], as shown in Figure 2.1. In order to abbrevi-

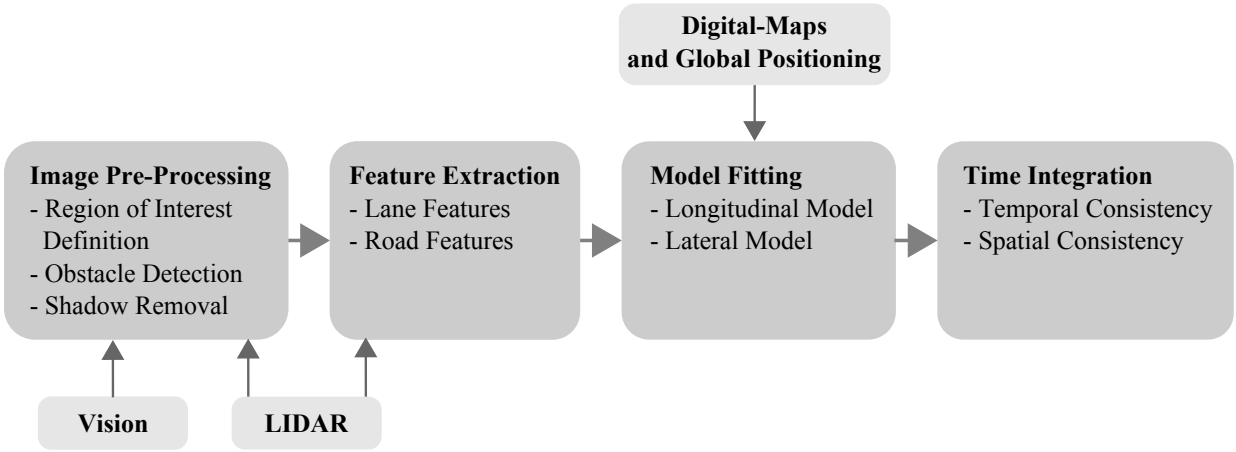


Figure 2.1: Generic system decomposition for lane and road perception approaches [McC06, Hil12].

ate the following exposition, Section 2.2 mainly focuses on the feature extraction module and Section 2.3 on the model fitting module, as they are the most relevant for this thesis. Time integration is not discussed, since it is not addressed in this thesis. Finally, Section 2.4 covers approaches in the field of scene understanding which aim not only to estimate the topology of roads, but also to infer more complex scene properties, such as the relative location of vehicles and road infrastructure [Gei11, Spe13].

2.1 Sensors for Lane and Road Detection

As discussed in the previous chapter there exists a large variety in topology and appearance of target scenarios for lane and road detection. Consequently, different sensors were employed for the task of lane and road perception. This section, briefly reviews their main characteristics and typical usage.

2.1.1 Monocular Vision

The most frequently used system for lane and road perception is monocular vision which consists of a single camera that is typically front mounted in the middle of the car. Commonly, it is placed in the wiping area of the windscreen to ensure a clear field of view. The application of vision-based systems has two main reasons. First, the main modality used by human drivers are visual data, hence the road infrastructure is designed to support the driver while navigating through the road network. The road infrastructure commonly comprises lane markings and road boundaries which are designed to support the driver during its visual perception tasks. Therefore, it makes great sense to use vision-based perception approaches for lane and road perception [Hil12, Dic92]. Second, cameras became a mass product, and thus they are comparable inexpensive. Further by applying different machine vision approaches, cameras can be used as a multi-purpose sensor for various applications (e.g., lane keeping assistance, adaptive distance control, emergency brake systems and traffic sign detection).

2.1.2 Light Detection and Ranging

Light Detection And Ranging (LIDAR) is an active time of flight device that can measure the 3D environment of the vehicle. Compared to vision sensors LIDAR sensors have two main advantages. First, they are an active light source and thus are less affected by e.g., shadow or complete darkness. Second, LIDAR provides measurements with high range accuracy up to several hundred meters. However, since lane markings commonly have no 3D structure, LIDAR-based lane marking detection often relies only on intesity measures [Die05b].

Nonetheless, LIDAR sensors provide very accurate range information and therefore are well suitable for the segmentation of road and non-road areas [Hom12] or for detecting elevated road-edges, such as curbstones or berms [Man13, Kon10, Kon11, Die05b]. Thus, in contrast to 2D approaches, 3D approaches allow for greater robustness in detecting the 3D structure around the vehicle, including obstacles (e.g., parking or driving cars) or the road shoulder (e.g., guard rail, curbstone or berms).

2.1.3 Stereo Vision

Stereo vision systems comprise two cameras to provide visual data similar to mono vision as well as 3D range data. Hence, stereo systems can be considered a combination of mono vision and LIDAR in a single sensor which allows for small packaging. However, the accuracy of the 3D range data obtained by stereo systems crucially depends on their baseline [Bro13]. While a larger baseline allows for more accurate range information, it also leads to additional computational complexity and a larger packaging. Therefore, in contrast to LIDAR sensors, stereo-based range data commonly provide good accuracy only for a close range, which limits their area of application. Nonetheless, stereo systems are an ideal multi-purpose sensor for detecting multiple lane and road cues, such as lane markings, curbstones and the general 3D structure of roads, as detailed in [Dan09].

2.1.4 Radar

Radar is less commonly used for lane and road detection, since it lacks the power of detecting both lane marking and 3D structure. Nevertheless, radar sensors are used to detect other vehicles which may hint for the presence of lanes in unstructured areas or on snowy roads [Wei12, Ada11].

2.1.5 Digital-Maps and Global Positioning

Since the DARPA urban challenge, the combination of digital-maps and a known ego-vehicle position is used to guide autonomous vehicles with limited perception capabilities [Urm08]. Similarly, in the more recent INTERSAFE-2 project a combination of digital-maps and global positioning is used to introduce Advanced Driver Assistance Systems (ADAS) for urban intersections [Kna10b, Hom11]. A general disadvantage of these systems, however, is that they have high demands regarding the accuracy and reliability of both Global Positioning System (GPS) and map information.

Commercial GPS receivers obtain a localization accuracy of 5-10 m [Win05]. This accuracy can be improved to less than 1 m using a combination of GPS and an Inertial Measurement Unit (IMU) [Car06]. Promising sources for highly accurate map data are aerial images or GPS measurements collected by ground vehicles [Rau12]. For example, it was shown in [Urm08] that the resolution of map data obtained form aerial images can be 0.25 cm and higher. Accordingly, the main accuracy gap results from the global positioning system.

The most critical aspect of systems relying on GPS and map data is the reliability of both modalities. For example, the reliability of GPS crucially depends on the visibility of a large number of satellites. The visibility of satellites, however,

can be interrupted by e.g., bridges on highways or high-building in urban areas. In order to cope with this temporal loss of satellite connections, often GPS is integrated with IMU systems, which is a current line of research [Car06]. The most questionable aspect, however, is if highly accurate map data can be provided for a large area and be kept up to date.

The bottom line is that a commercial usage of global positioning and map data as the only modalities for lane and road perception is not expected. Thus, lane and road perception based on on-board sensing, as proposed in this thesis, remains the most likely solution for commercial ADAS applications.

2.2 Cues for Lane and Road Detection

The main goal of feature extraction (see Figure 2.1) is to reduce the huge amount of information contained in the visual data. The most fundamental requirement on a feature is that it should contain all relevant data for supporting the task of lane and road detection. Since, commonly different feature cues are used to define lanes and roads, highly different features are considered for lane and road perception. Therefore, in the following, two feature types are explicitly distinguished. First, lane boundary features, such as solid or segmented lane markings [McC06]. Second, road features, such as intensity profiles, color, texture or edge features [Fra07, Wan13, Alv12, Kue11, Woj08]. Further, approaches are discussed that use a combination of features in order to archive the required robustness in challenging rural and urban scenarios.

2.2.1 Lane Cues

Commonly lanes are defined by a left and a right lane marking. A particular challenge during lane marking detection arises from their varying appearance. In fact, their shape may vary between continuous lines and dashed lines. Additionally, their color is subject to change, e.g., white or yellow. However, if lane markings can be detected they are a strong hint for the presence of lanes.

Extraction algorithms for local lane marking features are generally based on geometric and photometric lane marking characteristics [Vei08]. Often, these criteria are combined in different ways. Both criteria are e.g., used in positive-negative gradient extractors [Klu95b] which aim to find gradients along an expected marking geometric. On the other hand, photometric approaches assume a light lane marking on a dark background. This leaded to approaches based on e.g., global thresholds [Vei08], local thresholds or symmetric local threshold [Cha97]. Whereas, the two local threshold methods aim to handle non-uniform lightning conditions by analyzing the intensity within a finite area [Guo12a]. Even though

local threshold methods are mainly based on photometric properties they may employ additional geometric criteria.

Besides the detection of lane boundary markings, also the detection of lane markings located in the lane center, such as stoplines [Ned09] and turn arrows [Dan10] have gained increasing interest. In [Ned09] stopline detection is performed based on local thresholds. Similarly, in [Dan10] a gradient-based arrow detection is performed on horizontal line regions to find arrow candidates. These candidates are further refined using a grey level segmentation. Finally, a decision tree is used to classify their type, i.e., forward, left, right, forward-left and forward-right.

2.2.2 Road Cues

In order to provide reliable results on sparsely marked or even unmarked roads, as typical for rural and urban areas, several approaches have been proposed that avoid the explicit need for lane markings. The key challenge in these approaches is that different cues may hint for the location of road boundaries, depending on the target environment. For example, on highways roads may be bounded by guardrails or barriers. Urban roads, on the other hand, may be bounded by curbstones. In addition, some rural and urban roads may not have markings or even elevated road shoulder and thus only differ from non-road areas by their color or texture. Consequently, a variety of road features is used for the task of road detection.

A common assumption in road detection is that an elevation gap separates the road from the non-road area [Dan09, Kon11, Sie11]. Based on this assumption, a stereo-system is used in [Dan09] to extract the three-dimensional structure of the scene. A similar approach is taken in [Sie11] where curbstones and road surface are detected based on the assumption that curbstones form a vertical structure separating road surface and adjacent horizontal surfaces, such as sidewalks and traffic isles. In [Kon11] the 3D scene structure is directly detected using a LIDAR sensor and used for road segmentation. Note that, all the above approaches expect the road surface to be planar. Other approaches focus on the diversity in appearance between road and non-road areas [Fra07, Kuh11]. However, these approaches rely on the strong assumption that the road appearance is uniform which is often not the case [McC06]. An exception is the texture-based approach presented in [Bro06, Ras05] that use the footprints caused by wheels of preceding vehicles to detect the geometrical structure of unpaved roads. Another road characteristic is that the boarders of paved roads generate visible edges. However, since road edges are often weak and ambiguous, it is beneficial to fuse them with other cues as proposed in [Fra07].

2.2.3 Multi-Cue Approaches

One common characteristic of many early lane and road detection approaches is that they rely on a single cue. This cue is used regardless of how well it is performing. Thus, these approaches make no attempt to account for changing lane and road characteristics at the fundamental level of perception, i.e., from a well-structured highway to semi-structured lanes of rural or urban roads.

In order to achieve the level of robustness, required for commercial systems targeting such challenging scenarios, lane and road detection approaches have been proposed that combine multiple lane and road cues [Apo04, Fra07, Dan09]. A multi-cue detection approach dedicated to rural roads is presented in [Fra07] where the visual input is segmented in road and non-road areas. This segmentation is performed by computing intensity, color, edges and texture for each pixel in the image. Since the combination of multiple features leads to multimodal probability distributions for both lane states and observations a Particle Filter (PF) [Isa98a] is applied to solve for the non-Gaussian lane detection problem. Similarly, a PF is used for multi-cue lane detection in urban scenarios in [Dan09]. This approach considers three different features obtained by a stereo vision system, i.e., gradient based lane marking extractions, curbstones obtained by stereo-based range data and edge features extracted from the 3D range data, which approximate the road area.

All the methods mentioned above allow for the incorporation of multiple lane and road cues. Similarly, this thesis is concerned with the task of multi-cue lane and road perception to robustly estimate the course of multi-lane roads. In contrast to the above approaches, the presented approach explicitly considers different feature classes, such as lane marking types (e.g., dashed or continuous) or turn arrow types (e.g., left or right). This allows to estimate not only the topology of roads, but also semantic road properties, such as is a lane change allowed or the turn direction of lanes.

2.3 Models for Lane and Road Detection

Many lane and road detection approaches developed so far follow the principle introduced by Dickmanns [Dic87] where a geometric lane model is fitted to the visual features in a top down manner to obtain a high-level representation of lanes and roads. Such models are used for both lanes and roads and are commonly defined as a two-dimensional path with a left and a right boundary. A path is commonly represent by its boundary points or its centerline where each of the centerline positions may have an additional lane width extend [Duc10]. The key idea of model fitting is to obtain a compact high-level representation of the path.

Further, it aims to improve the erroneous lane and road features by restricting them to a smooth path [Dic92, Pom95] with additional constraints on its width and curvature. This model fitting is performed in both the image domain and the Bird's Eye View (BEV) [Mal91]. However, model fitting in the BEV is more common, as the transforming of the frame into the BEV simplifies the fitting procedure. Additionally, any driving maneuver or vehicle control task is performed in the metric 2D space of the BEV. Note that, to transform a visual frame into the BEV commonly a flat road surface is assumed.

As detailed in the following sections, the models for the driving path can be divided into two categories. First, longitudinal models, which describing the course of lanes and roads by means of e.g., curvature and curvature rate. Second, longitudinal models that specify the dependencies between lanes and lane boundaries (e.g., parallel/constant lane width, splitting or merging). During model fitting each approach has to cope with the erroneous feature extraction results in the form of missing data and a large number of outliers (e.g., clutter). In order to cope with these challenges, many approaches employ Random Sampling Consensus (RANSAC) [Fis81] for model fitting [Dan09, Kim08, Bor09], as it is known to be robust to outliers. An exception to the above examples is the approach presented in [Hur13]. Instead of detecting lanes based on model fitting, this approach employs a conditional random field to detect lane boundaries based on local dependencies between lane markings. A common property of many model fitting approaches is that their input data is a set of points extracted from the boundaries of lanes and roads. In addition, the information of the boundary direction is added in [Hua09]. Other approaches first group points into lines [Kim08] to reduce computational complexity during model fitting.

2.3.1 Geometrical Lane Models

Since its first proposal in [Dic88] the clothoid model is the most popular geometrical model used for lane and road detection. The reason for its popularity is twofold. First, most highways and highway-like roads are constructed according to clothoid models (see e.g., [ras95]). Accordingly, a clothoid model allows for a good representation of the actual road geometry and therefore is expected to archive a high robustness during model fitting. Second, clothoid models are differentiable and continuous which makes them well suitable for any vehicle control applications [Beh96], such as lane keeping or lane change assistance [Bar12b].

However, clothoid models make strong assumptions regarding a smooth road curvature and are therefore limited to highways and highway-like roads [Sch03]. As the target environments became more challenging, the application of more general models has been proposed, such as splines [Loo10] or snakes [Wan04,

Kim08], parabolic segments [Klu95a, Hua09], line elements [Apo03] or deformable templates [Klu95a]. Splines are smooth piecewise polynomial functions and are commonly used to represent curves. Several spline models have been used to represent both lane centerlines and lane boundaries. In spline models, the curve is parameterized by a set of control points which are either on [Kim08] or near [Wan04] the curve. The advantage of splines over e.g., clothoids is that a small change of their parameters also causes a small change of their curvature (see e.g., [Hil12]. However, the most crucial part about spline fitting is to find the best control point candidates among the erroneous lane marking detections. Further, the complexity of splines increases drastically with the number of control points.

Most of the above lane perception approaches are designed for a specific geometrical model. In contrast, the proposed model is general in the sense that it is not limited to a specific geometrical model and thus applicable to a greater variety of scenarios.

2.3.2 Lateral Lane Models

In multi-lane detection, the proposed algorithms detect not only the ego-lane, but also adjacent lanes. These approaches commonly make several assumptions on the lateral lane structure. A common assumption made by early approaches is that the width of lanes is constant, i.e., lane boundaries are parallel [Pom95, Dic92]. This strong assumption expects the width of lanes to be known (typically between 3.0-3.5 m) and completely couples left and right lane boundaries. Similarly, in more recent works, an a priori lane width is used as an initial value for a RANSAC-based lane centerline fitting [Hua09] that accounts for small variations. A weaker assumption is that the width of individual lanes is approximately constant and may not vary for a finite area of the road. Boundary constraints following this assumptions are e.g. used in [Duc10] where roads segments are used to support lane boundary detection. While each individual segment has a constant lane-width between each pair of segments a variation of the lane-width is allowed to account for variations of the lane width, as common in urban intersections.

The above approaches explicitly or implicitly assume adjacent lanes to be parallel. While this strong assumption allows for both robustness and simplicity, it also limits their applicability to highway or highway-like scenarios. In order to cope with more complex lane and road topologies, including merging and splitting lanes, approaches have been proposed making fewer assumptions on the road structure [Hur13, Kim08]. Both approaches follow the strategy to first detect the lane boundaries of individual lanes, based on local dependencies. Then, in [Kim08] a probabilistic reasoning approach is used to estimate the lane path by choosing the the best pair of boundary hypothesis. In [Hur13], the low-level de-

pendencies are used to formulate a conditional random field that infers the position and course of lane boundaries. In [Hur13], the individual boundary hypothesis are considered as the detection result, and thus no dependencies between lane boundaries are considered. As the presented results show, this lack of contextual knowledge leads to poor performance in scenarios where clutter arises in the lane center.

All aforementioned projects are targeted at tasks such as highway driving, lane-keeping/-following or overtaking. In contrast, this thesis addresses the more challenging task of lane and road perception at urban intersections which are much more complex in terms of topology and geometry. Most important, the presented approach aims to estimate semantic knowledge on intersection roads, such as the turn directions of individual lanes and the location of stoplines. Such semantic knowledge can not be estimated by either of the above approaches.

2.4 Scene Understanding

While typical lane and road detection approaches mainly aim to estimate the topology of lanes and roads in a given scene, scene understanding approaches aim to infer a richer amount of scene knowledge. More precisely, scene understanding approaches aim to detect the scene topology as well as e.g., the presence of parking [Spe11] or activities of vehicles driving in the scene [Gei11]. Many approaches treat scene understanding as a segmentation problem. Bileschi [Bil06] proposes a method which segments street scenes in classes, such as cars, pedestrians, roads and trees using a biologically inspired image representation. A conditional random field is used by Wojek et al. [Woj08] to jointly perform object detection and scene labeling. Sturgess et al. [Stu09] developed a segmentation of road scenes based on appearance cues and structure-from-motion features. Another segmentation approach is presented in Ess et al. [Ess09]. Their traffic scene segmentation allows them to assign semantic labels like road types, cars or pedestrian crossings to individual segments.

Existing approaches for high-level scene understanding often use generative graphical models. Wang et al. [Wan09] propose a hierarchical Bayesian network to perform activity detection in traffic scenes from a static platform. Interdependent Dirichlet processes are used in [Kue10] to understand the behavior of moving objects in the scene. A generative model for 3d scene interpretation was proposed by Wojek et al. [Woj10]. Their model jointly performs multi-class object detection, object tracking, scene labeling and 3d geometric relations. For inferring about 3d scene context as well as 3d multi-objects a reversible-jump Markov Chain Monte Carlo (MCMC) scheme is employed. Geiger et al. [Gei14] also proposed the use of reversible-jump MCMC to infer geometrical, topological properties of scenes as well as semantic activities.

Closely related to the proposed approach is the work of Spehr et al. [Spe11, Spe13] where a compositional hierarchical scene understanding approach for parking lots is proposed. They proposed a hierarchical decomposition of a parking-lot scene into geometrical primitives like u-shapes and l-shapes. These primitives are again decomposed into simple observable line features. Thus, the compositional hierarchical model encodes the dependencies of low-level features and high-level scene properties. Further, it allows to represent dependencies between different scene components in a probabilistic graphical model. Most important, however, it allows to share parts between different scenes which allows for real-time computation.

Similarly, this thesis proposes a compositional hierarchical model for multi-lane roads. However, the proposed approach considers more general road scenes and as part of this effort propose a hybrid compositional framework, which allows to estimate the geometry and topology of roads as well as semantic properties, such as the turn direction of lanes.

3 Graphical Models and Inference

A key component of this thesis are probabilistic graphical models (see Figure 3.1), since they explicitly take into account that our knowledge about the environment is imperfect. The following sections, introduce different kinds of graphical models, including Markov random fields (Section 3.2.2), Bayesian networks (Section 3.2.4) and factor graphs (Section 3.2.5). Further, different parametric and nonparametric probability distributions underlying the graphical models used in this thesis (Section 3.3) are discussed. In addition, it is detailed how graphical models can be used to infer knowledge about quantities of interest, given some noisy observations (Section 3.4). Finally, illustrative example that relates the introduced methods to the hierarchical graphical models used throughout this thesis are presented.

3.1 Brief Review of Graph Theory

A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a set of nodes or vertices $\mathcal{V}$ and a set of edges $\mathcal{E}$. Each edge $(i, j) \in \mathcal{E}$ is associated with a distinct pair of nodes $i \in \mathcal{V}$ and $j \in \mathcal{V}$. Edges may be directed or undirected. In a directed graph an edge $(i, j) \in \mathcal{E}$ has a particular direction pointing from the parent node i to the child node j. In this case edges are commonly depicted as an arrow (see Figure 3.1a). In the undirected case an edge (i, j) has no direction and hence is identical to the edge (j, i) (see Figure 3.1b). Some other terms used throughout this thesis are:

Parent For a directed graph, the parents $\Gamma(j)$ of the node j are given by $\Gamma(j) = \{i \in \mathcal{V} | (i, j) \in \mathcal{E}\}$.

Child For a directed graph, the children $\Xi(j)$ of the node j are given by $\Xi(j) = \{k \in \mathcal{V} | (j, k) \in \mathcal{E}\}$.

Neighbors In the directed case the neighbors $\Upsilon(i)$ of node i can be defined by $\Upsilon(i) = \Xi(i) \cup \Gamma(i)$. In the undirected case the $\Upsilon(i)$ are those nodes directly connected to node i.

Clique A clique is a set of nodes that are all neighbors of each other. A clique is called maximal clique if it can not be extended by including any more node without ceasing it to be a clique.

Path A path from node i to j is a sequence of nodes with an edge from one node in the sequence to its descendant.

Cycle, Loop A cycle is a directed path that starts and ends at the same node. A path with more then two nodes is called a loop, if regardless of the edge direction it starts and returns to the same node.

Singly Connected If there is only one path from any node i to any node j a graph is called singly connected.

Tree, Forest Trees are singly connected graphs. A forest is a disjoint union of trees.

Root In a directed graph, the root is a node with no parents.

Leaf In a directed graph, the leaf is a node with no children.

This section focuses on concepts of graph theory relevant to this thesis. A more general introduction to graph theory is given in [Gib91, Die05a].

3.2 Probabilistic Graphical Models

Probabilistic graphical models, as shown in Figure 3.1, have a wide application in statistical physics, pattern recognition, machine learning and computer vision. Generally, they allow to decompose complex joint probability distributions over various variables into a product of smaller and simpler subsets of theses variables by introducing conditional independence assumptions [Bar12a, Mur06, Bis06].

Formally, a graphical model is defined as a graph $G = (\mathcal{V}, \mathcal{E})$ where each node $i \in \mathcal{V}$ corresponds to a random variable $\boldsymbol{x}_i$. An edge $(i, j) \in \mathcal{E}$ corresponds to a probabilistic relationship between two random variables $i \in \mathcal{V}$ and $j \in \mathcal{V}$. Hence, a graphical model is a graph $\mathcal{G}$ that represents a joint probability distribution $p(\boldsymbol{x})$ by encoding the conditional independence of its variables $\boldsymbol{x} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$.

In order to illustrate the importance of independence assumptions, let us consider the joint distribution $p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$ over binary variables $\boldsymbol{x}_i$. Independently specifying the table $p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$ requires 2^N entries. Accordingly, computing the marginal $p(\boldsymbol{x}_i)$ requires summing over the 2^{N-1} states of the other variables. Even for a small number of variables storage and manipulation of the density is clearly infeasible. The key idea of introducing conditional independence properties is to restrict the possible ways variables can interact which leads to efficient inference algorithms. Before introducing the different kinds of graphical models, the following section reviews some fundamental rules of probability theory.

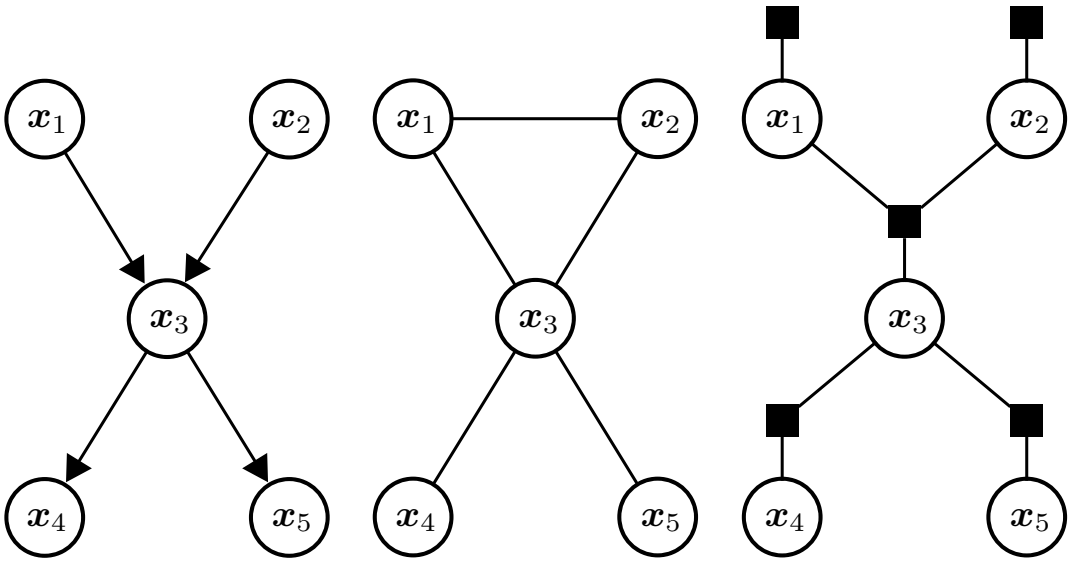


Figure 3.1: Graphical models of different families. (a) Bayesian network, (b) Markov random field (moralized) and (c) factor graph. All graphs represent the same distribution $p(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4, \boldsymbol{x}_5)$ [Pea88, Sud06, Spe13].

3.2.1 Fundamentals of Probability Theory

The concepts and rules of probability theory provide the framework to solve the tasks addressed in this thesis. It helps to structure problem, and is the basis for formulating graphical models. This section, introduces important rules of probability theory.

Joint Probabilities

Given two Events A and B the probability of the joint event can be written as

$$p(A, B) = p(A|B)\, p(B)\,, \tag{3.1}$$

which is commonly called the product rule. In order to calculate the marginal distribution $p(A)$, it has to be summed over all possible states of B as

$$p(A) = \sum_b p(A, B) = \sum_b p(A|B = b)\, p(B = b)\,. \tag{3.2}$$

This is often referred to as the sum rule or the rule of total probability. Similarly, the marginal distribution $p(B)$ can be obtained. In the case of continuous distribution, the sum in Equation 3.2 has to be replaced by an integral as in Equation 3.6.

Applying the product rule multiple times, yields the chain rule of probability

$$p(\boldsymbol{x}_{1:N}) = p(\boldsymbol{x}_1)p(\boldsymbol{x}_2|\boldsymbol{x}_1)p(\boldsymbol{x}_3|\boldsymbol{x}_1,\boldsymbol{x}_2) \\ p(\boldsymbol{x}_4|\boldsymbol{x}_1,\boldsymbol{x}_2,\boldsymbol{x}_3)\ldots p(\boldsymbol{x}_N|\boldsymbol{x}_{1:N-1}), \tag{3.3}$$

where the notation $1:N$ refers to the set $\{1,2,\ldots,N\}$.

Conditional Probability and Rule of Bayes

The conditional probability of event A given event B is defined as

$$p\left(A|B\right) = \frac{p\left(A,B\right)}{p\left(B\right)}. \tag{3.4}$$

The relationship between conditional probabilities is given by the rule of Bayes

$$p\left(A|B\right) = \frac{p\left(B|A\right)p\left(A\right)}{p\left(B\right)}, \tag{3.5}$$

where $p\left(A|B\right)$ is the posterior distribution and $p\left(B|A\right)$ is the likelihood that event A caused event B. The marginal distribution $p\left(A\right)$ provides information about A without knowing B and is called prior of A. The evidence or marginal likelihood $p(B)$ is a normalization term, which is given by

$$p\left(B\right) = \int_a p\left(B|A\right)p\left(A\right)da. \tag{3.6}$$

Independence and Conditional Independence

If the joint distribution $p\left(A,B\right)$ can be expressed as the product of two marginals

$$p\left(A,B\right) = p\left(A\right)p\left(B\right) \tag{3.7}$$

then A and B are unconditional independent or marginal independent. However, this is not the case in most real world applications. More common are distribution where A and B are conditionally independent given C. This is the case if the joint distribution can be written as a product of joint marginals, i.e.,

$$p\left(A,B|C\right) = p\left(A|C\right)p\left(B|C\right). \tag{3.8}$$

Such conditional independence assumptions are used to construct graphical models, which are introduced in the following sections.

3.2.2 Undirected Graphical Models

Markov random fields (MRFs), can be visualized by an undirected graphical model $G = (\mathcal{V}, \mathcal{E})$. Here, edges $\mathcal{E}$ and nodes $\mathcal{V}$ represent probabilistic relationships and random variables of the MRF, respectively. Thereby, edges connect pairs of random variables and are undirected (see. Figure 3.1b). The constraints or local dependencies between random variables are represented by potentials, where each potential $\psi(\boldsymbol{x})$ is a non-negative function $\psi(\boldsymbol{x}) \geqslant 0$ of the variable $\boldsymbol{x}$. Potentials are not necessarily a probability distribution. In fact, a potential satisfying $\sum_{\boldsymbol{x}} \psi(\boldsymbol{x}) = 1$ is considered as a special case of a potential.

Conditional independence properties in MRFs

In order to explain conditional independence in MRFs, let us consider an undirected graph $G = (\mathcal{V}, \mathcal{E})$ with three disjoint subsets of nodes $\mathcal{A} \in \mathcal{V}, \mathcal{B} \in \mathcal{V}$ and $\mathcal{C} \in \mathcal{V}$. In this example, variables $\mathcal{A}$ are independent of $\mathcal{B}$ in the graph G iff $\mathcal{C}$ separates $\mathcal{A}$ and $\mathcal{B}$ in the graph G. Intuitively, this means after removing all nodes in $\mathcal{C}$ there is no path left connecting $\mathcal{A}$ and $\mathcal{B}$. This is called the global Markov property, which is given by

$$p(\boldsymbol{x}_{\mathcal{A}}, \boldsymbol{x}_{\mathcal{B}}|\boldsymbol{x}_{\mathcal{C}}) = p(\boldsymbol{x}_{\mathcal{A}}|\boldsymbol{x}_{\mathcal{C}})p(\boldsymbol{x}_{\mathcal{B}}|\boldsymbol{x}_{\mathcal{C}}). \tag{3.9}$$

An example for the global Markov property is depicted in Figure 3.1b, where the graphical structure indicates that $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are conditional independent of $\boldsymbol{x}_4$ and $\boldsymbol{x}_5$ given $\boldsymbol{x}_3$.

Another important property is the local Markov property, which implies that a variable $\boldsymbol{x}_i$ is conditional independent of all other variables $\boldsymbol{x}_{\mathcal{V}\setminus i}$ given its neighbors $\boldsymbol{x}_{\Upsilon(i)}$

$$p(\boldsymbol{x}_i|\boldsymbol{x}_{\mathcal{V}\setminus i}) = p(\boldsymbol{x}_i|\boldsymbol{x}_{\Upsilon(i)}). \tag{3.10}$$

In addition, the pairwise Markov property states that two nodes i and j are conditionally independent given all other nodes if there is no connecting edge

$$p\left(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{x}_{\mathcal{V}\setminus i,j}\right) = p\left(\boldsymbol{x}_i|\boldsymbol{x}_{\mathcal{V}\setminus i,j}\right) p\left(\boldsymbol{x}_j|\boldsymbol{x}_{\mathcal{V}\setminus i,j}\right). \tag{3.11}$$

As can be seen, global Markov implies local Markov which implies pairwise Markov. Further, it was shown in [Kol09] that for $p(\boldsymbol{x} > 0)$, which is the case if all potentials are strictly positive, all three Markov properties are equivalent.

Hammersley-Clifford theorem

The Hammersley-Clifford theorem [Cli90] states that a positive distribution $p(\boldsymbol{x}) > 0$ satisfies the conditional independence properties of an undirected Graph

G with a set of maximal cliques $\mathcal{C}$, iff $p(\boldsymbol{x})$ can be represented as a product of factors, each corresponding to a maximal clique $c \in \mathcal{C}$, i.e.,

$$p(\boldsymbol{x}) = \frac{1}{Z} \prod_{c \in \mathcal{C}} \psi_c(\boldsymbol{x}_c), \tag{3.12}$$

where Z is a real valued number given by

$$Z = \sum_x \prod_{c \in \mathcal{C}} \psi_c(\boldsymbol{x}_c) \tag{3.13}$$

is known as partition function, which ensures the overall distribution sums to 1. The most important statement of the Hammersley-Clifford theorem, however, is the equivalency of MRF and the Gibbs distribution which can be expressed as

$$p(\boldsymbol{x}) = \frac{1}{Z} \exp\left(-\sum_c E(\boldsymbol{x}_c)\right). \tag{3.14}$$

Here, $E(\boldsymbol{x}_c) > 0$ is an energy associated with the variables in clique c. The corresponding MRF can be defined by using clique-potentials

$$\psi_c(x_c) = \exp\left(-E(\boldsymbol{x_c})\right). \tag{3.15}$$

In such energy based models, low energies correspond to high probability states.

3.2.3 Pairwise Markov Random Fields

In many applications, as in this thesis, it is convenient to use pairwise Markov random fields instead of the more general MRFs, where cliques are restricted to pairs of random variables connected by the edges of the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. For pairwise MRFs, the joint probability distribution is expressed as a product of potential functions defined on that graph's edges as

$$p(\boldsymbol{x}) \propto \prod_{(i,j) \in \mathcal{E}} \psi_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j). \tag{3.16}$$

The Hammersley-Clifford theorem guarantees that pairwise MRFs[1] are Markov with respect to $\mathcal{G}$, because neighboring nodes always form a clique.

So far, it was assumed that all random variables in the graphical model are hidden. This goal of this thesis, however, is to estimate the values of hidden variables

[1]The origin of pairwise MRFs is the physical statistics, where pairwise MRFs with only binary variables are known as Ising models [Yed00].

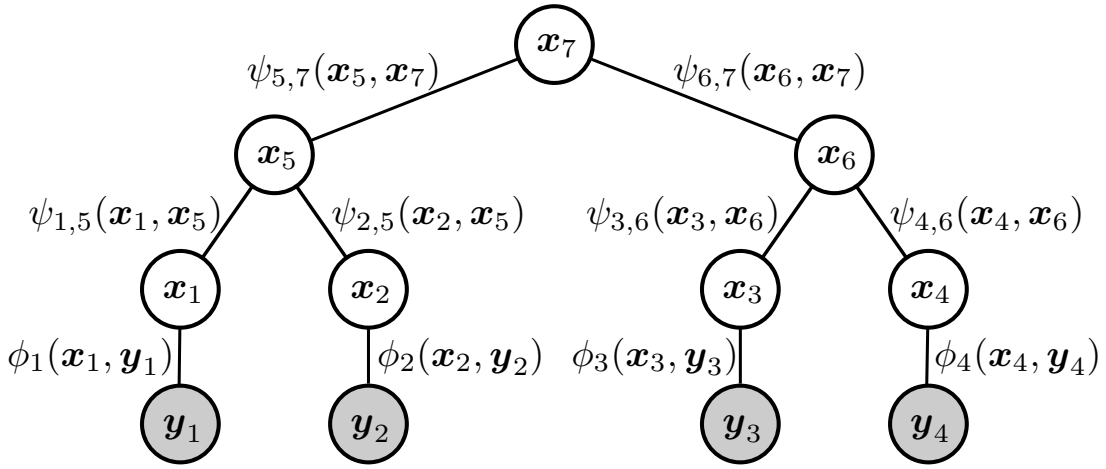


Figure 3.2: Pairwise Markov random field. Shaded nodes represent observable random variables $\boldsymbol{y}_i$ and open nodes hidden random variables $\boldsymbol{x}_i$. Further illustrated are observation potentials $\phi_i(\boldsymbol{x}_i, \boldsymbol{y}_i)$ and potentials $\psi_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ encoding the relationship between hidden and observable variables and pairs of hidden random variables, respectively.

$\boldsymbol{x}$ given some noisy observations $\boldsymbol{y}$. In order to simplify the following expositions, the nodes $\mathcal{V} = \{\boldsymbol{x}, \boldsymbol{y}\}$ are partitioned into two disjoint sets, where the set $\boldsymbol{x}$ corresponds to hidden variables $\boldsymbol{x} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$ and the set $\boldsymbol{y}$ to observable variables $\boldsymbol{y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_M\}$. Similarly, the potential functions are partitioned in a set corresponding to edges connecting two hidden variables $\psi_{ij}\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$ and a second set representing local observation a.k.a. observation potentials $\phi_i\left(\boldsymbol{x}_i, \boldsymbol{y}\right)$. In this case, the desired posterior distribution can be written as:

$$p\left(\boldsymbol{x}|\boldsymbol{y}\right) = \frac{p\left(\boldsymbol{y}|\boldsymbol{x}\right) p\left(\boldsymbol{x}\right)}{p\left(\boldsymbol{y}\right)} \propto \prod_{(i,j)\in\mathcal{E}} \psi_{ij}\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) \prod_{i\in\mathcal{V}} \phi_i\left(\boldsymbol{x}_i, \boldsymbol{y}\right). \tag{3.17}$$

In some cases, it might be possible to decompose the observations $\boldsymbol{y}$ into noisy local observation corresponding to single nodes $i, i \in \boldsymbol{y}$, so that the observation potential can be written as $\phi_i\left(\boldsymbol{x}_i, \boldsymbol{y}\right) = \phi_i\left(\boldsymbol{x}_i, \boldsymbol{y}_i\right)$. Figure 3.2 shows an example of a pairwise MRF, where shaded nodes represent local observations $\boldsymbol{y}_i$.

3.2.4 Directed Graphical Models

Directed graphical models or Baysian Networks (BNs) (see Figure 3.1a) are useful to express causal relationships between random variables. Formally, BNs are based on a directed acyclic graph $G = (\mathcal{V}, \mathcal{E})$. Each node $i \in \mathcal{V}$ in a BN represents a random variable $\boldsymbol{x}_i$. The directed edges in a BN are pointing from a parent node to its children representing the conditional dependencies of the child nodes given

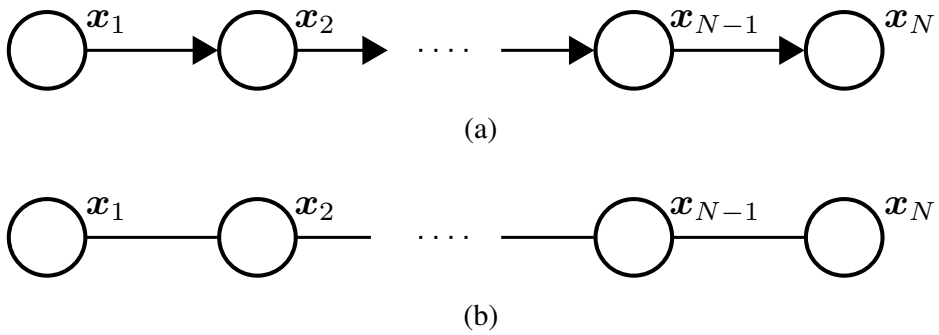


Figure 3.3: Directed (a) and undirected (b) chain graph over the same joint probability distribution $p(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{N-1}, \boldsymbol{x}_N)$. For both cases the factorization is given by $p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) = \prod_{i=1}^{N-1} \psi(\boldsymbol{x}_i, \boldsymbol{x}_{i+1})$.

its parents. The joint probability distribution of a BN factorizes as

$$p\left(\boldsymbol{x}\right) = \prod_{i \in \mathcal{V}} p\left(\boldsymbol{x}_i | \boldsymbol{x}_{\Gamma(i)}\right), \tag{3.18}$$

where $p\left(\boldsymbol{x}_i | \boldsymbol{x}_{\Gamma(i)}\right)$ is the conditional probability distribution of node $\boldsymbol{x}_i$ given its parents $\boldsymbol{x}_{\Gamma(i)}$.

A simple example for a BN is given in Figure 3.3a, where the joint probability distribution for the directed chain graph is defined as a product of conditional probabilities

$$p(\boldsymbol{x}) = p(\boldsymbol{x}_1)p(\boldsymbol{x}_2|\boldsymbol{x}_1)p(\boldsymbol{x}_3|\boldsymbol{x}_2)\ldots p(\boldsymbol{x}_N|\boldsymbol{x}_{N-1}). \tag{3.19}$$

Further, the chain graph example can be used to explain the relationship between directed and undirected graphical models. In the case of the directed chain graph depicted in Figure 3.3a, the directed graph can easily be converted into an undirected graph. For the undirected chain graph depicted in Figure 3.3b, the maximal cliques are pairs of neighboring nodes, which leads to the following factorization of the joint probability distribution

$$p(\boldsymbol{x}) = \frac{1}{Z}\psi_{1,2}(\boldsymbol{x}_1, \boldsymbol{x}_2)\psi_{2,3}(\boldsymbol{x}_2, \boldsymbol{x}_3)\ldots\psi_{N-1,N}(\boldsymbol{x}_{N-1}, \boldsymbol{x}_N). \tag{3.20}$$

As can be seen, the conversion is done by identifying

$$\begin{aligned} Z &= 1 \\ \psi_{1,2}(\boldsymbol{x}_1, \boldsymbol{x}_2) &= p(\boldsymbol{x}_1)p(\boldsymbol{x}_2|\boldsymbol{x}_1) \\ \psi_{2,3}(\boldsymbol{x}_2, \boldsymbol{x}_3) &= p(\boldsymbol{x}_3|\boldsymbol{x}_2) \\ &\vdots \\ \psi_{N-1,N}(\boldsymbol{x}_{N-1}, \boldsymbol{x}_N) &= p(\boldsymbol{x}_N|\boldsymbol{x}_{N-1}), \end{aligned}$$

where the marginal $p(\boldsymbol{x}_1)$ is absorbed into the first clique potential. Another common example is illustrated in Figure 3.1, where the factor $p(\boldsymbol{x}_3|\boldsymbol{x}_1, \boldsymbol{x}_2)$ depends on the variables $\boldsymbol{x}_1, \boldsymbol{x}_2$ and $\boldsymbol{x}_3$. In order to express this factor using an undirected graph, an additional edges has to be added between the parents of $\boldsymbol{x}_3$ (see Figure 3.1b). This process is called moralization and allows to identify $\psi_{123}(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3) = p(\boldsymbol{x}_3|\boldsymbol{x}_1, \boldsymbol{x}_2)$. In general, any directed graph can be converted into an undirected graph by adding additional edges to the graph. However, this means that also some conditional independence properties have to be dropped. More details on the relationship between directed and undirected graphical models can be found in [Kol09, Bis06, Bar12a, Mur06].

3.2.5 Factor Graphs

A factor graph [Ksc01, Fre02] is an alternative graphical model that unifies directed and undirected graphical models, and is commonly used to design inference algorithms. Formally, a factor graph is an undirected bipartite graph $\mathcal{G} = (\mathcal{V}, \mathcal{F}, \mathcal{E})$, where round nodes $\mathcal{V}$ correspond to random variables and squared nodes to factors $\mathcal{F}$ defined on the edges $\mathcal{E}$ of the graph (see Figure 3.1c). If $x_f = \boldsymbol{x}_i, i \in f$ is the corresponding set of variables for each factor $f \in \mathcal{F}$ in a factor graph, then it defines the joint probability distribution $p(x)$ as

$$p(\boldsymbol{x}) \propto \prod_{f \in \mathcal{F}} \psi_f(\boldsymbol{x}_f), \tag{3.21}$$

where the factors $\psi_f(\boldsymbol{x}_f)$ are functions of the corresponding set of variables $\boldsymbol{x}_f$.

Generally, most directed and undirected graphical models can be represented using a factor graph. For example, MRFs can always be written as a factor graph by using one factor node per clique in the MRF. Let us now consider the example, depicted in Figure 3.1. The last section, explained how to convert the BN depicted in Figure 3.1a into the moral graph illustrated in Figure 3.1b. An example for a corresponding factor graph is depicted in Figure 3.1c, where the factors corresponding to the marginals $p(\boldsymbol{x}_1)$ and $p(\boldsymbol{x}_2)$ are kept explicit. Note that, they could also be absorbed into the factor $\psi_f(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3)$, which corresponds to the definition of the MRF given in the previous section.

3.3 Random Variables and Probability Distributions

This section, presents the different types of probability distributions commonly referred to in this thesis. A brief introduction to discrete and continuous probability distribution is given as well as to Monte Carlo methods and kernel density estimation. The former are parametric models and the latter nonparametric models.

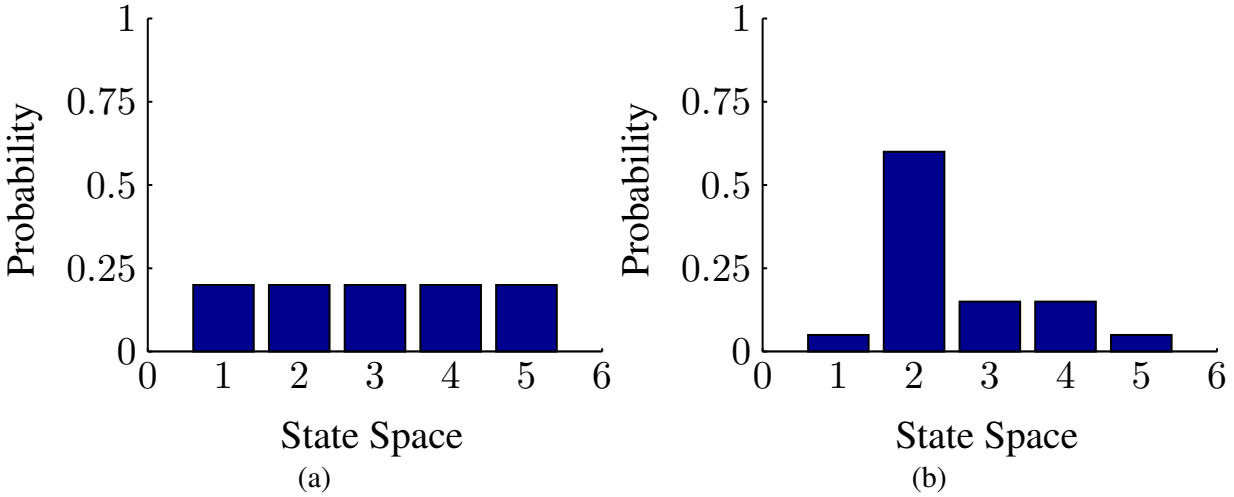


Figure 3.4: Two discrete distributions with discrete values $\mathrm{Val}(X) = \{1, 2, 3, 4, 5\}$. (a) Uniform distribution and (b) non-uniform distribution.

Parametric models have the advantage of commonly being fast to compute, but the disadvantage of making strong assumptions about the structure of the underlying data. Nonparametric models are more flexible, since their number of parameters grows with the size of the underlying dataset. The disadvantage of nonparametric models is that they are often computational intractable for large datasets.

3.3.1 Discrete Random Variables

Discrete random variables $\boldsymbol{x}$ are variables that can take one of K discrete, categorical values, so that $\mathcal{X} = \{1, \ldots, K\}$. Any probability mass function [2] or distribution $p(\boldsymbol{x})$ is then parameterized by the probability $\pi_k \triangleq \Pr[\boldsymbol{x} = k]$ of the K discrete outcomes. Here, it has to be satisfied that $\pi_k \in [0, 1]$ and $\sum_{k \in \mathcal{X}} \pi_k = 1$.

An example, for two different probability mass functions is given in Figure 3.4, where both distribution comprise the values $\mathrm{Val}(X) = \{1, 2, 3, 4, 5\}$.

3.3.2 Continuous Random Variables

The most commonly used probability distribution for continuous random variables is the multivariate Gaussian distribution. Formally, the probability density (PDF) function of a D dimensional multivariate Gaussian distribution is given by

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2} |\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right], \quad (3.22)$$

[2] A probability mass function consists of a finite number of Dirac delta distributions.

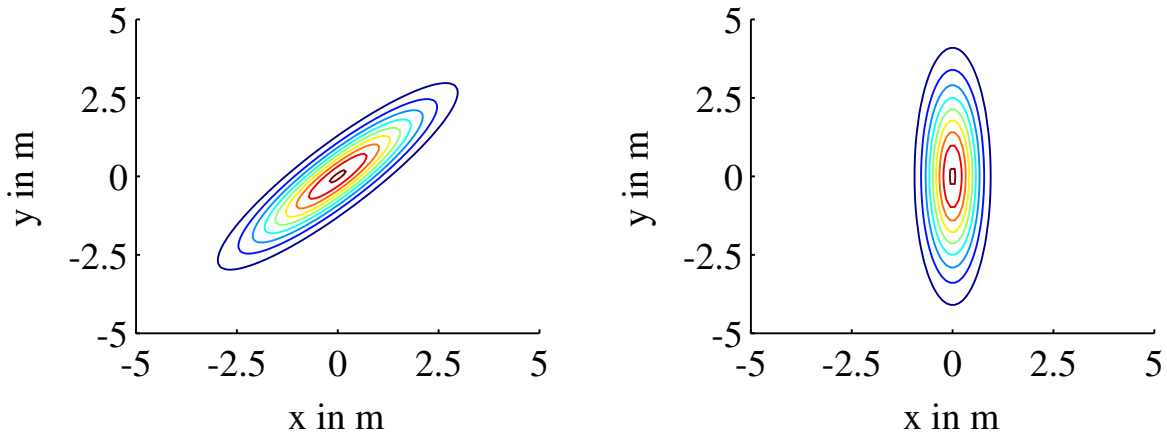


Figure 3.5: Two multivariate Gaussian distribution. (a) A full covariance matrix with elliptical contour. (b) A diagonal covariance matrix leading to an axis aligned ellipse.

where $\boldsymbol{\mu} \in \mathbb{R}^D$ is the mean vector and $\boldsymbol{\Sigma} \in \mathbb{R}^{D \times D}$ the covariance matrix. The normalization constant $(2\pi)^{D/2} |\boldsymbol{\Sigma}|^{1/2}$ ensures that the probability density function integrates to 1. An example, for different multivariate Gaussian distribution is depicted in Figure 3.5. One of the reasons why Gaussian distributions are popular for inference tasks is that a product of one or more Gaussian distributions is also a Gaussian distribution. The product of N Gaussian distributions $p\left(\boldsymbol{x}_i | \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\right), i \in \{1, \ldots, N\}$ is given as

$$p\left(\boldsymbol{y}\right) = \prod_{i=1}^{N} p\left(\boldsymbol{x}_i\right) = \mathcal{N}\left(\boldsymbol{y} | \boldsymbol{\mu}_{\boldsymbol{y}}, \boldsymbol{\Sigma}_{\boldsymbol{y}}\right), \tag{3.23}$$

where

$$\boldsymbol{\Sigma}_{\boldsymbol{y}}^{-1} = \sum_{i=1}^{N} \boldsymbol{\Sigma}_i^{-1} \quad \boldsymbol{\Sigma}_{\boldsymbol{y}}^{-1} \boldsymbol{\mu}_{\boldsymbol{y}} = \sum_{i=1}^{N} \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\mu}_i. \tag{3.24}$$

Another important property of Gaussian distributions is that the conditional distribution of two or more jointly Gaussian variables is also Gaussian [Ste03, Bis06].

3.3.3 Nonparametric Methods

In many practical applications, the distribution underlying a dataset is unknown. Hence, usually the model parameters can not specified and some kind of approxi-

mation has to be used. In this case, Monte Carlo methods [Mac98, Rob04], introduced in [Met49] provide a numeric approximation of the target density by using a weighted set of Dirac delta distribution or samples to approximation the target density $p(\boldsymbol{x}_i)$ as follows

$$p(\boldsymbol{x}_i) = \sum_{k=1}^{N} \pi_i^{(k)} \delta(\boldsymbol{x}_i - \boldsymbol{s}_i^{(k)}) \qquad \sum_{k=1}^{N} \pi_i^{(k)} = 1, \tag{3.25}$$

where $\{(\boldsymbol{s}_i^{(k)}, \pi_i^{(k)})\}_{k=1}^{N}$ is a set of weighted samples. Monte Carlo methods have the advantage that they efficiently represent complex distributions, but the disadvantage is that many samples are needed for a good approximation of the target density.

Some applications require a continuous and strictly positive density estimation. This gives rise to a nonparametric density estimation method known as Parzen window density estimator [Par62] or Kernel Density Estimator (KDE). KDE is a unsupervised density estimation method that smooths the raw samples set by assigning a smoothing kernel $\kappa(\cdot)$ to each sample $\boldsymbol{s}_i^{(k)}$. The approximation of the target density $p(\boldsymbol{x}_i)$ is then given by

$$p(\boldsymbol{x}_i) = \sum_{k=1}^{N} \pi_i^{(k)} \kappa(\boldsymbol{x}_i - \boldsymbol{s}_i^{(k)}). \tag{3.26}$$

A popular choice is a Gaussian kernel, since it guarantees a smooth and strictly positive density estimation. The Gaussian kernel is defined as

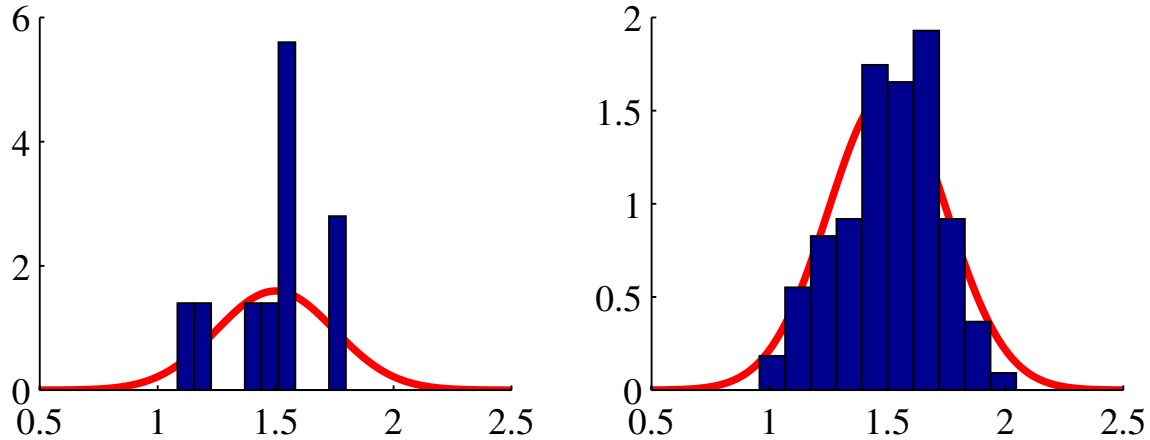
$$\kappa\left(\boldsymbol{x}\right) = \frac{1}{\left(\sqrt{2\pi}\boldsymbol{\Lambda}\right)^{D}} \exp\left(-\frac{|\boldsymbol{x}|^2}{2\boldsymbol{\Lambda}^2}\right), \tag{3.27}$$

where the bandwidth parameter h is used to control the width of the kernel and D is the dimensionality of the variable $\boldsymbol{x}$. Using the Gaussian kernel, the resulting density estimate is given by a Gaussian mixture model [Mur06, Bis06] as

$$p(\boldsymbol{x}_i) = \sum_{k=1}^{N} \pi_i^{(k)} \mathcal{N}(\boldsymbol{x}_i; \boldsymbol{s}_i^{(k)}, \boldsymbol{\Lambda}_i) \qquad \sum_{k=1}^{N} \pi_i^{(k)} = 1. \tag{3.28}$$

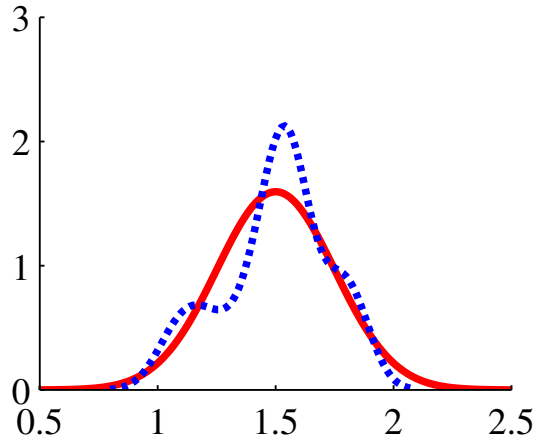
The bandwidth $\boldsymbol{\Lambda}_i$ is usually automatically selected using cross-validation or the computational efficient rule of thumb [Sil86].

Examples for nonparametric density estimation are given in Figure 3.6. As can be seen, the accuracy of Monte Carlo approximation increases with the number of

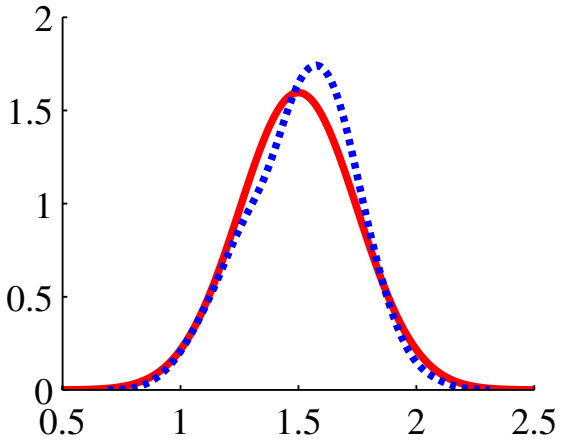


(a) 10 samples

(b) 100 samples



(c) 10 samples



(d) 100 samples

Figure 3.6: Results of nonparametric density estimations for 10 and 100 samples drawn from the true Gaussian distribution (red). (a) and (b) sample histograms. (c) and (d) KDE using a Gaussian kernel with automatic bandwidth selection according to the rule of thumb [Sil86].

samples. The top row shows a histogram[3] that is based on the samples drawn from the original Gaussian distribution. The bottom row depicts the smoothed results using a Gaussian kernel.

Note that, the mixture models of KDE requires significantly more storage space and evaluation time, then the raw sample set of the Monte Carlo approximation. Hence, in many practical application[4], the raw sample set is used.

3.4 Inference in Graphical Models

The previous sections, introduced graphical models as a compact way to represent complex joint probability distributions. The main use of this compact representation is to perform efficient inference. In general, inference refers to the task of estimating the posterior distribution $p(\boldsymbol{x}|\boldsymbol{y})$ of hidden random variables $\boldsymbol{x}$ given some noisy observations $\boldsymbol{y}$. In many practical applications, computing the full joint distribution is computational expensive and the posterior marginal distribution

$$p(\boldsymbol{x}_i|\boldsymbol{y}) = \sum_{\mathcal{X}_{\mathcal{V}\setminus i}} p(\boldsymbol{x}|\boldsymbol{y}) \quad \text{or} \quad p(\boldsymbol{x}_i|\boldsymbol{y}) = \int_{\mathcal{X}_{\mathcal{V}\setminus i}} p(\boldsymbol{x}|\boldsymbol{y}) d\boldsymbol{x}_{\mathcal{X}_{\mathcal{V}\setminus i}} \tag{3.29}$$

are used as an efficient summary. Here $\mathcal{V} \setminus i$ denotes the set of all nodes $\mathcal{X}$ in the graph except i. The focus of the following sections are exact and approximate inference algorithms for preforming efficient marginal inference in tree-structured graphical models.

3.4.1 Discrete Belief Propagation

We start by introducing belief propagation (BP) [Pea88, Yed01, Yed02] for a MRF encoded by a Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where all random variables are discrete. Using BP, the posterior marginal distributions or belief is computed as

$$b_i(\boldsymbol{x}_i) = p(\boldsymbol{x}_i|\boldsymbol{y}) = \frac{1}{Z}\phi_i(\boldsymbol{x}_i, \boldsymbol{y}_i) \prod_{j \in \Upsilon(i)} m_{ji}(\boldsymbol{x}_i), \tag{3.30}$$

where the real number Z denotes the partition function that normalizes the probability distribution. This belief update is performed by combining the local observation potential $\phi_i(\boldsymbol{x}_i, \boldsymbol{y}_i)$ with all incoming messages $m_{ji}(\boldsymbol{x}_i)$. Messages $m_{ji}(\boldsymbol{x}_i)$

[3] A histogram [Sil86, Bis06] is the most fundamental nonparametric representation of a density function.

[4] One very popular application using the raw sample set is the standard particle filter [Dou01, Isa98a]. A smoothed sample set is e.g., used for the regularized particle filter [Aru02, Mus01].

send from hidden node j to hidden node i can intuitively be understood as a summarized belief about the state of node i in respect to the evidence on node j. The message update is defined as

$$m_{ji}(\boldsymbol{x}_i) \propto \sum_{\boldsymbol{x}j} \psi_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)\phi_j(\boldsymbol{x}_j, \boldsymbol{y}_j) \prod_{k \in \Upsilon(j)\setminus i} m_{kj}(\boldsymbol{x}_j). \tag{3.31}$$

Generally, the message update can be performed in two stages. First, the partial belief

$$b_{j\setminus i}(\boldsymbol{x}_j) = \phi_j(\boldsymbol{x}_j, \boldsymbol{y}_j) \prod_{k \in \Upsilon(j)\setminus i} m_{kj}(\boldsymbol{x}_j), \tag{3.32}$$

is calculated which summarizes the belief about node j, except the message send from node i to j. Then, the partial belief over node j is converted into the belief over node i using the corresponding clique potential $\phi_i(\boldsymbol{x}_i, \boldsymbol{y}_i)$. As mentioned before, a brute force inference scheme, which simply enumerates all possible states of $\boldsymbol{x}$ and evaluates $p(\boldsymbol{x})$ requires $\mathcal{O}(L^N)$ computation time, and is therefore impracticable. The BP algorithm presented in this section can calculate all marginals for a N node tree in $\mathcal{O}(NL^2)$ time, which is a significant improvement.

Furthermore, it has been shown in [Pea88] that the BP algorithm only converges to the correct marginal posterior distribution in tree-like graphical models. Thus, Equation 3.30 only holds for tree-like graphical models.

In analogy to Equation 3.31, BP is sometimes also called sum-product algorithm [Ksc01, Bis06]. If discrete BP is restricted to Hidden Markov Models, it is equivalent to the well known forward-backward algorithm [Rab89].

3.4.2 Continuous Belief Propagation

So far, it was assumed that, all random variables in the graphical model are discrete. However, this thesis also deals with continuous variables. In this case the message update equation 3.31 of the BP algorithm has to be rewritten as

$$m_{ji}(\boldsymbol{x}_i) = \int_{\boldsymbol{x}_j} \psi_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)\phi_j(\boldsymbol{x}_j, \boldsymbol{y}_j) \prod_{k \in \Upsilon(j)\setminus i} m_{kj}\left(\boldsymbol{x}_j\right) d\boldsymbol{x}_j, \tag{3.33}$$

where the sum over all states of $\boldsymbol{x}_j$ is replaced by an integral. Note that both Equation 3.17 and Equation 3.30 hold for discrete as well as continuous variables. In the cases, where the potentials $\phi_j(\boldsymbol{x}_j, \boldsymbol{y}_j)$ and $\psi_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ are both Gaussian also the posterior marginal distribution at each node is Gaussian and the integral in Equation 3.33 can be calculated exactly [Wei01].

3.4.3 Nonparametric Belief Propagation

So far, BP was introduced for the restricted classes of graphical models with discrete variables or continuous Gaussian variables. However, in many application, as in this thesis, the underling distribution are non-Gaussian due to ambiguities and thus multimodal sensory evidence. This is what gave rise to the development of nonparametric belief propagation (NBP) [Isa03, Sud03], which is a generalization of sequential Monte Carlo methods [Isa98a, Dou01] to arbitrary graphs.

Nonparametric Messages and Beliefs

The key observation underling NBP is that in the non-Gaussian case the integration required to perform the message update in Equation 3.33 can not be solved exactly. Therefore, instead of performing inference exactly, NBP uses Monte Carlo techniques and KDE to approximate both beliefs and messages.

Using Monte Carlo techniques, messages could be approximated using a set of discrete samples $\{(\boldsymbol{s}_i^{(k)}, \pi_i^{(k)})\}_{k=1}^{L}$. Due to the independence assumptions encoded by the graph, samples for different messages from independent proposal functions have to be generated. If these samples are defined on a continuous sample space this means that they are distinct with the probability one. Consequently, the product of Equation 3.30 is guaranteed to be zero. In order to ensure that the message product is non-degenerative Gaussian kernels (see Section 3.3.3) are used to construct strictly positive messages

$$m_{ji}(\boldsymbol{x}_i) = \sum_{k=1}^{L} \pi_{ji}^{(k)} \mathcal{N}(\boldsymbol{x}_i; \boldsymbol{s}_{ji}^{(k)}, \boldsymbol{\Lambda}_{ji}). \tag{3.34}$$

Similarly, the belief is approximated as

$$b_i(\boldsymbol{x}_i) = \sum_{k=1}^{L} \pi_i^{(k)} \mathcal{N}(\boldsymbol{x}_i; \boldsymbol{s}_i^{(k)}, \boldsymbol{\Lambda}_i). \tag{3.35}$$

Belief Update and Message Propagation

Let us now consider the belief update of Equation 3.30, where all incoming messages are represented nonparametrically as in Equation 3.34. If further the observation potential is represented as a Gaussian mixture model, the belief update requires to calculate the product of $d = |\Upsilon(j)| + 1$ Gaussian mixtures. If each of these Gaussian mixtures has L components, the resulting product has L^d components. Hence, the brute force approach to perform inference would require $\mathcal{O}(L^d)$

time, which is time exponential in the number of mixtures. Even for a small number of mixtures with few components, this is intractable. In order to keep inference tracable, the NBP algorithm approximates the mixture product by a set of L independent samples

$$\boldsymbol{s}_i^{(k)} \sim \phi_i(\boldsymbol{x}_i, \boldsymbol{y}_i) \prod_{j \in \Upsilon(i)} m_{ji}(\boldsymbol{x}_i) \qquad k = 1, \ldots, L. \tag{3.36}$$

Given these samples, KDE methods can be employed to reconstruct the belief as in Equation 3.35.

The NBP algorithms performs the message update of Equation 3.33 in two stages. First, it draws L independent samples from the partial belief estimate

$$\boldsymbol{s}_j^{(k)} \sim \frac{1}{Z_j} \varphi_{ij}(\boldsymbol{x}_j) b_{j \setminus i}(\boldsymbol{x}_j) \qquad k = 1, \ldots, L, \tag{3.37}$$

where $b_{j \setminus i}(\boldsymbol{x}_j)$ is the partial belief of node j (see Equation 3.32) and $\varphi_{ij}(\boldsymbol{x}_j) = \int_{\mathcal{X}_i} \psi_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) d\boldsymbol{x}_i$ is the marginal influence, which captures the influence of $\psi_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ on $\boldsymbol{x}_j$ and is used to reweight the samples. Then the auxiliary samples are propagated to node i using the associated pairwise clique potential

$$\boldsymbol{s}_{ji}^{(k)} \sim \frac{1}{Z_i^{(k)}} \psi_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j = \boldsymbol{s}_j^{(k)}) \quad Z_i^{(k)} = \int_{\boldsymbol{x}_i} \psi_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j = \boldsymbol{s}_j^{(k)}) d\boldsymbol{x}_i, \tag{3.38}$$

where the double subscript implies that the samples $\boldsymbol{s}_{ji}^{(k)}$ are distributed according to $m_{ji}(\boldsymbol{x}_i)$. As in the case of the belief update, the resulting sample set and KDE methods can be used to reconstruct the messages as in Equation 3.34.

It is important to note that in many application, as in this thesis, pairwise potentials only depend on the difference in their arguments, so that $\psi_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_i) = \tilde{\psi}_{ij}(\boldsymbol{x}_i - \boldsymbol{x}_j)$. In this case, the marginal influence is constant and can be neglected [Sud06, Sig08, Fel05].

Sampling from Products of Gaussian Mixtures

Mixture products are the mechanisms by which NBP updates both beliefs and messages. As mentioned before, exact calculation of these mixture products is often intractable. In order to overcome this issue, several solutions have been proposed. For example, in [Sud03] good results were archived using a Gibbs sampler that performs product sampling in K iterations and requires $\mathcal{O}(KdL^2)$ time. Based on the assumption that the mixtures have diagonal covariance structure, Ihler et al. [Ihl03] proposed an approximate sampling scheme which perform the product sampling in $\mathcal{O}(KdL)$ time.

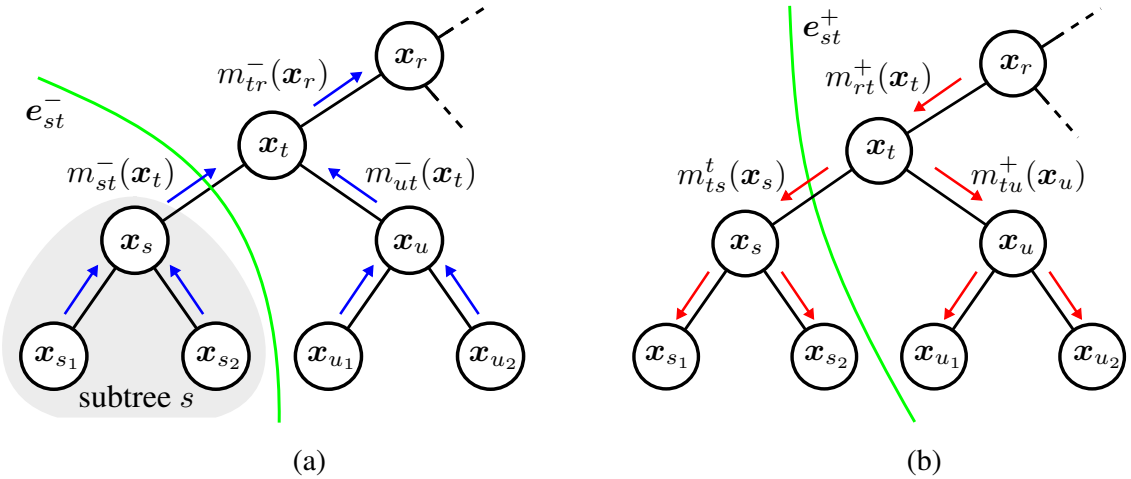


Figure 3.7: Message passing on a tree-structured graph. (a) Bottom up phase. (b) Top down phase.

Sampling schemes dedicated to real-time applications have been proposed in [Spe13], where nearest neighbor search is used to reduce the computational coast of product sampling, resulting in a computational complexity of $\mathcal{O}(dL)$. This low computational complexity is archived by not multiplying each Gaussian with each Gaussian of the other mixture, but instead multiplying each Gaussian with the most probable Gaussian. The weights π_r of the product result can then simply be computed as

$$\pi_r = \frac{\prod_{i=1}^{i=2} \pi_i \mathcal{N}(x; \mu_i, \boldsymbol{\Lambda}_i)}{\mathcal{N}(x; \mu_r, \boldsymbol{\Lambda}_r)}. \tag{3.39}$$

3.4.4 Message Passing in Hierarchical Models

So far, BP and nonparametric BP were introduced for a simple example, where a outgoing messages was propagated as soon as all inputs were available. Applied to tree-structured graphs, this standard message passing schedule results in propagating messages from the leaves upwards to the root and back down. In the following, this message passing schedule is explained using the tree-structured graph depicted in Figure 3.7, which has the same structure as the graphical models used throughout this thesis. For illustrative reasons, let us assume that inference can be solved using continuous BP as introduced in Section 3.4.2. Note, the presented message passing schedule can be applied analogously for NBP.

In order to implement a message passing schedule on a undirected graph, a definition for the root is needed, which is defined by selecting an arbitrary node. Then, all edges are oriented away from the root, which gives a definition of parent

and child. An example is depicted in Figure 3.7 where node r is defined as the root. Accordingly, it can now be stated that e.g., node t is the parent of node s.

Let us now consider the task of estimating the belief of node t. In general, this calculation can be performed in two phases. First, during the bottom up phase, the belief of node t conditioned only on its local observation potential and evidence that is below node t in the graph is calculated. This bottom up belief is denoted as $b^-(\boldsymbol{x}_t) = p(\boldsymbol{x}_t|\boldsymbol{e}_t^-)$, where $\boldsymbol{e}_t^-$ is the summary of the local evidence at node t and evidence form below node t. Second, in the top down phase, the belief of node t is obtained by combining the bottom up belief with the evidence provided by the parents of node t. In analogy to Equation 3.30 the bottom up belief $b^-(\boldsymbol{x}_t)$ of node t can be written as

$$b^-(\boldsymbol{x}_t) \propto p(\boldsymbol{x}_t|\boldsymbol{e}_t^-) \propto \phi_t(\boldsymbol{x}_t, \boldsymbol{y}_t) \prod_{c \in \Xi(t)} m_{ct}^-(\boldsymbol{x}_t), \tag{3.40}$$

where $m_{ct}^-(\boldsymbol{x}_t)$ refers to the incoming messages from nodes s and t. Now e.g., $m_{st}^-(\boldsymbol{x}_t)$, can be computed according to Equation 3.33 as

$$m_{st}^-(\boldsymbol{x}_t) \propto \int_{\boldsymbol{x}_s} \psi_{st}(\boldsymbol{x}_s, \boldsymbol{x}_t)\phi_s(\boldsymbol{x}_s, \boldsymbol{y}_s) \prod_{c \in \Xi(s)} m_{cs}^-(\boldsymbol{x}_s) d\boldsymbol{x}_s. \tag{3.41}$$

Since the above message update is computed recursively, the messages $m_{st}^-(\boldsymbol{x}_t)$ can be understood as a summary of what node t should know about its subtree s. Formally, this is denoted as $m_{st}^-(\boldsymbol{x}_t) \coloneqq p(x_t|\boldsymbol{e}_{st}^-)$, where $\boldsymbol{e}_{st}^-$ summarizes all the evidence in the subtree s (see Figure 3.7a). This recursive procedure continues until all messages reached to the root. After the belief of the root is estimated, the bottom up phase terminates and the top down phase starts.

Let us now consider the task of calculating the belief state of node s. During the bottom up phase the bottom up belief was already computed by combining the messages from the children of node s with the local observation potential. Hence, only the bottom up belief of node s has to be combined with the top down messages of its parents

$$b(\boldsymbol{x}_s) \propto b^-(\boldsymbol{x}_s) \prod_{t \in \Gamma(s)} m_{ts}^+(\boldsymbol{x}_s). \tag{3.42}$$

In the above example, this means that the message send form node t to node s has

to be computed, which is defined as

$$m_{ts}^{+}(\boldsymbol{x}_s) \coloneqq p(\boldsymbol{x}_s|\boldsymbol{e}_{st}^{+}) \propto \int_{\boldsymbol{x}_t} \psi_{st}(\boldsymbol{x}_s, \boldsymbol{x}_t)\phi_t(\boldsymbol{x}_t, \boldsymbol{y}_t) \prod_{c\in\Xi(t)\setminus s} m_{ct}^{-}(\boldsymbol{x}_t) \prod_{p\in\Gamma(t)} m_{pt}^{+}(\boldsymbol{x}_t)d\boldsymbol{x}_t, \tag{3.43}$$

where $\boldsymbol{e}_{st}^{+}$ is the summary of all the evidence that is not part of the subtree s (see Figure 3.7b). Note, Equation 3.43 is equivalent to Equation 3.33. However, for illustrative reasons, the message origin is kept explicit. The top down phase ends when all leaves received the messages.

The message passing algorithm presented in this section is a synchronous message passing schedule, since it amounts in processing each level of the graphical model synchronously. An alternative, is to apply an asynchronous message passing schedule [Eli06, Wai08] which propagate messages in a serial order defined by a fixed schedule. The key challenge in asynchronous message passing is to guarantee that each node receives all its input messages. Empirical results show that often asynchronous message passing schedules converge faster then the naive synchronous approach [Eli06]. Consequently, this thesis, mainly uses a fixed asynchronous message passing schedule.

4 Compositional Hierarchical Models of Multi-Lane Roads

This chapter presents a novel compositional hierarchical framework [Töp13] which represents multi-lane roads as a composition of parts to provide a unified probabilistic framework for rapid multi-lane road perception. The following sections are structured as follows. Section 4.1 details the lane and road cues employed for the addressed perception tasks, and presents how they are incorporated into the proposed framework. The Compositional Hierarchical Model (CHM) of multi-lane roads is introduced in Section 4.2, alongside with details on how dependencies between low-level cues and high-level road topologies are encoded. Finally, Section 4.3 introduces the applied inference algorithm, including the notion of using part-sharing [Zhu10, Spe11] and depth-first message passing [Töp13] for computational efficiency.

In the proposed approach, road and lane perception is formulated as the problem of inference in a probabilistic graphical model. More precisely, multi-lane roads are represented in a CHM [Spe11, Tor10, Mur03] which is encoded by a pairwise Markov random field (see Section 3.2.3). In this CHM the root corresponds to a full model of a multi-lane road with all its intricacies (e.g., number and position of lanes and parallel, splitting and merging lanes), and the nodes on the lower-levels of the CHM to a recursive decomposition of the root object into parts and subparts, as shown in Figure 4.1. This figure shows that this decomposition leads to a layered object representation with decreasing part complexity in direction of the leaves.

This compositional hierarchical representation has several convenient properties. First, the decomposition simplifies the perception of complex multi-lane roads, because the perception problem is divided into sub-problems of lower complexity. In addition, this decomposition explicitly encodes the dependencies between observable low-level cues and complex road topologies. Second, each node in a CHM makes inference over a finite local area which ensures that e.g., clutter partial occlusions, and local illumination changes have a limited effect on the overall perception process [Sig08, Fel05, Isa03, Xie03, Moh01, Fis73]. Third, edges between nodes in a CHM encode probabilistic dependencies. These ensure not only the compatibility of parts by introducing weak spatial constraints, but also incorporate a priori scene knowledge, such as expected lane widths or assumptions on lane topologies. During the perception, this prior knowledge allows for both re-

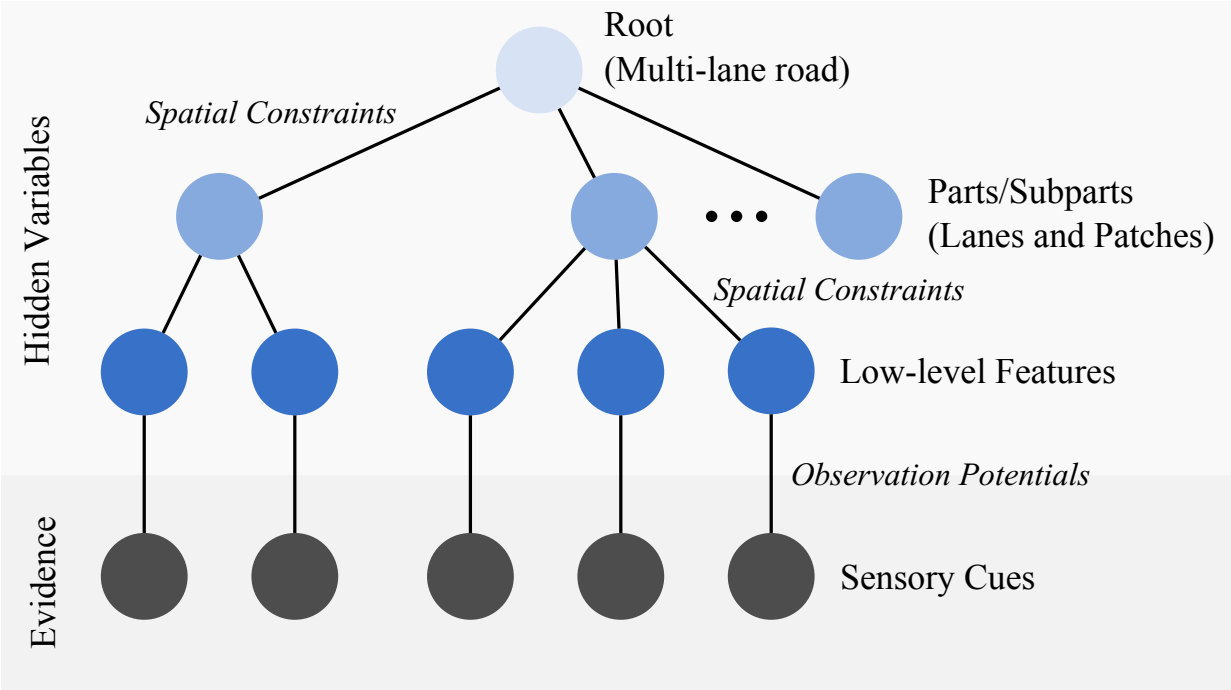


Figure 4.1: Simplified structure of the proposed CHM of multi-lane roads. Components, corresponding to the nodes of the model and represent multi-lane roads on different levels of abstraction. Edges between pairs of parts and parts and feature detections, encode weak spatial constraints and observations potentials, respectively.

ducing the search space to regions in which object-parts are likely to occur and to reject patterns that seem to be valid, but are located in unlikely places. Therefore, incorporating a priori scene knowledge is not only computational attractive, but improves the overall performance of the proposed approach. Fourth, most importantly, given a small number of features CHMs have the property of being able to construct a large number of combinations which growth exponentially with each level of the hierarchy. In fact, this expressive power is the key in applying the proposed framework to varying road topologies, as proposed in Chapter 5.

Given the hierarchical graphical model lane and road perception can efficiently be performed using Belief Propagation (BP) (see Section 3.4.3). However, since the state space represented by the proposed model is continuous and multi-dimensional exact inference is computational intractable. Therefore, a version of Nonparametric Belief Propagation (NBP) (see Section 3.4.3) is exploited that performs efficient approximate inference. Most importantly, in NBP messages and beliefs are represented nonparametrically which allows to account for the multi-modal, non-Gaussian sensory evidence.

In summary, the proposed hierarchical framework can be viewed as having four distinct components: (1) local evidence distributions, (2) a hierarchical graphical

model, (3) a set of weak spatial constraints encoded by the edges of the graph, and (4) an inference algorithm that provides the ability to infer the belief state of each node in the graph. In the following, each one of these is discussed in turn.

4.1 Sensor Evidence

In this thesis, lane and road perception exclusively relies on low-level evidence, obtained from the visual input during feature extraction (see Figure 2.1). Hence, all methods presented in this thesis crucially depend on the reliability of the feature extraction results. This section, motivates the used perceptual lane and road cues, and the employed feature extraction approaches. Further, it specifies how features are incorporated into the CHM, including the representation of features and the formulation of observation potentials which encode the dependencies between hidden and observable variables in the CHM.

4.1.1 Cues for Lane and Road Detection

Like most structured environments roads have certain characteristics that distinguish them from the background clutter in the scene. In fact, there exists a vast amount of cues that can be used to support lane and road perception (see Section 2.1). Generally, the following lane and road cues can be identified [Apo04, Vei08]:

- While the texture may vary between a road and its environment, it is often homogeneous within a single road.
- Similarly, the color can be assumed to be constant within a single road. However, shadows may change its brightness.
- The boundary separating the road from non-road areas if often characterized by a road edge (e.g., curbstone or guard rail).
- The most common lane cue on structured roads is the lane marking which may differ in appearance, i.e., dashed, continuous, arrows and stoplines.

Each of the above cues has been used in recent lane and road perception approaches (see Section 2.1) and has both advantages and disadvantages over the other cues. Most importantly, however, no single cue is expected to be reliable in all scenarios. Therefore, a combination of multiple cues is used to achieve both the reliability and flexibility that is required for commercial ADAS applications.

The cues selected for the presented approach are designed to be simple and efficient, while covering a variety of scenarios. Further, the features should be robust

against clutter and local illumination changes. These considerations motivate the usage of lane marking and road edge cues. Lane markings are the most common cue on structured roads and if present a strong cue for lane and road perception. Most importantly, the various types of lane marking cues (e.g., continuous lane markings, dashed lane markings, stoplines and turn arrows) encode important semantic road properties. Road edge cues are used as a complimentary input source to improve the performance in less structured urban or rural scenarios, where lane markings are not reliable. Further, both of these cues have the advantages that there exists a variety of well developed and efficient feature extraction approaches.

More formally, given the visual input, two different feature detection approaches are employed. First, lane marking features $\boldsymbol{m} = \{\boldsymbol{m}_1, \ldots, \boldsymbol{m}_{N_m}\}$ are extracted from the visual input using the symmetrical local threshold method. This method is used since according to the evaluation conducted in [Vei08] it yields the best results in the general case. In particular, it has been shown that this method performs well compared to other state-of-the-art approaches in scenarios with variable lighting conditions (e.g., shadow, bright sun and cloudy weather), variable scene content (e.g., dense urban to countryside) and variable road types (e.g., highways, urban roads and country roads). Furthermore, an edge detector is used to gain road edge features $\boldsymbol{r} = \{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{N_r}\}$ in image regions where markings are missing, but road edges such as curbs are present. A Sobel detector at multiple scales is employed for a scale invariant edge detection. Exemplary results of applying these two feature extraction approaches to a single input image are shown in Figure 4.2.

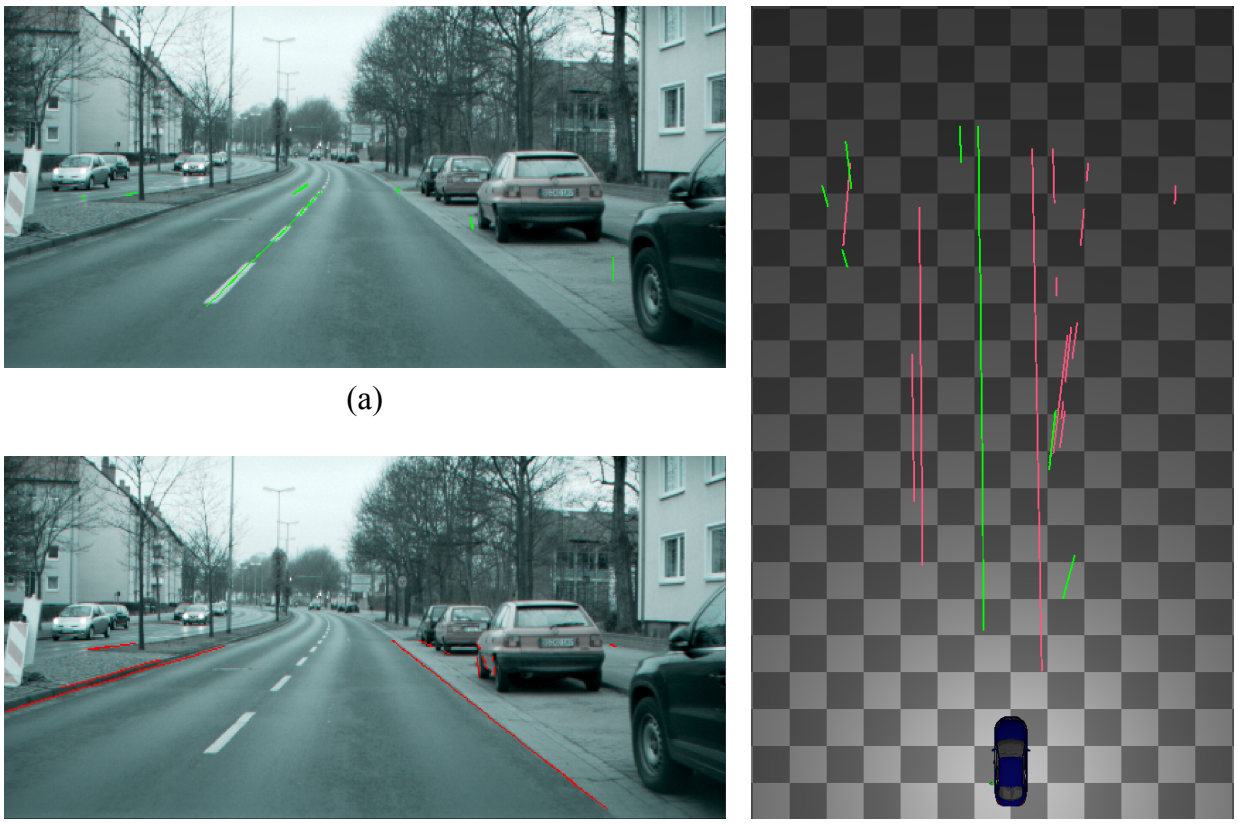
The approach presented in this thesis, mainly relies on these two feature extraction approaches. However, in the following chapters a subsequent classification step is used that estimates the class of lane marking features to estimate semantic road properties (e.g., turn direction of lanes) (see Chapter 6).

4.1.2 Feature Representation and Observation Potentials

Given the visual input, low-level features are obtained by applying the two feature extraction approaches detailed above. Subsequently, the feature extraction results are transformed from the image domain into the vehicle coordinate system [Mal91][1], where a flat road surface is assumed. The vehicles coordinate system is located at the center of the rear axis on the road surface, and follows the common axis definition ($x =$ forward, $y =$ left, $\vartheta =$ yaw angle).

Formally, the two feature extraction approaches obtain a set of lane marking features $\boldsymbol{m} = \{\boldsymbol{m}_1, \ldots, \boldsymbol{m}_{N_m}\}$ and a set of road edge features $\boldsymbol{r} = \{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{N_r}\}$. A lane marking feature $\boldsymbol{m}_i = (x_i, y_i, \vartheta_i)$ is defined by its location $(x_i, y_i) \in \mathbb{R}^2$

[1] We choose to perform lane and road perception in the vehicle coordinate system or bird's eye view, since most vehicle control task are performed in the 2D plane of the vehicle (see Section 2.3).



(a) (b) (c)

Figure 4.2: Low-level feature input for lane and road perception. (a) Lane marking features (green). (b) Road edge features (red). (c) Projection into the vehicle coordinate system.

and orientation $\vartheta_i \in [0, 2\pi)$. Similarly, a road edge feature is defined as $\boldsymbol{r}_i = (x_i, y_i, \vartheta_i)$. These features constitute the observable random variables $\boldsymbol{r}$ and $\boldsymbol{m}$ of the CHM which have corresponding hidden random variables $\boldsymbol{x}_i^f$ that are defined on the same three dimensional state space that constitute the first level $\mathcal{L}_1$ of the CHM (see Figure 4.4).

Definition 1 (Feature)**.** *A feature variable* $\boldsymbol{x}_i^f = (x_i, y_i, \vartheta_i)$ *is specified by its position* $(x_i, y_i) \in \mathbb{R}^2$ *and orientation* $\vartheta_i \in [0, 2\pi)$ *in the vehicles coordinate system.*

As in Section 3.2.3, the dependencies between hidden feature variables $\boldsymbol{x}_i^f$ and the observations $\boldsymbol{r}$ and $\boldsymbol{m}$ are encoded using observation potential $\phi_i(\boldsymbol{x}_i^f, \boldsymbol{r})$ and $\phi_i(\boldsymbol{x}_i^f, \boldsymbol{m})$, respectively. These, observation potentials model the probability of making an observation $\boldsymbol{m}$ or $\boldsymbol{r}$ conditioned on the state of the associated variable $\boldsymbol{x}_i^f$. In the proposed model, the probability of observing a lane marking given the

state of the feature variable $\boldsymbol{x}_i^f$ is modeled by the observation potential

$$\phi_i(\boldsymbol{x}_i^f, \boldsymbol{m}) = \epsilon^0 \mathcal{N}_0(\boldsymbol{x}_i^f; 0, \boldsymbol{\Sigma}_0) + (1-\epsilon^0) \sum_{k=1}^{N_m} \pi_{i,k} \mathcal{N}(\boldsymbol{x}_i^f; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_{i,k}) \tag{4.1}$$

and analogously for observation potentials $\phi_i(\boldsymbol{x}_i^f, \boldsymbol{r})$ accounting for road edge features. Here, $\boldsymbol{\Sigma}_{i,k} \in \mathbb{R}^{3\times3}$ is the covariance matrix of the k^{th} mixture component. Formally, this observation potential is a kernel density estimation of the true likelihood which is obtained by assigning a Gaussian kernel to each feature $\boldsymbol{\mu}_k$. $\pi_{i,k}$ denotes the probability of the association of the k-th feature with the i-th hidden variable. Further, the observation potential is augmented by a zero mean, high-variance Gaussian outlier process $\mathcal{N}_0(\boldsymbol{x}_i; 0, \boldsymbol{\Sigma}_0)$ that is adjusted to represent 20% ($\epsilon^0 = 0.2$) of the total likelihood [Isa98a, Sud03, Sig04, Thr05]. The outlier process is essential in the presented approach, as it ensures that the message product is non-degenerative in the presence of partial occlusions (see Section 4.3.1).

An illustrative example of the feature extraction results and the proposed KDE-based approximation of the true observation likelihood is depicted in Figure 4.3. This figure shows the approximation of the true likelihood using the proposed observation potential for a continuous and a dashed lane marking.

4.2 Compositional Hierarchical Model

In the proposed framework, a multi-lane road is represented in a CHM which is encoded by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with nodes $\mathcal{V}$ and edges $\mathcal{E}$. The nodes $\mathcal{V}$ correspond to three disjoint sets of variables $\mathcal{V} = \boldsymbol{x} \cup \boldsymbol{m} \cup \boldsymbol{r}$, where $\boldsymbol{x}$ denotes the set of hidden random variables $\boldsymbol{x} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$. Each hidden variable $\boldsymbol{x}_i \subseteq \boldsymbol{x}$ represents a part or a sub-part of the multi-lane road which is represented by the root node of the graphical model (see Figure 4.1) and is defined on a multidimensional continuous state space. Further, the observable variables $\boldsymbol{m}$ and $\boldsymbol{r}$ correspond to the lane marking features and the road edge features, respectively.

The edges $\mathcal{E}$ between pairs of hidden variables define spatial constraints $\psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ which encode the dependencies between two neighboring hidden variables $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Intuitively, spatial constraints can be thought of as the probability of configuration $\boldsymbol{x}_i$ conditioned on the spatial probability distribution over $\boldsymbol{x}_j$. A key aspect in the design of spatial constraints is that multiple samples have to be drawn from the potentials during message passing (see Section 3.4.3). Therefore, the formulation of potentials has a large influence on the computational complexity and should be as simple as possible, while sufficiently expressing spatial

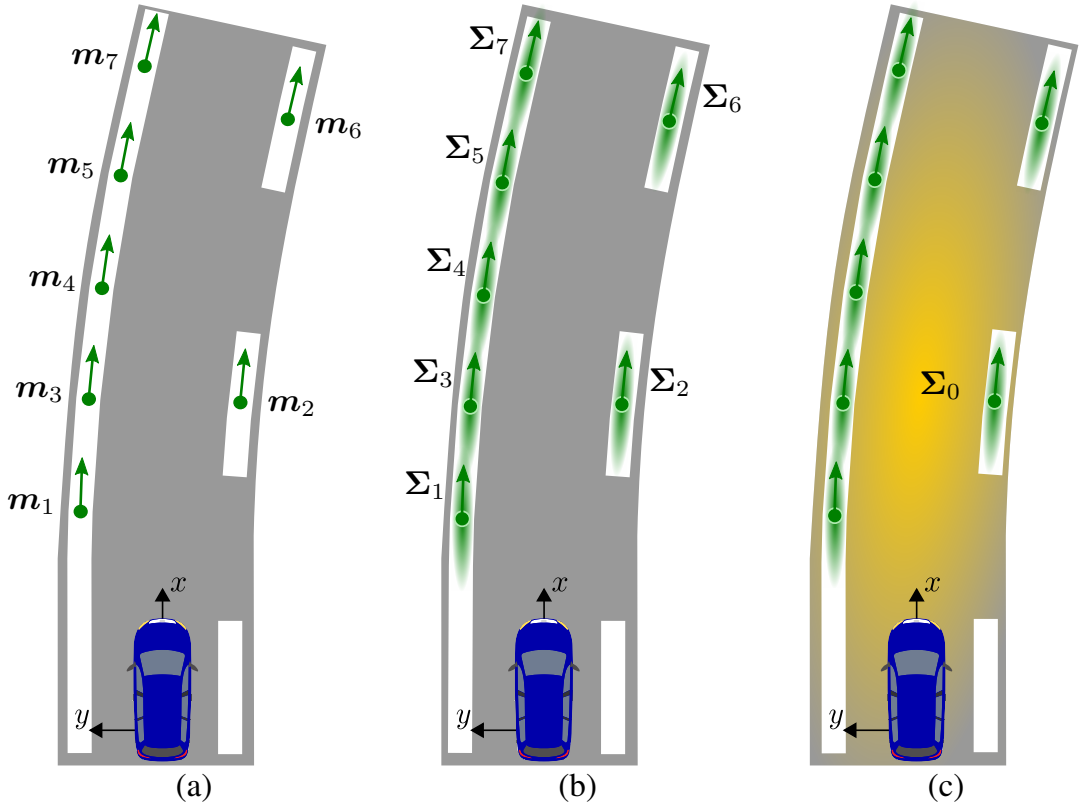


Figure 4.3: Observation potential. (a) Feature extraction results. (b) Approximation of the true likelihood without outlier process (green). (c) Approximation of the true likelihood with outlier process (yellow).

uncertainties. Furthermore, edges between hidden and observable random variables encode observation potentials $\phi_i(\boldsymbol{x}_i, \boldsymbol{m})$ and $\phi_i(\boldsymbol{x}_i, \boldsymbol{r})$, as defined in Equation 4.1.

Given the above definition and assuming that the lane marking observations $\boldsymbol{m}$ and road edge observations $\boldsymbol{r}$ are independent given $\boldsymbol{x}$, the joint probability distribution factorizes as

$$p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N | \boldsymbol{m}, \boldsymbol{r}) \propto \prod_{(i) \in \mathcal{I}_m} \phi_i(\boldsymbol{x}_i, \boldsymbol{m}) \prod_{(i) \in \mathcal{I}_r} \phi_i(\boldsymbol{x}_i, \boldsymbol{r}) \prod_{(i,j) \in \mathcal{E}} \psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j), \tag{4.2}$$

where $\mathcal{I}_m$ denotes the indexes of the set of cliques that are contained in $\boldsymbol{x} \cup \boldsymbol{m}$ and $\mathcal{I}_r$ the indexes of the cliques in $\boldsymbol{x} \cup \boldsymbol{r}$. The above factorization is also shown in Fig. 4.4. This figure shows a CHM for a two lane road that is represented by the root $\boldsymbol{x}^r_{31}$. This road is bounded by road edges (e.g., curbstones), and the individual lanes are separated by lane markings. The road is decomposed in a left lane $\boldsymbol{x}^l_{29}$ and a right lane $\boldsymbol{x}^l_{30}$. These lanes are then recursively decomposed

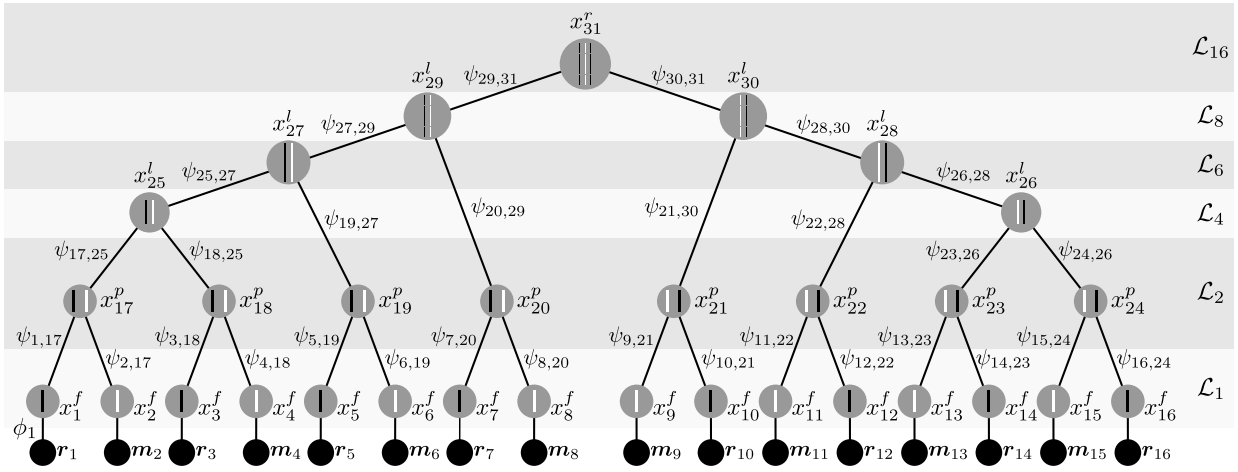


Figure 4.4: CHM of a two lane road. This figure shows the factorization of the joint probability distribution in Eq. 4.2 using an undirected graphical model. Hidden random variables are depicted in grey and symbols illustrate their type, i.e., features, patches, lanes and multi-lane roads. Observable variables are shown in black and dependencies between random variables are highlighted using edges.

into lane-segments, local driveable areas $\boldsymbol{x}_i^p$ and features $\boldsymbol{x}_i^f$ which are directly observable.

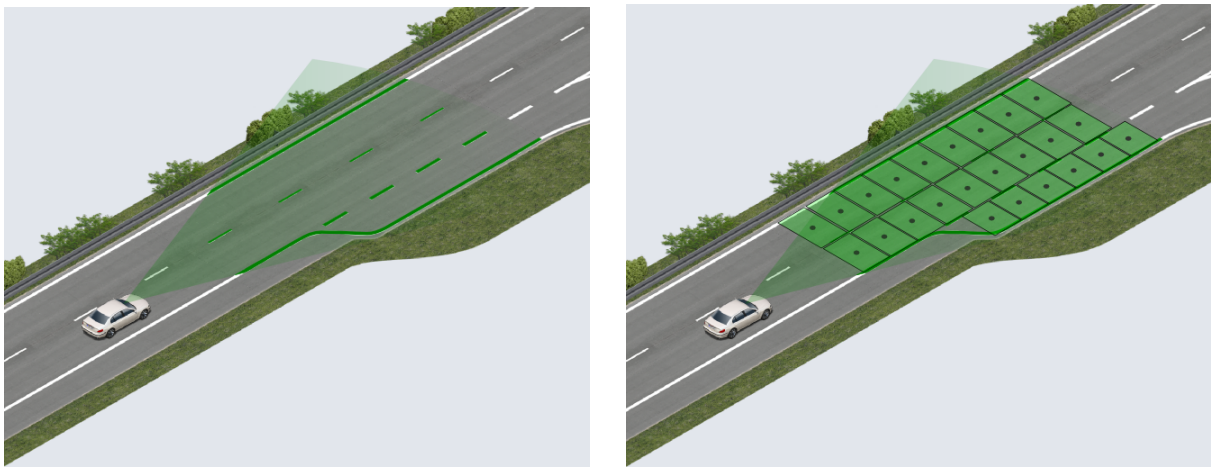
The specification of the hidden variables $\boldsymbol{x}$ comprised in the CHM and their spatial dependencies are given in the following.

4.2.1 Features and Local Driveable Areas

Recall that the leaves of the model comprise the lane marking observations $\boldsymbol{m}$ and the road edge observations $\boldsymbol{r}$. Each observable variable $\boldsymbol{m}_i \subseteq \boldsymbol{m}$ or $\boldsymbol{r}_i \subseteq \boldsymbol{r}$ is connected by an edge to a hidden feature variable $\boldsymbol{x}_i^f = (x_i, y_i, \vartheta_i)$ on the first level of the CHM (see Figure 4.6a). Further, each of the edges between hidden and observable variables have an associated observation potential as defined in Equation 4.1.

Definition 2 (Hierarchical Level)**.** *The level n of a variable in the model is directly determined by the number of associated low-level features and is denoted by $\mathcal{L}_n$. Feature variables $\boldsymbol{x}_i^f$ have exactly one associated feature and hence define the first level $\mathcal{L}_1$ of the CHM. Patch variables ,on the other hand, comprise two features and thus comprise the second level $\mathcal{L}_2$ of the CHM.*

The next higher level of the CHM comprises hidden random variables that define finite driveable areas and are referred to as patches. Each patch is defined by a left and a right lane boundary feature (i.e., lane marking or road edge), as shown in Figure 4.5. Formally, patches are defined by a five-dimensional state vector



(a) (b)

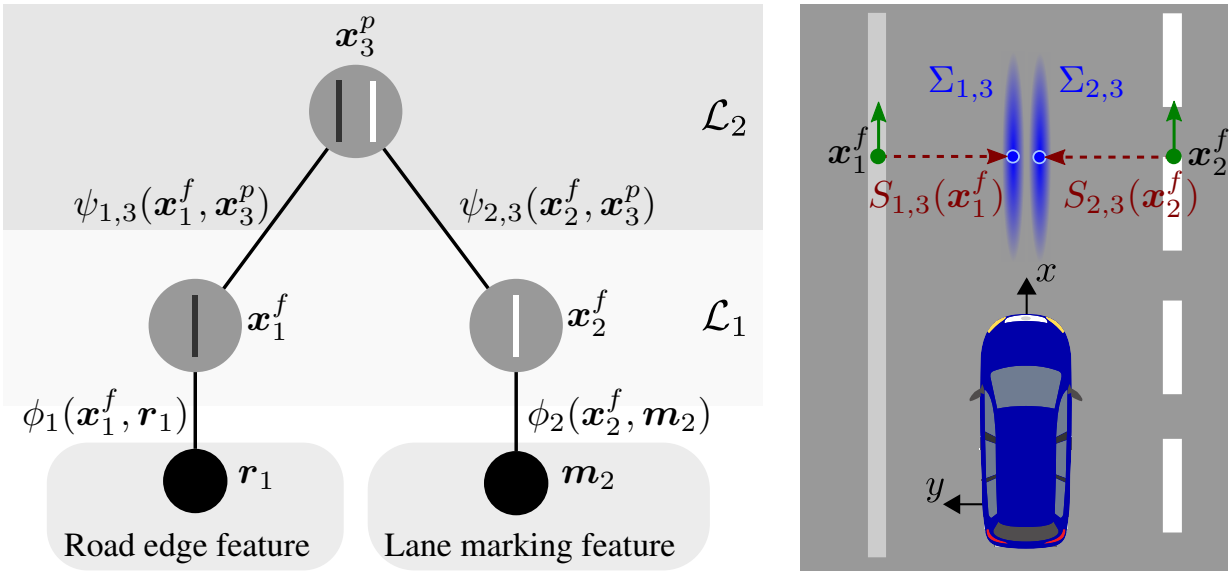
Figure 4.5: Patch-based lane representation. (a) Detected lane marking features. (b) Patches representing finite driveable areas of the lane.

$\boldsymbol{x}_j^p = (x_j, y_j, \vartheta_j, w_j, \nu_{l_p})$. Here, $(x_j, y_j) \in \mathbb{R}$ is the patch location, $\vartheta_j \in [0, 2\pi)$ its orientation angle, $w_j \in \mathbb{R}^+$ its width and $\nu_{l_p} \in \mathbb{R}^+$ its length (see Figure 4.5). The patch length ν_{l_p} is a constant design parameter that specifies the segmentation of the lane centerline, as detailed in Section 4.2.2.

Definition 3 (Patch)**.** *A patch $\boldsymbol{x}^p$ specifies a local driveable area and is defined by a left and a right low-level feature. The position $(x, y) \in \mathbb{R}^2$ and orientation $\vartheta \in [0, 2\pi)$ of a patch are defined in the vehicles coordinate frame. The width $w \in \mathbb{R}^+$ of a patch is equivalent to the distance between the two associated features. The length $\nu_{l_p} \in \mathbb{R}^+$ of a patch is a constant design parameter.*

In the CHM patches are represented by hidden random variables $\boldsymbol{x}_i^p$ on $\mathcal{L}_2$, as shown in Figure 4.6a. It can be seen that observable variables $\boldsymbol{m}_i$ and $\boldsymbol{r}_i$ comprise the leaves of the CHM, while the hidden levels $\mathcal{L}_1$ and $\mathcal{L}_2$ correspond to feature variables $\boldsymbol{x}_i^f$ and patch variables $\boldsymbol{x}_j^p$, respectively. The edges of this model represent the observation potentials given by Equation 4.1. Further, edges between hidden feature variables $\boldsymbol{x}_i^f$ and hidden patch variables $\boldsymbol{x}_j^p$ encode weak spatial constraints which are modeled as

$$\psi_{i,j}(\boldsymbol{x}_i^f, \boldsymbol{x}_j^p) = \mathcal{N}\left(\boldsymbol{x}_j^p; S_{i,j}(\boldsymbol{x}_i^f), \boldsymbol{\Sigma}_{i,j}\right), \tag{4.3}$$



(a) (b)

Figure 4.6: CHM of a patch. (a) Graphical model of a patch, where nodes correspond to observable variables $\boldsymbol{m}_i$ and $\boldsymbol{r}_i$ as well as to hidden feature variables $\boldsymbol{x}_i^f$ and the hidden patch variable $\boldsymbol{x}_i^p$. The edges encode spatial constraints $\psi_{i,j}(\boldsymbol{x}_i^f, \boldsymbol{x}_j^p)$ between pairs of hidden variables and observation potentials. (b) Illustration of the modeled spatial constraints, where spatial uncertainties are depicted by showing 2D Gaussian distributions. Here, dark colors correspond to more likely locations.

where for a left feature the transformation function

$$S_{i,j}(\boldsymbol{x}_i^f) = \begin{pmatrix} x_i \\ y_i \\ \vartheta_i \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} \frac{1}{2}\sin(\vartheta_i) & 0 & 0\ 0\ 0 \\ 0 & -\frac{1}{2}\cos(\vartheta_i) & 0\ 0\ 0 \\ 0 & 0 & 1\ 0\ 0 \\ 0 & 0 & 0\ 1\ 0 \\ 0 & 0 & 0\ 0\ 1 \end{pmatrix} \begin{pmatrix} \nu_{w_p} \\ \nu_{w_p} \\ 0 \\ \nu_{w_p} \\ \nu_{l_p} \end{pmatrix} \tag{4.4}$$

returns the predicted location of variable $\boldsymbol{x}_j^p$ based on the expected lane width $\nu_{w_p} \in \mathbb{R}^+$ and the state of variable $\boldsymbol{x}_i^f$. The covariance matrix $\boldsymbol{\Sigma}_{i,j} \in \mathbb{R}^{4\times4}$ is a design parameter that allows to express uncertainties regarding the spatial dependencies, as illustrated in Figure 4.6b.

An illustration of the spatial dependencies between the feature variables $\boldsymbol{x}_1^f$ and $\boldsymbol{x}_2^f$ and the patch variable $\boldsymbol{x}_3^p$ is given in Figure 4.6b. It can be seen how each feature predicts the location of the patch in the lane-center based on the a priori patch width ν_{w_p}. However, since the actual patch width is unknown, the individual predictions are not exact. In order to model this spatial uncertainties, the

covariance matrix $\mathbf{\Sigma}_{i,j}$ of the spatial constraint $\psi_{i,j}(\boldsymbol{x}_i^f, \boldsymbol{x}_j^p)$ is used. For example, in Figure 4.6b, both covariance matrices $\mathbf{\Sigma}_{1,3}$ and $\mathbf{\Sigma}_{2,3}$ model a small uncertainty orthogonal to the lane boundaries, while the uncertainty in lane boundary direction is relatively large. This definition of spatial uncertainties has two key motivations. First, clutter arising in the center of the lane amounts in patch prediction located in regions of low belief which ensures that clutter has a small influence on the overall results. Second, the feature extraction only obtains a finite set of discrete features. Particularly, if both the left and the right lane boundary are dashed it is likely that only few features can be extracted from these lane boundaries. Thus the true lane boundary is poorly approximated by the feature set. In order to account for this issue, a relatively large uncertainty along the lane ceterline is modeled. During inference, this ensures that even for a sparse feature set valid patch hypotheses are generated.

The key benefit of the proposed patch representation is its generality. In principal, it can be composed of any local lane or road cue that allows to predict the lane center. Further, it allows to cope with clutter arising in the lane center, caused by e.g., discontinuities in the road texture or color. Moreover, most roads can be defined as a set of patches. This reduces not only model complexity, but also allows to develop efficient inference algorithms, as detailed in Section 4.3.2.

4.2.2 Local Driveable Areas and Lanes

Patches, as introduced in the preceding section, represent local driveable areas. Consequently, composing lanes amounts in combining several patches. Hence, lanes are defined as a composition of a finite number of N_p individual patches $\boldsymbol{x}^p$, its width w^l and its length l^l as $\boldsymbol{x}_i^l = \{\boldsymbol{x}_1^p, \ldots, \boldsymbol{x}_{N_p}^p, w_i^l, l_i^l\}$. Here, each patch $\boldsymbol{x}_i^p = (x_i, y_i, \vartheta_i)$ is defined by its position $(x_i, y_i) \in \mathbb{R}^2$ and orientation $\vartheta_i \in [0, 2\pi)$. Further, the length of a lane $l_i^l \in \mathbb{R}^+$ is defined as the sum of Euclidean distances $d(\cdot)$ between subsequent lane segments as

$$l_i^l = \sum_{n=1}^{N_p-1} d(x_n^p, x_{n+1}^p). \tag{4.5}$$

Definition 4 (Lane)**.** *A lane $\boldsymbol{x}^l = \{\boldsymbol{x}_1^p, \ldots, \boldsymbol{x}_{N_p}^p, w^l, l^l\}$ is composed of a finite number of N_p patches $\boldsymbol{x}_i^p = (x_i, y_i, \vartheta_i)$, where $(x_i, y_i) \in \mathbb{R}^2$ and $\vartheta_i \in [0, 2\pi)$ are the position and orientation of a patch, respectively. Further, w^l specifies the width of the lane. Thus, a lane is defined as a polygonal path with piecewise constant orientation and width. The length $l^l \in \mathbb{R}^+$ of a lane is implicitly given by the number of comprised patches as defined in Equation 4.5.*

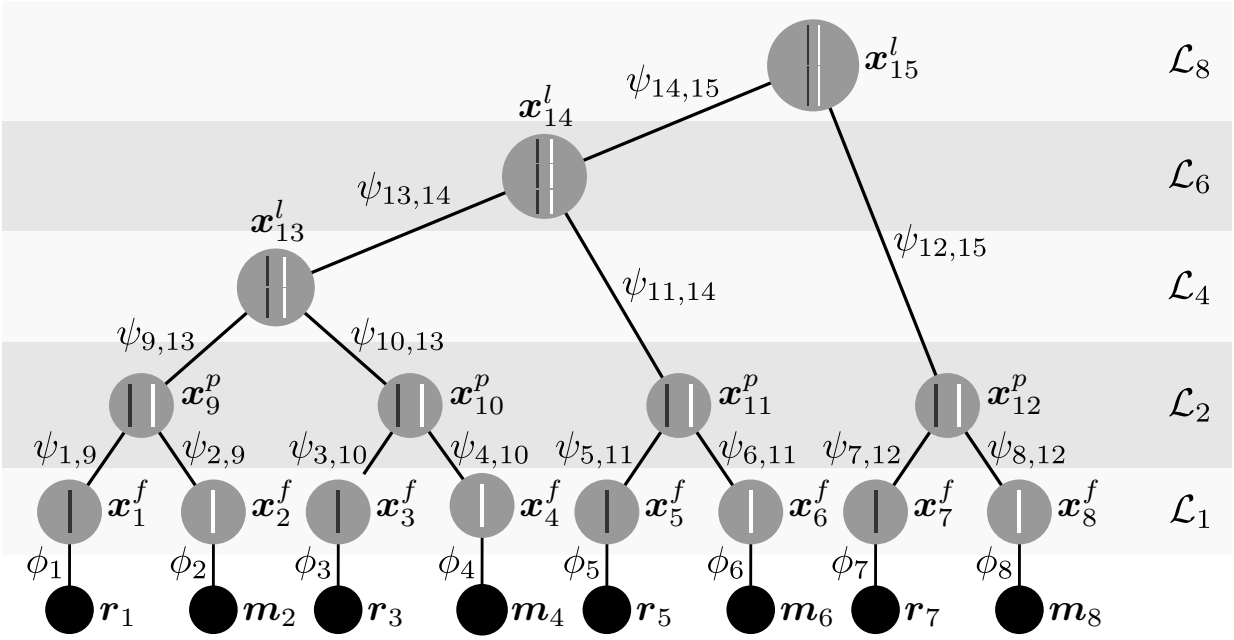


Figure 4.7: CHM of a lane with multiple levels. ($\mathcal{L}_1$) observable low-level features, ($\mathcal{L}_2$) patches and ($\mathcal{L}_4 - \mathcal{L}_8$) lanes of different length. The length of lanes is implicitly given by the number of corresponding patches. The edges of the graphical model encode observation potentials $\phi_i(\boldsymbol{x}_i, \boldsymbol{m}_i)$ and $\phi_i(\boldsymbol{x}_i, \boldsymbol{r}_i)$ as well as spatial dependencies $\psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j)$.

The hierarchical structure of the proposed lane representation is depicted in Figure 4.7, where for illustrative simplicity, a lane composed of only four patches[2] is depicted. Further, the short notations $\phi_i := \phi_i(\boldsymbol{x}_i, \boldsymbol{y}_i)$ and $\psi_{i,j} := \psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ are used. As before the levels $\mathcal{L}_1$ and $\mathcal{L}_2$ comprise feature variables $\boldsymbol{x}^f_i$ and patches variables $\boldsymbol{x}^p_i$. The level $\mathcal{L}_4$, $\mathcal{L}_5$ and $\mathcal{L}_6$ comprise lane variables, where the length l^l_i increases with each level of the hierarchy.

A key advantage of the proposed hierarchical lane representation is the convenient from of potentials which is critical for low computational complexity during inference. The CHM of lanes comprises two additional spatial constraints. Namely, constraints $\psi_{i,j}(\boldsymbol{x}^p_i, \boldsymbol{x}^l_j)$ encoding the spatial dependencies between patch and lane variables and constraints $\psi_{i,j}(\boldsymbol{x}^l_i, \boldsymbol{x}^l_j)$ between neighboring lane variables. Both of these spatial constraints are defined as multivariate diagonal Gaussian model as in Equation 4.3. However, they encode different spatial dependencies.

Particularly convenient are potentials $\psi_{i,j}(\boldsymbol{x}^p_i, \boldsymbol{x}^l_j)$ which encode the spatial relationship between the lane variables $\boldsymbol{x}^l_j$ and its child $\boldsymbol{x}^p_i$, since it is reasonable to

[2]During the experiments CHMs with lanes comprising up to 50 elements with an individual length of $\nu_{l_p} = 2\,m$ are used.

expect that $\boldsymbol{x}_i^p$ and its adjacent lane element have the same spatial configuration. On the other hand, spatial constraints between pairs of lane variables $\psi_{i,j}(\boldsymbol{x}_i^l, \boldsymbol{x}_j^l)$ are used to predict the expected location of the subsequent lane element using the transformation function

$$S_{i,j}(\boldsymbol{x}_i^l) = \begin{pmatrix} x_i \\ y_i \\ \vartheta_i \\ w_i \end{pmatrix} + \begin{pmatrix} \cos(\vartheta_i) & 0 & 0 & 0 \\ 0 & \sin(\vartheta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \nu_{l_p} \\ \nu_{l_p} \\ 0 \\ 0 \end{pmatrix}, \tag{4.6}$$

where ν_{l_p} is the constant patch length introduced in the previous section. In the example shown in Figure 4.7, these dependencies are encoded by $\psi_{13,14}$ and $\psi_{14,15}$, while the transformation function $S_{i,j}(\boldsymbol{x}_i^p)$ of $\psi_{9,13}$, $\psi_{10,13}$, $\psi_{11,14}$ and $\psi_{12,15}$ simply returns the mean of the associated patch variable.

Another key benefit of the proposed lane representation is that it allows to represent lanes of different length. This is essential since the goal is to reason about roads composed of multiple-lanes with potentially different length. This is particularly important in intersection, because they often comprise merging or splitting lanes. This aspect is further investigate in Section 5.3.2. Further, the definition of lanes as a composition of similar elements leads to a high reusability of parts which is beneficial for the development of the efficient inference algorithms presented in Section 4.3.2.

Most importantly, however, is the great flexibility of the proposed lane representation. In contrast to many state-of-the-art approaches, it does not impose hard constraints on the longitudinal lane geometry (e.g., clothoid or spline). Further, lanes are not restricted to a specific lateral model (e.g., parallel lanes or constant lane width). This great flexibility is a key benefit of the presented approach, since it makes it applicable to scenarios beyond highways and highway-like roads.

Yet, most ADAS applications involving vehicle control require a smooth lane representation and a small number of false detections. This commonly leads to the introduction of model assumptions on the lateral and longitudinal road topology [Hil12, McC06]. While in principal the proposed lane representation could easily be extend to e.g., a clothoid model by introducing additional model parameters, it would limit its field of applicability. Therefore, instead of introducing specific lateral or longitudinal lane models, such assumptions are considered as properties of a specific road type as detailed next.

4.2.3 Lanes and multi-lane Roads

In the proposed model, roads are the most complex objects and correspond to the root of the CHM, as shown in Figure 4.4. Roads are composed of a set of lanes and introduce constraints on both the lateral and the longitudinal road structure.

Definition 5 (Road)**.** *A road is defined as* $\boldsymbol{x}^r = \{\boldsymbol{x}^l, (\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{N_l}), (w_1, \ldots, w_{N_l})\}$. *Here,* $\boldsymbol{x}^l = \{\boldsymbol{x}^p_1, \ldots, \boldsymbol{x}^p_{N_p}, l^l\}$ *is the road centerline that is composed of* N_p *lane segments, as in Definition 4. Further, the lateral offset vectors* $(\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{N_l})$ *define the offsets between each of the* N_l *lanes of the road to the road centerline. These offset vectors are either constants to encode parallel lanes or functions of the road centerline to encode splits or merges. Finally, each value* $w_i \in \mathbb{R}^+, i \in N_l$ *defines the width of each lane.*

Thus, roads comprise all available information of the levels below and are considered as the output level for ADAS applications.

An example of a hierarchical representation of a road is depicted in Figure 4.4, where a CHM of a two-lane road is shown. It can be seen that the layer $\mathcal{L}_8$ comprise the left lane $\boldsymbol{x}^l_{29}$ and right lane $\boldsymbol{x}^l_{30}$, where the structure of their subtrees is equivalent to the CHM shown in Figure 4.7. The potentials $\psi_{29,31}(\boldsymbol{x}^l_{29}, \boldsymbol{x}^r_{31})$ and $\psi_{30,31}(\boldsymbol{x}^l_{30}, \boldsymbol{x}^r_{31})$ are used to encode the spatial dependencies between the road variable $\boldsymbol{x}^r_{31}$ and its lane subparts $\boldsymbol{x}^l_{29}$ and $\boldsymbol{x}^l_{30}$.

For example, if a road $\boldsymbol{x}^r$ has a parallel lane structure and its centerline piecewise linear path, the spatial dependency to its i-th lane with $i \in N_l$ is defined as

$$\psi_{i,j}(\boldsymbol{x}^l_i, \boldsymbol{x}^r_j) = \psi_{i,j}(\boldsymbol{x}^p_i, \boldsymbol{x}^p_j)\psi_{i,j}(w_i, w_{j,i}), \tag{4.7}$$

where the spatial dependency of the road centerline and the lane centerline is defined as

$$\psi_{i,j}(\boldsymbol{x}^p_i, \boldsymbol{x}^p_j) = \sum_{k=1}^{N_p} \mathcal{N}\left(\boldsymbol{x}^p_{i,k}; \boldsymbol{x}^p_{j,k} + \boldsymbol{n}^p_{j,k}\boldsymbol{\lambda}_{j,i}, \boldsymbol{\Sigma}_{i,k}\right). \tag{4.8}$$

Here, $\boldsymbol{n}^p_k$ is the normal vector of the k-th road centerline element. Furthermore, $\boldsymbol{\Sigma}_{i,k}$ is the covariance matrix that models a spatial uncertainty, as illustrated in Figure 4.6b. Further,

$$\psi_{i,j}(w_i, w_{j,i}) = \mathcal{N}\left(w_i; w_{j,i}, \delta_{i,j}\right) \tag{4.9}$$

defines the dependency of the width w_i of the lane and the expected lane width $w_{j,i}$ of the i-th lane of the road. Note that, the above example is only one possibility of defining the dependencies between lanes and roads. In fact, a straight forward extension to the presented model is to use road centerlines that follow a clothoid or a spline model.

The motivation for introducing model assumptions on the road level is twofold. First, it allows to benefit of the rich amount of a priori scene knowledge contained in road construction guidelines [ras95] which specify the topology of roads rather

than the geometry of lanes. Incorporating such a priori knowledge into the spatial constraints of the model aims to increase the reliability of the proposed approach, as further detailed in Section 5.3.1. Second, in Section 5.3.2, the proposed model is generalized to scenario with diverse topologies. Towards this goal, different road types are introduced, each representing a specific road topology, e.g., road with two parallel lanes or road with two splitting lanes. In the CHM introducing new road types only requires to modify the road variable and the corresponding spatial constraints. Hence, introducing new road types does not alter the lower-levels of the CHM. In fact, this is a key aspects of the presented approach, since it allows to compose any road type of the lanes specified in the previous section, and thus to define different roads as a set of common parts. This allows to not only generalizing it to scenario with diverse topologies, but also to account for heterogeneous demands of ADAS applications and to develop computational efficient inference algorithms. A detailed discussion of these aspects is given in Section 5.3.2.

Note that, like most models, the above formulation is only an approximation. For example, in Figure 4.4, the graphical model implicates that the two low-level variables $\boldsymbol{x}_8^f$ and $\boldsymbol{x}_8^f$ are conditional independent given the root. However, in practice they are often not independent, since the observations $\boldsymbol{y}_8$ and $\boldsymbol{y}_9$ correspond to the same lane or road cue. However, as the experimental results show this approximation works well in practice.

This thesis mainly focuses on the task of lane and road perception based on low-level sensory cues, as it is a key advantage of the proposed framework. However, the presented framework also offers the possibility to make high-level variables observable. For example, vehicles detection or vehicle trajectories could be used to make patch variables or lane variables observable. Although including such high-level observations is expected to increase the reliability in scenarios, such as construction sites or snow covered roads, this aspect is not further investigated (see Section 8.2).

4.2.4 Periodic Variables and Marginal Influence

One challenge in modeling both observation potential $\phi_i(\boldsymbol{x}_i, \boldsymbol{y}_i)$ (see Section 4.1.2) and spatial constraints $\psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is that the random variables representing angles $\vartheta \in [0, 2\pi)$ do not possess a natural origin. In order to avoid issues regarding the choice of the origin, the angular dimension are modeled by a von Mises-Fisher distribution [Ban05]

$$\mathcal{M}(\vartheta; \mu, \tau) = \frac{1}{2\pi I_0(\tau)} \exp\{\tau \cos(\vartheta - \mu)\} \tag{4.10}$$

which is a generalization of a Gaussian distribution to an arbitrary-dimensional shell. Here, $I_0(\cdot)$ denotes the modified Bessel function of the first kind and order

zero. τ is known as the concentration parameter which corresponds to the inverse variance σ^{-2} of a Gaussian distribution $\tau \approx \sigma^{-2}$. The von Mises-Fisher distribution is a convenient choice, since it can be derived from a bivariate Euclidean Gaussian distribution with mean $(\cos\vartheta, \sin\vartheta)$ [Sud06] and thus easily be added to the Gaussian model of the spatial constraints.

A convenient property of the proposed model is that all variables are specified in respect to the vehicle coordinate system. Hence, the spatial constraints only depend on the difference between neighboring variables $\psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \tilde{\psi}_{i,j}(\boldsymbol{x}_i - \boldsymbol{x}_j)$ so that the marginal influence $\varphi_{i,j}(\boldsymbol{x}_j)$ can be neglected (see Section 3.4.3).

4.3 Inference of a single Road Topology

In the presented framework, the task of lane and road detection is equivalent to computing the marginal posterior distribution $p(\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | \boldsymbol{m}, \boldsymbol{r})$ or belief $b_i(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ over all hidden variables in the CHM.

As illustrated in Figure 3.7 the belief estimate $b_i(\boldsymbol{x}_i)$ of a single part $\boldsymbol{x}_i$ is computed by combining all incoming messages at variable $\boldsymbol{x}_i$ with the local observation potential. For the proposed CHM the belief update equation is given by

$$b_i(\boldsymbol{x}_i) \propto \phi_i(\boldsymbol{x}_i, \boldsymbol{m})\, \phi_i(\boldsymbol{x}_i, \boldsymbol{r}) \prod_{j \in \Xi(i)} m_{j,i}(\boldsymbol{x}_i) \prod_{k \in \Gamma(i)} m_{k,i}(\boldsymbol{x}_i), \tag{4.11}$$

where the two products contain messages from the children $\Xi(i)$ and the parents $\Gamma(i)$ of node i, respectively. A convenient property of the presented framework is that evidence is exclusively injected into the model via the leaves which allows to perform the belief update in two stages as detailed in Section 3.4.4. First, the bottom up belief state

$$b_i^-(\boldsymbol{x}_i) \propto \phi_i(\boldsymbol{x}_i, \boldsymbol{m})\, \phi_i(\boldsymbol{x}_i, \boldsymbol{r}) \prod_{j \in \Xi(i)} m_{j,i}(\boldsymbol{x}_i) \tag{4.12}$$

of the variables in the model is computed by passing messages from the observable leaves to the root. During this phase, each variable $\boldsymbol{x}_i$ only receive messages from their children $\Xi(i)$. Second, during the top down phase, messages are passed down from the root to the leaves. This allows to compute the belief $b_i(\boldsymbol{x}_i)$ by combining the bottom up belief with the messages each variables $\boldsymbol{x}_i$ receives from its parent $\Gamma(i)$ as

$$b_i(\boldsymbol{x}_i) \propto b_i^-(\boldsymbol{x}_i) \prod_{k \in \Gamma(i)} m_{k,i}(\boldsymbol{x}_i). \tag{4.13}$$

Generally, these two phases have fundamentally different goals. While the aim of the bottom up phase is the fast generation of high-level hypotheses (e.g., lanes and roads), the top down phase ensures the overall consistency of parts and their subparts.

In order to cope with the continuous and high dimensional state space of the random variables as well as with the multimodal non-Gaussian sensory evidence, inference is performed using NBP [Isa03, Sud03]. NBP is a generalization of the particle filter [Isa98a, Kan95, Gor93] for approximate inference in arbitrary graphs (see Section 3.4.3). In NBP the belief $b_i(\boldsymbol{x}_i)$ is approximated by a set of L importance weighted samples $\{(\boldsymbol{s}_i^{(k)}, \pi_i^{(k)})\}_{k=1}^{L}$. Each of these samples $\boldsymbol{s}_i^{(k)}$ represents a hypothesis for the spatial configuration of part $\boldsymbol{x}_i$ and is drawn from the product distribution

$$\boldsymbol{s}_i^{(k)} \sim \prod_{j \in \Xi(i)} m_{j,i}(\boldsymbol{x}_i) \prod_{k \in \Gamma(i)} m_{k,i}(\boldsymbol{x}_i) \tag{4.14}$$

according to the nearest neighbor product sampling method [Spe13] presented in Section 3.4.3. The weight $\pi_i^{(k)}$ of each sample $\boldsymbol{s}_i^{(k)}$ is then computed as

$$\pi_i^{(k)} \propto \phi_i(\boldsymbol{s}_i^{(k)}, \boldsymbol{m})\, \phi_i(\boldsymbol{s}_i^{(k)}, \boldsymbol{r}). \tag{4.15}$$

This weight reflects the spatial plausibility of the hypothesis $\boldsymbol{s}_i^{(k)}$ as well as its plausibility in respect to the sensory evidence, used to instantiated the observable variables $\boldsymbol{y}$. Finally, the computationally efficient rule of thumb [Sil86] is used to construct a kernel density estimation $b_i(\boldsymbol{x}_i)$ from the raw sample set $\{(\boldsymbol{s}_i^{(k)}, \pi_i^{(k)})\}_{k=1}^{L}$, by assigning a Gaussian smoothing kernel to each sample. This final step is needed to ensure that the message product is non-degenerative (see Section 3.4.3).

In the following, the inference algorithm used in the proposed hierarchical framework is detailed in several steps: Section 4.3.1 details the fundamental bottom up message passing and message fusion algorithm, Section 4.3.2 presents how part-sharing [Zhu10, Spe13] is used to avoid redundant computations during the bottom up phase, and Section 4.3.3 introduces a novel depth-first message passing schedule for lane detection [Töp13] that performs inference in several sweeps and is particularly suitable for real-time applications.

4.3.1 Message Fusion and Outlier Handling

In NBP, the messages used in standard belief propagation (see Section 3.4.2) are approximated by a smoothed particle set, and the conditional distribution used in standard particle filtering is replaced by the product of incoming messages.

The CHM in this thesis comprises two kinds of probabilistic constraints. First, potentials $\psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ encoding the spatial dependencies between pairs of hidden variable $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Second, observation potentials $\phi_i(\boldsymbol{x}_i, \boldsymbol{m})$ and $\phi_i(\boldsymbol{x}_i, \boldsymbol{r})$, as presented in Section 4.1.2. In order to simplify the following expositions, it is first assumed that the observations $\boldsymbol{m}$ and sr can be decomposed in local measurements $\boldsymbol{m}_i \subseteq \boldsymbol{m}$ and $\boldsymbol{r}_i \subseteq \boldsymbol{r}$. This allows to formulate single node observation potential $\phi_i(\boldsymbol{x}_i, \boldsymbol{m}_i)$ and $\phi_i(\boldsymbol{x}_i, \boldsymbol{r}_i)$. In this case, messages can generally be computed recursively according to Equation 3.33 as

$$m_{j,i}(\boldsymbol{x}_i) = \int_{\boldsymbol{x}_j} \psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j)\, \phi_i(\boldsymbol{x}_i, \boldsymbol{m}_i)\, \phi_i(\boldsymbol{x}_i, \boldsymbol{r}_i) \prod_{k \in \Upsilon(j) \setminus i} m_{k,j}\left(\boldsymbol{x}_j\right) d\boldsymbol{x}_j, \tag{4.16}$$

where $\Upsilon\left(j\right)$ is the set of neighbors of variable $\boldsymbol{x}_j$. This equation can be rewritten as

$$m_{j,i}(\boldsymbol{x}_i) = \int_{\boldsymbol{x}_j} \psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j) b_j^-(\boldsymbol{x}_j) d\boldsymbol{x}_j \tag{4.17}$$

to make the bottom up belief explicit. The computation of the message update in Equation 4.16 requires to calculate the product of several incoming messages. The efficient computation of these message products is a key challenge in NBP. For example, if each of the L samples in the message is convolved with a Gaussian kernel, then the explicit product of d messages requires (L^d) time to compute, which is impractical in most cases. In order to keep inference tractable, nearest neighbor product sampling [Spe13] is employed as described in Section 3.4.3. The nearest neighbor product sampling allows to compute the approximated product in $\mathcal{O}(Ld)$ time, which brings along the much desired computational efficiency.

An illustrative example of the message passing framework is shown in Figure 4.8, where message passing is based on the graphical model depicted on the right. At the beginning of the bottom up phase, two messages $m_{1,3}(\boldsymbol{x}_3^p)$ and $m_{2,3}(\boldsymbol{x}_3^p)$ are computed according to Equation 4.16. Since NBP is used, the bottom up belief states $b_1^-(\boldsymbol{x}_1^f)$ and $b_2^-(\boldsymbol{x}_2^f)$ are approximated by a set of weighted sample. In Figure 4.8 the samples comprised in the feature variables $\boldsymbol{x}_1^f$ and $\boldsymbol{x}_2^f$ are depicted by showing light blue dots. These samples are used to construct the two messages $m_{1,3}(\boldsymbol{x}_3^p)$ and $m_{2,3}(\boldsymbol{x}_3^p)$, using the corresponding spatial constraints $\psi_{1,3}(\boldsymbol{x}_1^f, \boldsymbol{x}_3^p)$ and $\psi_{2,3}(\boldsymbol{x}_2^f, \boldsymbol{x}_3^p)$, respectively, as in Equation 3.38. In Figure 4.8a samples from theses messages are illustrated by showing the predicted patch location with blue dots. As can be seen, the individual features may not be able to predict the patch location very precisely. However, as Figure 4.8b shows, the prod-

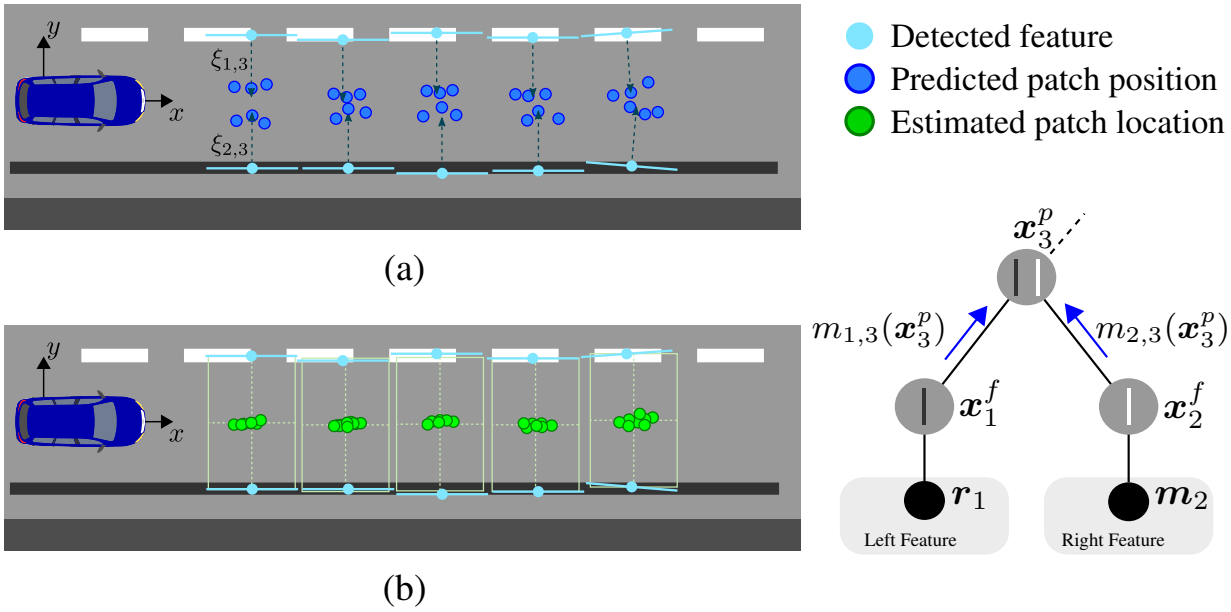


Figure 4.8: Message passing framework for an ideal feature set. In the proposed CHM feature variables send message to the patch variables predicting the location of patches. (a) Samples drawn from the messages predicting the location of patches. (b) Approximated belief $b_i^-(\boldsymbol{x}_3^p)$ after message fusion. Not illustrated are the Gaussian kernels which are assigned to each sample in order to smooth the belief estimate.

uct of the incoming messages at the patch variables produces precise estimates of the patch locations.

Partial Occlusions and Clutter: So far, it was assumed that inference is based on an ideal feature set. In a more realistic scenario, however, any lane detection approach has to deal with a large amount of outlier, due to clutter and occlusions. As discussed above CHMs directly address these issues, and therefore the presented framework shares this capability. This is achieved by the combination of both the outlier process (Section 4.1.2) and the a priori knowledge about the spatial configuration of parts encoded by the spatial constraints of the CHM.

In order to explain these mechanisms, a scenario with a partly occluded feature set is depicted in Figure 4.9a. As before, feature variables send messages $m_{j,i}(\boldsymbol{x}_j^p)$ to their associated patch variables containing predictions of possible patch locations during the bottom up phase. However, if features are occluded, such messages are empty. As depicted in Figure 4.9b, during message fusion these empty messages guarantee the result to be zero, and hence information provided by non-occluded features are eliminated.

In order to cope with this issue, the outlier process introduced in Section 4.1.2 is used, which ensures that messages are strictly positive by assigns a high vari-

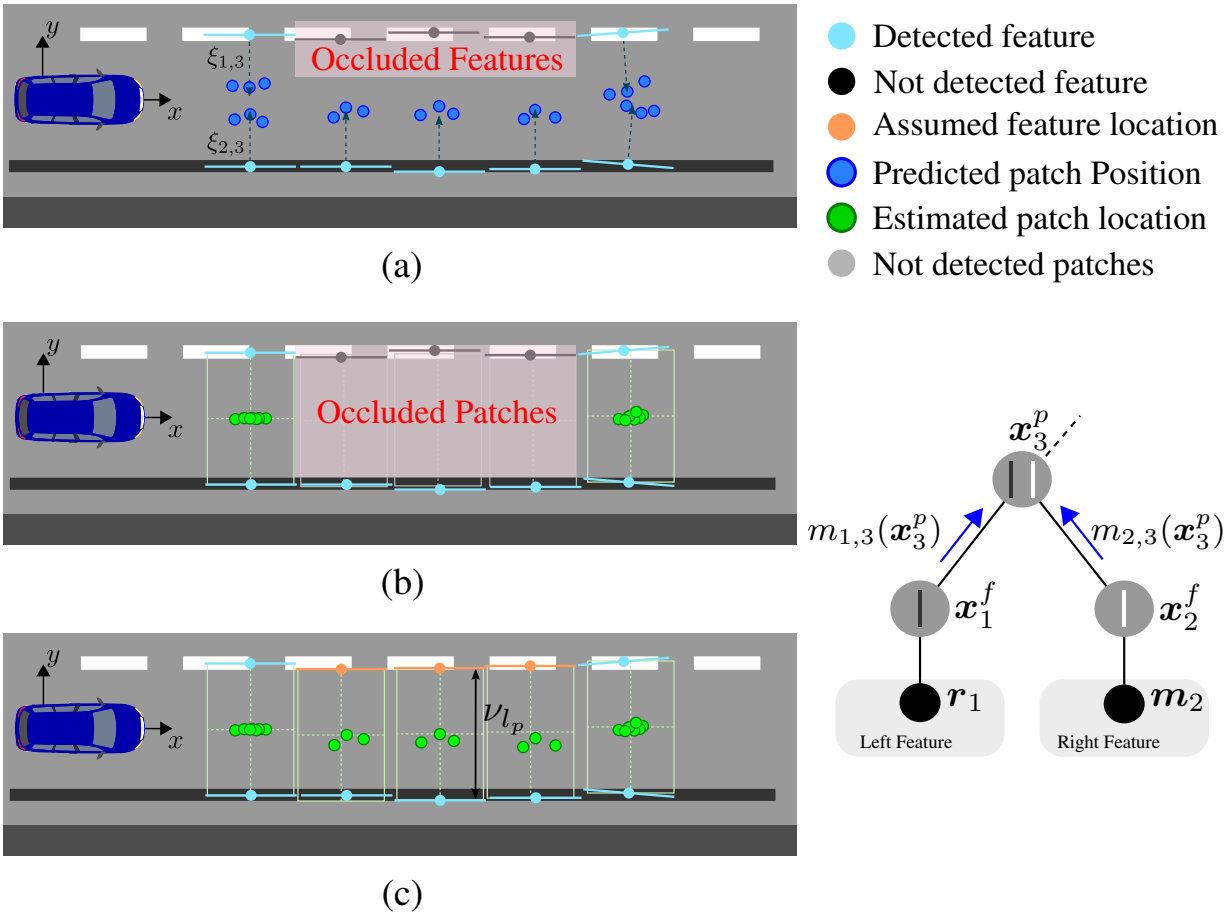


Figure 4.9: Message passing framework in the case of partial occlusion. (a) Predicted patch locations for a partly occluded feature set. (b) Belief $b_3^-(\boldsymbol{x}_3^p)$ over patch locations without outlier process. (c) Belief $b_3^-(\boldsymbol{x}_3^p)$ with outlier process.

ance Gaussian distribution to the belief state of the feature variables $\boldsymbol{x}_i^f$. As can be seen in Figure 4.9c, the outlier process ensures that partly occluded patch hypotheses are not eliminated. Instead, the location of partly occluded patches is estimated based on the assumed patch width ν_{w_p}, comprised in the spatial relation $\psi_{i,j}(\boldsymbol{x}_i^f, \boldsymbol{x}_i^p)$ (see Equation 4.3). However, since this location is only predicted by one corresponding feature, and further ν_{w_p} is usually not equivalent to the actual patch width, the patch location may not be constrained as tightly as in the non-occluded case. This spatial uncertainty is reflected by the associated importance weight $\pi_i^{p,(k)}$ which is computed according to Equation 4.15.

Similarly, a feature may lead to a patch hypothesis supported by the outlier process in the case of clutter. Consequently, both partly occluded and clutter hypotheses have relatively low corresponding weights. In fact, during the bottom up phase, it can not be distinguished between patches generated by clutter or partly occluded patches, and therefore it has to be allowed for both to be present in the

belief estimates. Even though this leads to the presence of many unlikely hypotheses it allows to avoid early decision which simplifies the detection of lanes and roads, as detailed in Section 4.3.3.

4.3.2 Part-Sharing

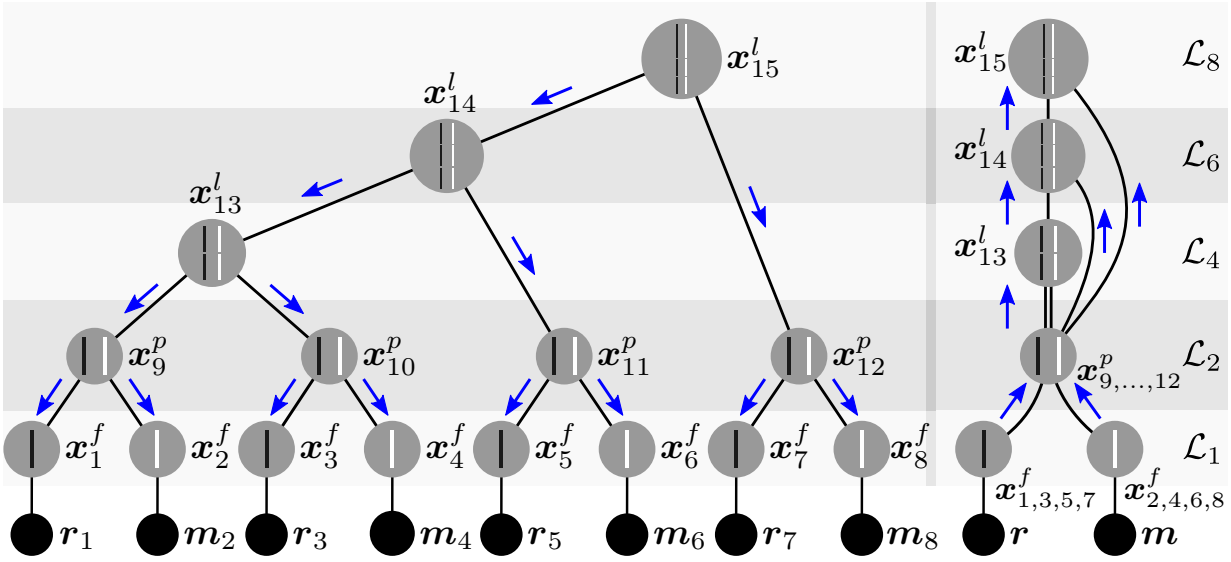
The previous section, introduced the message passing framework based on the assumption that observation $\boldsymbol{m}$ and $\boldsymbol{r}$ can be decomposed into local observations $\boldsymbol{m}_i \subseteq \boldsymbol{m}$ and $\boldsymbol{r}_i \subseteq \boldsymbol{r}$. In practice, however, such a decomposition is often unfeasible, and more importantly, the introduced framework derives this decomposition during inference. In this case, the message update is given as

$$m_{j,i}(\boldsymbol{x}_i) = \int_{\boldsymbol{x}_j} \psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j)\, \phi_i(\boldsymbol{x}_i, \boldsymbol{m})\, \phi_i(\boldsymbol{x}_i, \boldsymbol{r}) \prod_{k \in \Upsilon(j) \setminus i} m_{k,j}\left(\boldsymbol{x}_j\right) d\boldsymbol{x}_j. \tag{4.18}$$

In order to explain the difference to the inference problem addressed in the previous sections, let us consider the CHM of a lane depicted in Figure 4.7. As before, inference in this CHM can be performed using the bottom up/top down message passing schedule which begins with the computation of the bottom up belief over all feature variables $b_i^-(\boldsymbol{x}_i^f)$ conditioned on the observations $\boldsymbol{y}$. Since the bottom up belief state of all feature variables $b_i^-(\boldsymbol{x}_i^f)$ is now conditioned on all observations $\boldsymbol{y}$ it is equivalent for all feature variables. Consequently, all messages sent from the feature variables on $\mathcal{L}_1$ to the patch variables on $\mathcal{L}_2$ contain the same belief estimate. This means that the same message product has to be computed for each patch variable which leads to unnecessary computational complexity. In order to avoid such redundant computations, the propose framework employs part-sharing which is a method that originated in the field of vision based multi-view, multi-object detection [Zhu10] and was leveraged in [Spe11, Spe13] to 3d intelligent vehicle applications for efficient inference in Markov networks.

The fundamental idea of part-sharing is to merge those nodes i during the bottom up phase that receive the same messages from their children $\Xi(i)$. Thus, in the presented model, patch variables as well as variables in their subtrees can be combined as depicted in Figure 4.10b. This figure shows the resulting sharing structure that includes three sharing-nodes $\boldsymbol{x}^f_{1,3,5,7} = \{\boldsymbol{x}^f_1, \boldsymbol{x}^f_3, \boldsymbol{x}^f_5, \boldsymbol{x}^f_7\}$, $\boldsymbol{x}^f_{2,4,6,8} = \{\boldsymbol{x}^f_2, \boldsymbol{x}^f_4, \boldsymbol{x}^f_6, \boldsymbol{x}^f_8\}$ and $\boldsymbol{x}^p_{9,\ldots,12} = \{\boldsymbol{x}^p_9, \ldots, \boldsymbol{x}^p_{12}\}$ clarifying that the bottom up belief is shared between the combined hidden variables.

Using part-sharing, bottom up inference is based on this sharing structure, where the bottom up belief state of each variable is only calculated once, and then shared between its parents. This approach avoids not only redundant computations, but



(a) Compositional Hierarchy (b) Sharing Structure

Figure 4.10: Message passing using part-sharing. (b) During the bottom up phase messages (blue) are passed from the leaves to the root using the sharing structure. During this phase the belief is shared between nodes on $\mathcal{L}_1$ and $\mathcal{L}_2$ to avoid redundant computations. (a) During the top down phase, message passing is based on the structure of the CHM, since each node receives different contextual information from its parent.

also ensures that only one object instance has to be memorized. The bottom up phase ends, after the belief of the root is computed. Then, the top down message passing begins, as illustrated in Figure 4.10a. Since, the patch variables receive different messages from their parents during this phase, top down message passing is performed in the CHM. Before starting the top down phase, the sharing variables are decomposed into the comprised variables by assigning the bottom up belief of the sharing variable to all associated variables. Finally, the belief $b_i(\boldsymbol{x}_i)$ of each variable $\boldsymbol{x}_i$ is calculated by combining its bottom up belief with the incoming messages from its parents $m_{j,i}(\boldsymbol{x}_i), j \in \Gamma(i)$, according to Equation 4.11. This step is very important, since the messages of the parents contain contextual information, and therefore constrain the hypotheses to plausible domains.

Figure 4.11 illustrates the message passing framework using part-sharing by showing how the bottom up belief state is only computed once for similar parts, and shared during the top down phase. This example also clarifies the motivation for the proposed decomposition of roads into generic objects (e.g., features and patches). This decomposition guarantees a high degree of reusability which leads to a reduced computational complexity. Section 5.3.2 leverages this principle in order to formulate a model of roads with varying topologies. In this model, part-

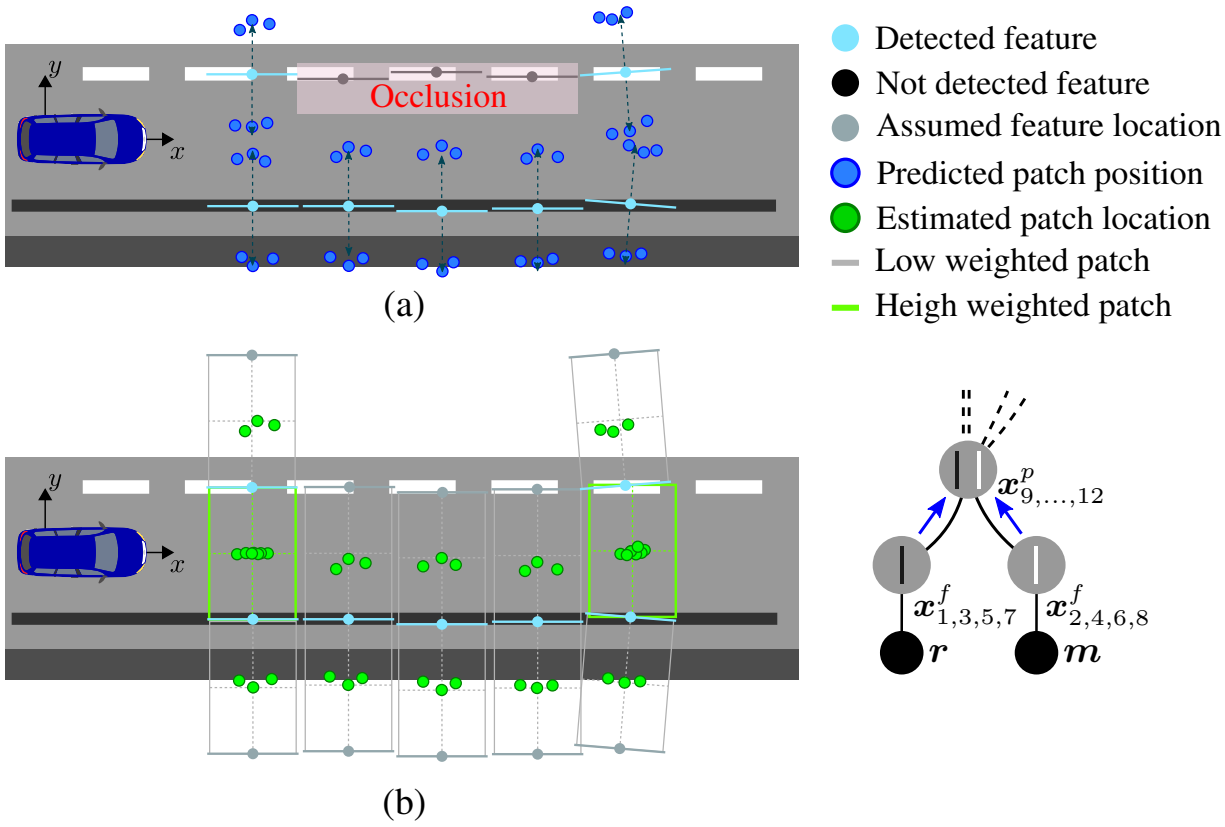


Figure 4.11: Message passing framework using part-sharing. (a) Each low-level feature variable sends messages to its associated patches. Samples from these messages are illustrated by showing predicted patch locations with blue dots. (b) The belief $b^-_{9,\ldots,12}(\boldsymbol{x}^p_{9,\ldots,12})$ over the patch locations is illustrated by showing green dots. Unlikely hypotheses (grey) are generated due to the presented approach for occlusions handling.

sharing is used not only to shared low-level components, but also to shared entire lanes between different road types.

Figure 4.11 shows another property of the proposed message passing framework. In contrast to the previous examples, each feature proposes locations of a left and a right patch. This is necessary, since during the bottom up phase a low-level feature can not be assigned to a specific feature variable $\boldsymbol{y}_i$. In other words, during the bottom up phase, it is not known if a feature is the right or the left feature of a patch. Thus, a left and a right patch location has to be proposed. In combination with the outlier process presented in Section 4.3.1, this leads to the presence of many unlikely hypotheses in the belief estimate (see Figure 4.11b).

Even though these additional hypotheses lead to a better approximation of the bottom up belief, they also cause additional computational complexity during inference. Particularly, during lane and road detection, since the number of possible lane hypotheses growth exponentially with the number of patch hypotheses. In order to cope with this issue, this thesis proposes the sequential message passing schedule presented in the next section.

4.3.3 Depth-first Message Passing

In NBP both performance and execution time crucially depend on the number of samples used during message passing. In general, using more samples results in a more accurate representation of the probability distribution $b_i(\boldsymbol{x}_i)$, but in turn it leads to additional computational complexity. This section introduces a novel sequential message passing schedule for lane detection [Töp13] that allows to reduce the number of samples required for lane and road detection by performing inference in a sequence of sweeps. This algorithm is based on the depth-first message passing algorithm for real-time inference in CHMs presented in [Spe13].

Lane detection: As detailed in Section 4.3.2, bottom up message passing is based on the sharing-structure which amounts in processing each level one by one (see Figure 4.10b). Intuitively, this can be understood as a breadth-first search in the hypotheses space, since on each level of the sharing structure all possible hypotheses are computed. The advantage of this approach is that it leads to a good approximation of the density distributions on all levels. The drawback of this procedure, however, is that the hypotheses space growth exponential with each level, and therefore computations are only tractable for a small number of levels. Furthermore, in the addressed application, the main goal is the fast detection of high-level hypotheses, and not an exhaustive search for low-level hypotheses. Therefore, this section introduces a sequential message passing schedule [Spe13, Töp13]

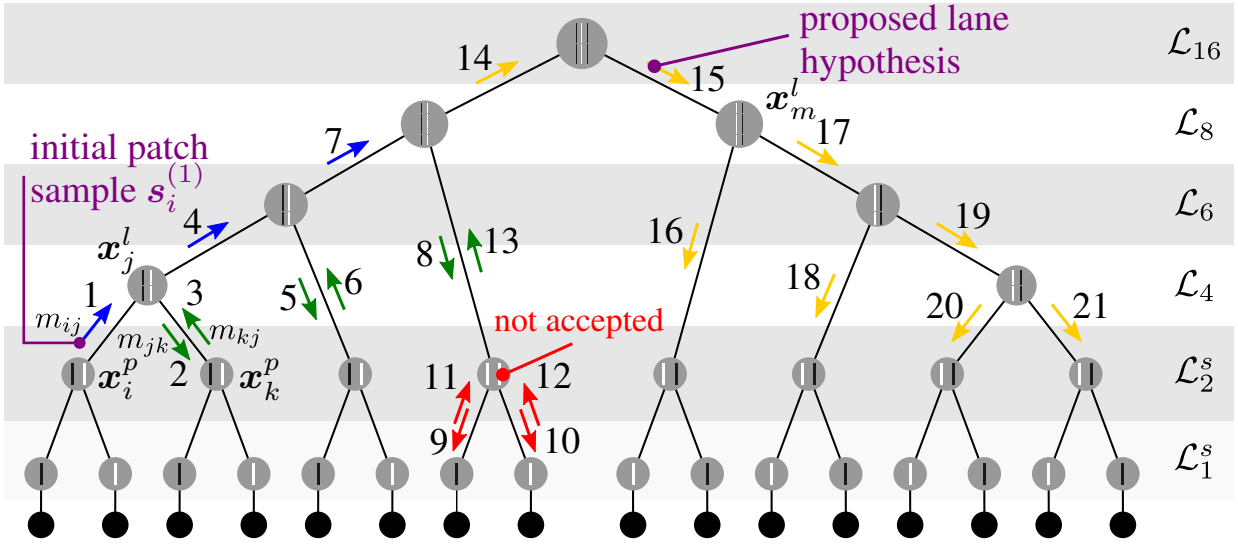


Figure 4.12: Lane and road detection using depth-first message passing. Based on an initially selected patch sample a lane hypothesis is generated (blue). An expectation-based step is used to ensure the consistency of the high-level hypothesis and sensory evidence (green). Further, road hypotheses can be evaluated using the CHM (orange). Note that the bottom up belief state of variables on level $\mathcal{L}_1^s$ and $\mathcal{L}_2^s$ is computed using part-sharing.

which is inspired by the depth-first traversal[3] for arbitrary tree structured graphs.

The fundamental idea of depth-first message passing is to perform bottom up message passing in several sweeps. In each sweep, only those patch hypotheses are propagated that are likely to be a part of valid high-level hypotheses (i.e., lanes and roads). In the case of lane and road detection, it is convenient to start the detection process from patches close to the vehicle, since they are located in areas of low uncertainty. After selecting a patch sample, the lane hypothesis can be extended away from the vehicle into areas of higher uncertainty. Towards this goal, a single patch sample $\{\boldsymbol{s}_i^{(k)}, \pi_i^{(k)}\}$ is selected from the nonparametric density $b_i^-(\boldsymbol{x}_i^p)$ in each sweep according to its weight

$$\boldsymbol{s}_i^{(k)} \sim \pi_i^{(k)}. \tag{4.19}$$

Subsequently, the selected sample is propagated through the CHM as shown in Figure 4.12. In this example, message passing is initiated at variable $\boldsymbol{x}_i^p$ by selecting a single sample $\boldsymbol{s}_i^{(1)}$ from the nonparametric density $b_i^-(\boldsymbol{x}_i^p)$, according to Equation 4.19. Then, Equation 3.38 is used to construct a message $m_{i,j}(\boldsymbol{x}_j^l)$ containing a single sample $\boldsymbol{s}_{i,j}^{(1)}$ predicting the location of $\boldsymbol{x}_j^l$. This message can

[3]Notice, depth-first traversal in tree structured graphs starts by first visiting the root, whereas the bottom up phase starts at the leaves of the graphical model.

now be used to update the belief $b_j^-(\boldsymbol{x}_j^l)$ by calculating the product of all incoming messages from its child $\boldsymbol{x}_k^p, k \in \Xi(j)$ (see Equation 4.11). This product is approximated using nearest neighbor product sampling [Spe13] which searches for samples $\boldsymbol{s}_{kj}^{(q_{nn})}$ in the incoming messages at $\boldsymbol{x}_j^l$ that are similar to $\boldsymbol{s}_{i,j}^{(1)}$. These samples are accepted according to the acceptance rate [Spe13]

$$A\left(\boldsymbol{s}_{kj}^{(q_{nn})}\right) = \exp\left(-\frac{1}{2}\left(\boldsymbol{s}_{i,j}^{(1)} - \boldsymbol{s}_{kj}^{(q_{nn})}\right)^T \boldsymbol{\Sigma}_{i,j}^{-1}\left(\boldsymbol{s}_{i,j}^{(1)} - \boldsymbol{s}_{kj}^{(q_{nn})}\right)\right), \quad (4.20)$$

where q_{nn} is the index of the nearest neighbor of the sample $\boldsymbol{s}_{i,j}^{(1)}$ and $\boldsymbol{\Sigma}_{i,j}$ is the covariance matrix of the spatial constraint $\psi_{i,j}(\boldsymbol{x}_i^p, \boldsymbol{x}_j^l)$. Using this acceptance rate, samples are accepted in a stochastic manner instead of making hard decision. This provides the possibility to preserve a certain amount of less likely hypothesis that may become more likely at a later stage of inference.

If no sample is accepted, a top down/bottom up search for evidence supporting the high-level hypothesis is initiated. This search can be thought of as an aligning process that ensures the consistency of the lane hypothesis and the low-level evidence.

This alignment process is illustrated in Figure 4.12 showing green messages. The alignment process has two distinct steps. First, a message $m_{j,k}(\boldsymbol{x}_k^p)$ is sent from variable $\boldsymbol{x}_j^l$ to variable $\boldsymbol{x}_k^p$, containing a single sample $\boldsymbol{s}_{jk}^{(1)}$. This single sample is again used in the nearest neighbor product sampling to search for samples in the messages $m_{h,k}(\boldsymbol{x}_k^p), h \in \Xi(k)$ send to $\boldsymbol{x}_k^p$ from its children. As before, samples $\boldsymbol{s}_{hk}^{(q_{nn})}$ are accepted according to Equation 4.20. However, since the messages $m_{h,k}(\boldsymbol{x}_k^p)$ contain the sensory evidence the acceptance rate is not used to decide if an additional top down search should be initiated, but to decide whether a sample is supported by an observation or if it corresponds to the outlier process (see Equation 4.1). This is crucial for handling occlusion as described in Section 4.3.1.

If for all messages $m_{h,k}(\boldsymbol{x}_k^p)$ a sample $\boldsymbol{s}_{hk}^{(q_{nn})}$ is accepted, the product of the incoming and the accepted sample is calculated which amounts in calculating the product of two Gaussian distributions. If no sample is accepted for some of the incoming messages, the contained samples are multiplied with the high-variance Gaussian outlier process. In the following, the product result is used to send a single sample message $m_{k,j}(\boldsymbol{x}_j^l)$ back to $\boldsymbol{x}_j^l$. Finally, the belief update at variable $\boldsymbol{x}_j^l$ is performed by computing the product of $m_{i,j}(\boldsymbol{x}_j^l)$ and $m_{k,j}(\boldsymbol{x}_j^l)$.

The above process is repeated for each lane variable in the CHM, and hence terminates if the last lane level is processed finishing a single sweep of the algorithm. Since a single sweep does not allow for a good approximation of the nonparametric densities of the lane variables, lane detection is performed in several sweeps. Each of these sweeps adds samples to the belief estimates, and thus improves the

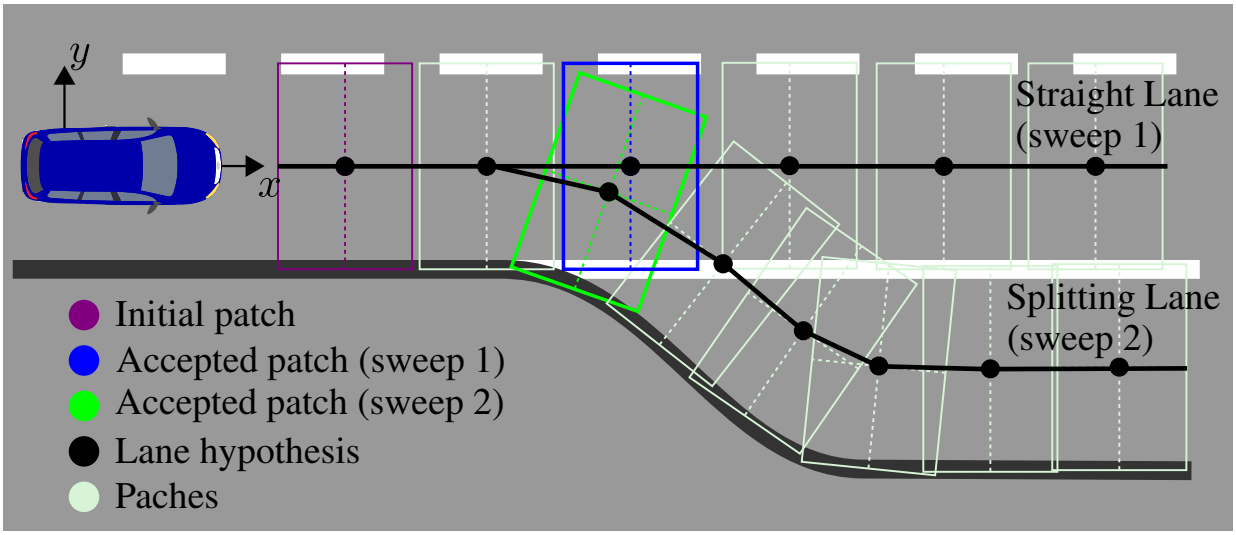


Figure 4.13: Illustration of two depth-first message passing sweeps for lane detection. For each lane segment a stochastic process (see Equation 4.20) is used to decide weather a patch is part of a lane-hypothesis or not. This process accounts for spatial uncertainties in e.g., splitting points and thus allows to detect different lanes in a single image frame.

accuracy of the results.

An example of depth-first message passing for lane detection is given in Figure 4.13, showing the results of two message passing sweeps. In this example, for both sweeps the initially patch that is selected according to Equation 4.19 is located close to the vehicle. Subsequently, lanes are detected following the message passing procedure shown in Figure 4.12. The lane detection results show two lane hypotheses which are detected in a single frame. Namely, a straight and a splitting lane. The reason why both lanes can be detected is the stochastic process for accepting patches given by Equation 4.20. This process, avoids making hard decision and hence it is likely that in one sweep a patch corresponding to the straight lane is accepted and in the next one a patch of the splitting lane.

The proposed message passing approach has several convenient properties. By using the selection criterion in Equation 4.19, it prefers hypotheses that are likely to be valid, and thus allows for a fast lane detection. Most importantly, however, samples are distributed according to the contextual information comprised in the CHM which has two main advantages. First, during the alignment process, inference can be limited to a finite area which reduces computational complexity. Second, it increases the recognition performance of the presented approach, since distributing samples based on contextual information allows to detect objects in areas of low belief. The latter is of particular importance, since in many situations the host lane can be detected with high confidence, while neighboring lanes are

harder to detect. In NBP this leads to the problem that many lane samples convert to the host lane, and thus many samples are needed to detect neighboring lanes leading to additional computational complexity. The proposed approach, however, overcomes this common issue of nonparametric approaches, as detailed next.

Road detection: An attractive property of depth-first message passing is that it allows to not only propose positions for subsequent lane elements, but also to propose entire lane hypotheses. An example of this process is illustrated in Figure 4.12 showing orange messages. As can be seen, as soon as messages are propagated up to the root, a message containing a single sample predicting the location of the neighboring lane can be constructed. Subsequently, a top down sweep can be performed to evaluate the likelihood of the proposed lane hypothesis, where at each level hypotheses are accepted according to Equation 4.20.

This procedure is comparable to importance sampling, since during the top down sweep, the importance weight of the lane sample is adjusted based on the sensory evidence. More precisely, the sample $\boldsymbol{s}_m^{(1)}$ predicting the neighboring lane is a sample drawn from the nonparametric belief estimate $b_m(\boldsymbol{x}_m^l)$ which equals the proposal function in this case. Further, the importance weight of $\boldsymbol{s}_m^{(1)}$ can easily be evaluated as in Equation 4.15, since the proposed CHM makes the high-level lane variable implicitly observable.

The ability to generate hypotheses based on the structure of the graphical model is a key benefit of the proposed framework, since in contrast to other nonparametric lane perception approaches [Dan09, Fra07] it allows to detect lanes in areas of low belief using a relatively small number of samples.

Nonetheless, depth-first message passing is inevitably a trade-off, in that a better approximation of some regions of the probability space results in a worse approximation of others. In depth-first message passing, the partial belief estimate is taken into account when deciding on where to distribute samples in the bottom up/top down sweep. Thus, potentially more of the computational resources is focused on relevant parts of the space. In fact, the experiential results of this thesis show that depth-first message passing allows to obtain reliable results while maintaining low computational complexity.

5 Hierarchical Lane and Road Perception for heterogeneous Road Scenes

This chapter presents a hierarchical approach for rapid lane and road perception in heterogeneous road scenarios that allows to account for the challenge of scenario diversity (see Section 1.2). The following sections are structured as follows: Section 5.1 details the addressed challenge, Section 5.2 specifies the set of features used in the experiments and Section 5.3 presents how a priori scene knowledge is used in the proposed framework as well as how similarities between different road types are used to facilitate inference. Finally, Section 5.4 presents the novel class of hybrid Compositional Hierarchical Model (CHM) which accounts for both spatial and classification uncertainties.

A crucial enabler for future Advanced Driver Assistance Systems (ADAS), is the development of a lane and road perception approach that can cope with the enormous diversity of target scenarios. In principal, there exist two main sources for scenario diversity. First, roads may vary in their topology, i.e., number, position and width of lanes, parallel merging, splitting and ending lanes and roads (cf. Definition 5). Second, lanes and roads may be defined by varying lane and road cues, such as different types of lane marking (e.g., dashed or continuous) or road edges (e.g., curbstone or guardrail) and thus differ in their appearance.

In order to account for varying road topologies, a representation of different road toplogies as a set of CHMs is proposed, where each CHM encodes a specific road topology. The approach brings the benefit of providing the possibility of adding an infinite number of CHMs to the presented framework and thus to make it applicable to arbitrary scenarios. Even though this is attractive by means of scalability it also leads to high computational complexity. In fact, performing inference in a set of hierarchies amounts in processing each CHM separately which is intractable due to the exponential growth of possible instances. However, it is reasonable to expect that many hierarchies in the set of CHM comprise common parts (see Figure 1.1). This allows to apply part-sharing, which has been proposed for efficient inference in the image domain and for 3d problems in [Zhu10] and [Spe11, Spe13], respectively.

Towards the goal of handling the appearance diversity, a novel class of hybrid CHMs is introduced, in which parts are represented on a hybrid (continu-

ous/discrete) state space and which allows to incorporate multiple low-level feature extraction approaches. As in the previous chapter, continuous variables are used to represent topological road properties, while the additional discrete variables encode additional road properties, such as if a lane is bounded by a dashed marking, a continuous marking or a curbstone. This approach is motivated by the observation that the topology and the appearance of a road are mostly independent (e.g., the topology of a lane is not influence by the type of a lane markings). The proposed hybrid CHM takes such independence assumptions into account and thus provides a computational efficient probabilistic approach for estimating both topological and appearance properties of lanes and roads.

Many recent works have addressed challenges related to scenario diversity. Similarly, to the proposed approach in [Apo03, Fra07, Dan09] multiple lane and road cues are used to support the task of lane and road perception. However, they mainly focus on detecting the host lanes and further do not provide the possibility of estimating additional properties, such as lane boundary types. More recently, Hur et al. [Hur13] proposed an approach towards multi-lane detection on roads with splitting, merging and parallel lanes. Even though their approach leads to promising results, it mainly relies on lane marking cues which limits its field of application. Further, it focuses on the detection of lane boundaries and thus only obtains limited information on the overall road topology.

5.1 Topology and Appearance of Roads

The diversity of appearance and topology of roads is a key challenge in lane and road perception, since it complicates the development of a generic lane and road perception approach [Hil12, McC06]. However, as roads are designed for the human driver they follow a certain structure. In fact, most roads are constructed according to standardized guidelines, such as [ras95]. These guidelines specify both the topology of roads for different environments (e.g., urban, rural and highway) and the appearance of visual cues used to support the human driver (e.g., different lane marking types). Figure 5.1 shows different road topologies according to [ras95] and the appearance diversity of a single road topology. It can be seen that the topology of roads differs depending on the environment. Highways, for example, comprise lanes with a width larger than the one of urban lanes to allow for fast driving. Not illustrated is that roads may comprise a varying number of lanes or even splitting and merging lanes.

Further, Figure 5.1 shows that even for a single road topology the appearance of the road may vary significantly. This is caused by the different cues used to define roads and lanes, such as continuous lane markings, dashed lane markings or curbstones. Especially, lane markings show a great diversity, since they are used to

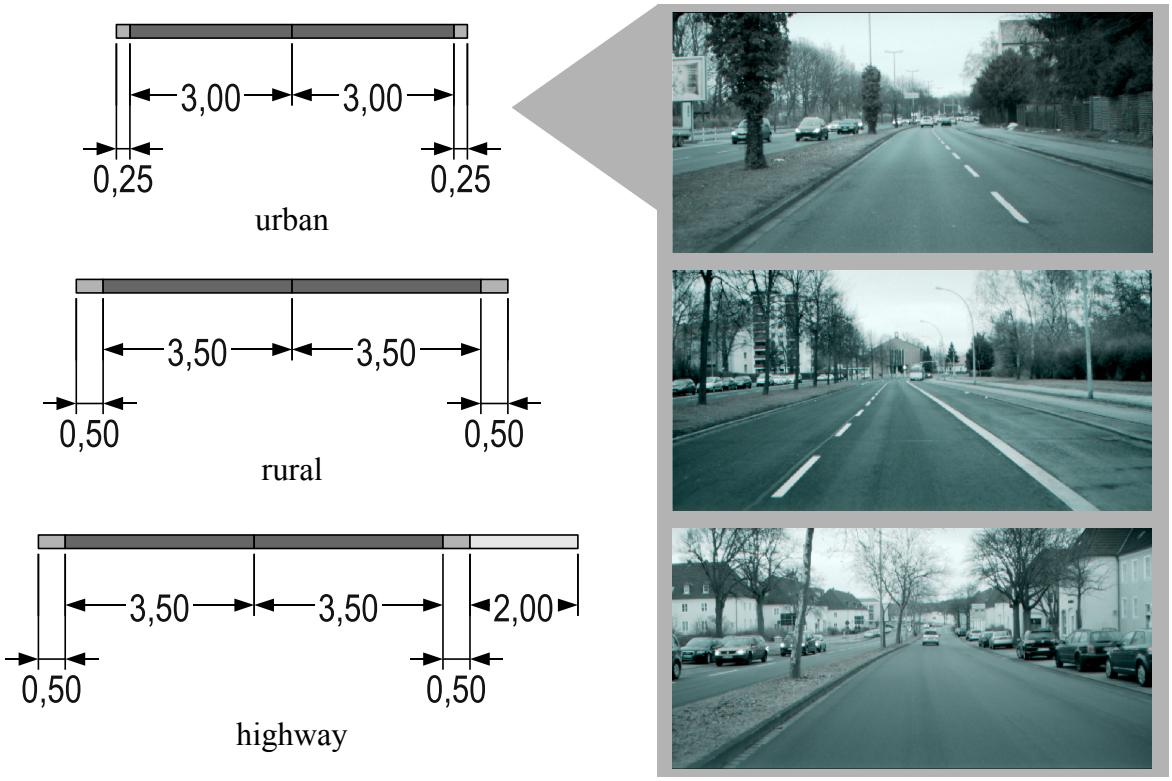


Figure 5.1: Examples for the diversity of road scenarios. (left) Road topologies as specified in [ras95] for urban roads, rural roads and highways with emergency lane. (right) Appearance diversity of the urban road topology.

encode traffic rules. For example, dashed lane markings commonly indicate that a lane change is allowed, while continuous lane markings prohibit a lane change. Finally, lane and road perception is complicated, since it is not guaranteed that a specific cue is reliable in all scenarios.

A lane and road perception system has to be able to cope with both the topological and the appearance diversity of roads. The following sections, present several extensions to the previously introduced framework to handle exactly the above challenges.

5.2 Multi-Cue Sensory Evidence

Lane and road perception in intelligent vehicles requires both, a framework that allows for efficient interpretation of low-level evidence and sensors that extract information from the vehicles environment. Since, there is no perfect low-level feature extraction approach, many recent lane and road perception systems rely on a combination of feature extraction approaches that are applied to the same input image [Fra07, Dan09, Gei11, Gei14, Spe11, Spe13]. Similarly, the proposed

approach relies on multiple low-level features obtained form a single frame, such as lane markings, road edges, stoplines and turn arrows. While the incorporation of turn arrows and stoplines in the proposed framework is detailed in Chapter 6, this chapter focuses on different kinds of lane and road boundary features.

Towards this goal, the two feature extraction approaches introduced in Section 4.1.1 are extended by an additional classification process which obtains the class of both lane marking and road edge features. Here, the feature classes or types *continuous narrow lane markings*, *continuous wide lane markings*, *dashed narrow lane markings*, *dashed wide lane markings* and *road edge* are distinguished.

Formally, all boundary feature extraction results are summarized as boundary features $\boldsymbol{b} = \{\boldsymbol{m}, \boldsymbol{r}\}$, where $\boldsymbol{m}$ comprises a set of N_m lane marking features $\boldsymbol{m} = \{\boldsymbol{m}_1, \ldots, \boldsymbol{m}_{N_m}\}$. Each lane marking feature $\boldsymbol{m} = (x_m, y_m, \vartheta_m, \boldsymbol{t}_b)$ is defined by its position $(x_m, y_m) \in \mathbb{R}^2$ and its orientation $\vartheta_m \in [0, 2\pi)$ in vehicle coordinate system. Furthermore, $\boldsymbol{t}_b$ denotes the discrete probability distribution over the different boundary feature classes, as discussed above. In addition, a set of N_r road edge features $\boldsymbol{r} = \{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{N_r}\}$ is detected, where in analogy to the marking features each road edge feature $\boldsymbol{r}$ is defined as $\boldsymbol{r} = (x_r, y_r, \vartheta_r, \boldsymbol{t}_b)$.

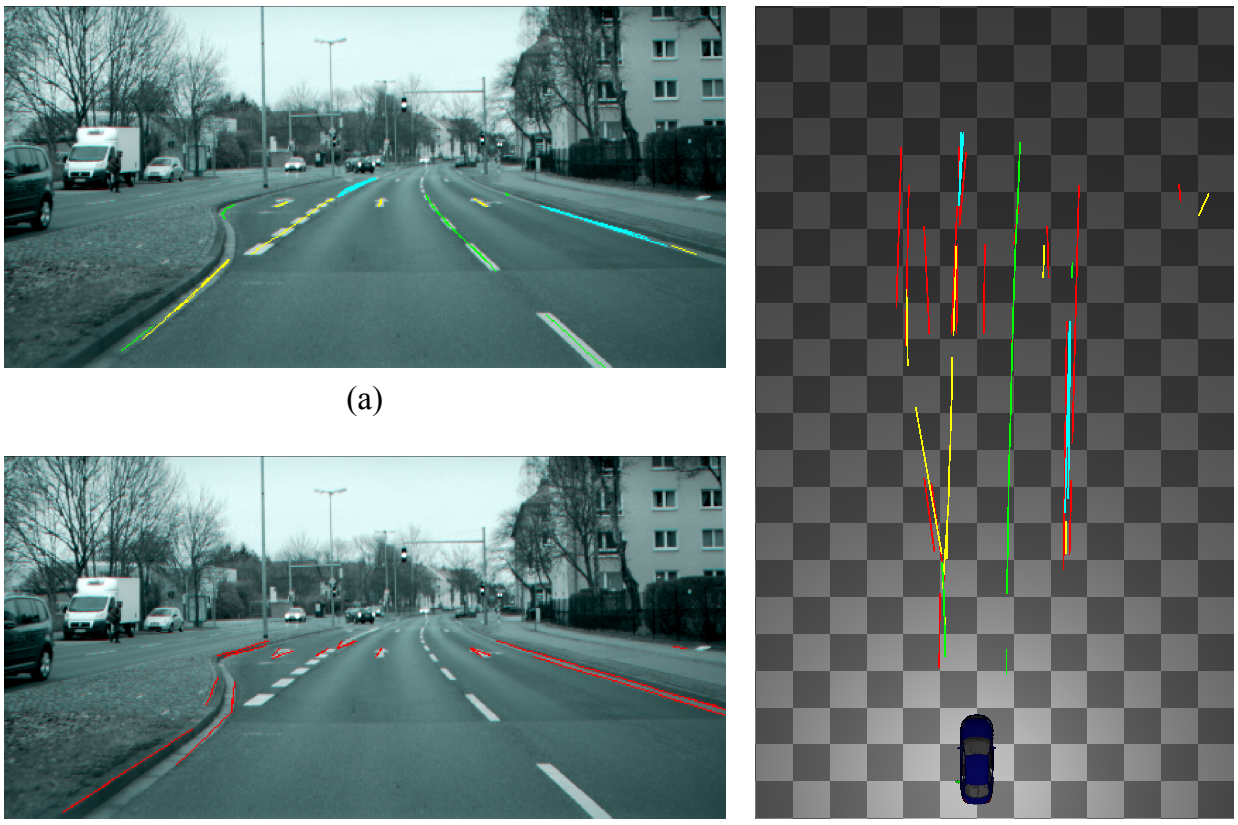
Sample results of applying these two feature extraction approaches to a single input image are shown in Figure 5.2. As expected, features obtained by the lane marking detector mostly correspond to lane markings, while the road edge detector extracts features from the curbstones. Nonetheless, Figure 5.2 also shows that this is not always the case. This observation is one of the motivations for the development of the hybrid compositional models, as proposed in Section 5.4.

5.3 Topological Diversity and Sets of Hierarchies

This section, details how the rich amount of a priori scene knowledge contained in road construction guidelines [ras95] can be incorporated into the proposed model to support the challenging task of lane and road perception in heterogeneous road scenes. Further, it details how diverse road topologies can be represented as a set of CHMs and how similarities between road topologies can be used to apply part-sharing [Zhu10, Spe11] for efficient inference.

5.3.1 Compositional Hierarchical Model of a Single Road

As detailed in Section 4.2, multi-lane roads are represented as a CHM that is encoded by an undirected tree-structured graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with edges $\mathcal{E}$ and nodes $\mathcal{V}$. The nodes $\mathcal{V}$ correspond to three disjoint sets of variables $\mathcal{V} = \boldsymbol{x} \cup \boldsymbol{m} \cup \boldsymbol{r}$. Here, hidden random variables $\boldsymbol{x} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N_x}\}$ define the different levels of



(a) (b) (c)

Figure 5.2: Low-level feature input for lane and road perception. (a) Dashed narrow lane marking features (green), dashed wide lane marking features (yellow) and continuous wide lane marking features (cyan). (b) Road edge features (red). (c) Projection into the vehicle coordinate system.

the CHM and represent a road on several abstraction levels, i.e., patches, lanes and roads (see Section 4.2). Furthermore, observable variables $\boldsymbol{m} = \{\boldsymbol{m}_1, \ldots, \boldsymbol{m}_{N_m}\}$ and $\boldsymbol{r} = \{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{N_r}\}$ correspond to lane marking observations and road edge observations, respectively.

Edges $\mathcal{E}$ encode either spatial relations $\psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ between pairs of hidden variables or observation potential $\phi_i(\boldsymbol{x}_i, \boldsymbol{m})$ and $\phi_i(\boldsymbol{x}_i, \boldsymbol{r})$, as defined in Equation 4.1. Note that, in this representation the results of the different feature extraction approaches are assumed to be independent.

The principal structure of a CHM is illustrated in Figure 5.3, showing the CHM of the highway with emergency lanes depicted in Figure 5.1. Note that, for illustrative simplicity the length of lanes or roads is not shown. In the proposed approach, such a road topology is considered as a specific target scenario which is represented by the root node of a CHM. The CHM defines the decomposition of the root object $\boldsymbol{x}_{11}^r$ in its parts and subparts. In the above example, $\boldsymbol{x}_{11}^r$ is decomposed into a two-lane road $\boldsymbol{x}_{10}^r$ and an emergency lane $\boldsymbol{x}_9^{p,l}$. The two-lane road $\boldsymbol{x}_{10}^r$ is

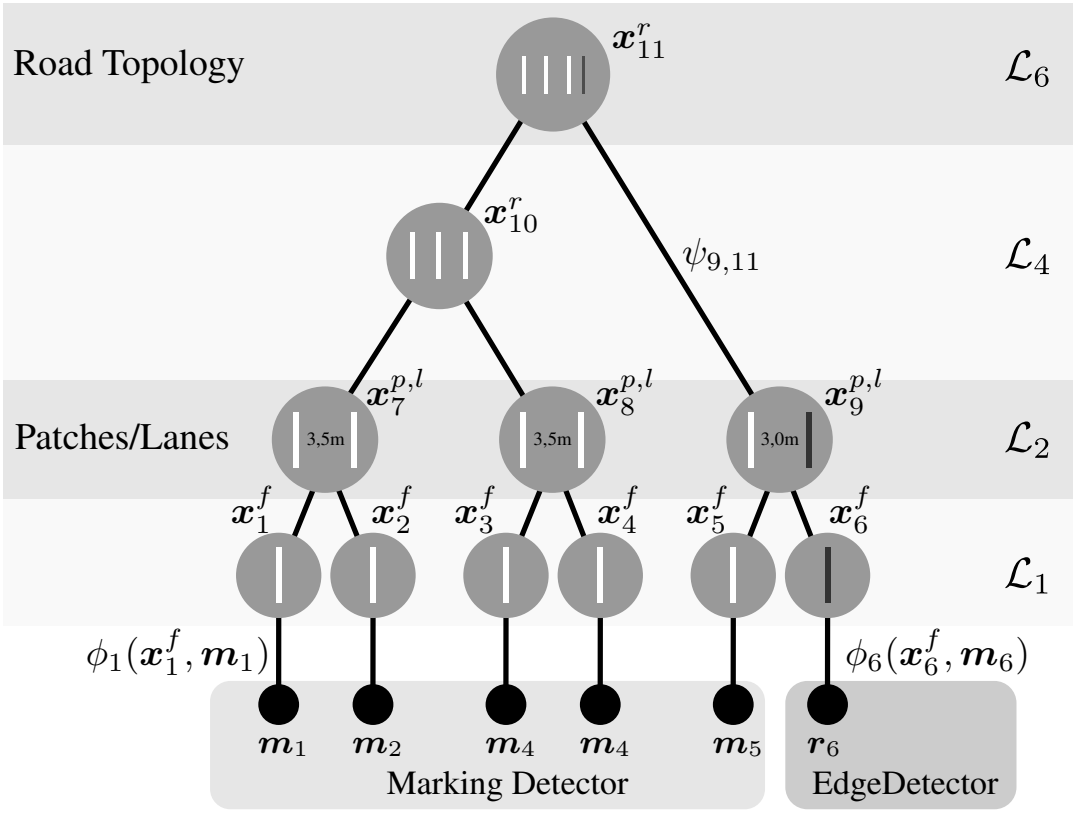


Figure 5.3: CHM of a single road scenario. The root node represents a specific target scenario and the CHM specifies its dependency to low-level observation $\boldsymbol{m}_i$ and $\boldsymbol{r}_i$. This model contains different part types to encode if e.g., a patch is composed of two lane marking features or a lane marking feature and a road edge feature.

further decomposed into its lanes $\boldsymbol{x}_7^{p,l}$ and $\boldsymbol{x}_8^{p,l}$. Finally, all lanes are decomposed into boundary features $\boldsymbol{x}_i^{f_b}$ that are directly observable.

The above example clarifies how a priori knowledge provided by construction guidelines [ras95] is used in the proposed framework. Namely, for each road topology defined in [ras95] a new CHM is specified, as shown in Figure 5.3. As a result, the spatial constraints of this CHM contain the topological a priori knowledge of the guideline.

For example, spatial relations $\psi_{i,j}(\boldsymbol{x}_i^f, \boldsymbol{x}_j^p)$ between patch and features contain a parameter ν_{w_p} which specifies the expected patch width (see Equation 4.4). In the proposed approach this parameter is specified as the lane width given in the roadway construction guideline. However, since there exist different lane widths, it may be necessary to specify multiple spatial constraints. This is illustrated in Figure 5.3, showing the value of ν_{w_p} in the nodes on level $\mathcal{L}_2$. Further, road guidelines specify the number of lanes and the lane structure (e.g., parallel, splitting or merging) of a road. In the proposed model these dependencies between neighbor-

ing lanes are encoded by spatial constraints between lane and road variables.

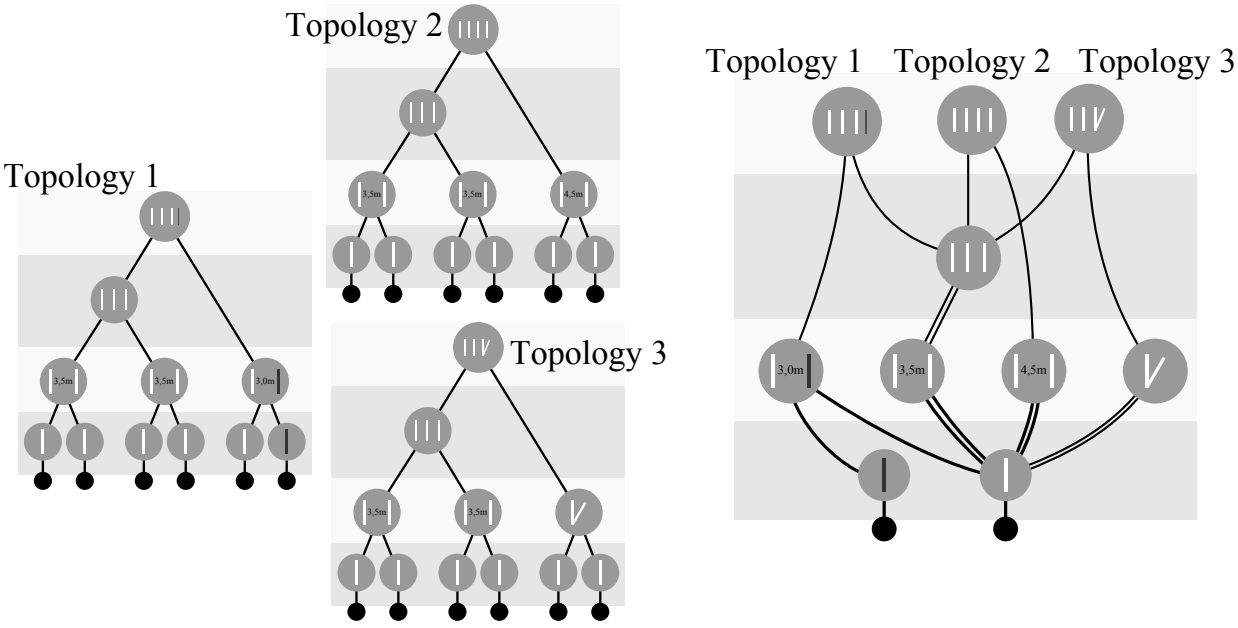
Even though, the knowledge provided by construction guidelines can not be expected to be exact, it allows to bound hypotheses to plausible domains. This is not only computational attractive, but also allows to handle outlier, due to clutter or occlusions, as detailed in Section 4.3.1.

Figure 5.3 also shows how distinct feature variables, shown by black and white symbols, and corresponding observation potentials $\phi_i(\boldsymbol{x}_i, \boldsymbol{m}_i)$ and $\phi_i(\boldsymbol{x}_i, \boldsymbol{r}_i)$ can be used to incorporate different low-level feature extraction approaches. Introducing distinct feature variables is the most direct approach to incorporate multiple cues in the perception process. This approach, however, has two main disadvantages. First, if a variety of feature types (e.g., dashed marking, continuous marking and curbstone) has to be modeled, an enormous number of parts has to be specified, due to the many possible compositions on the higher levels of the CHM. This leads to very complex models, where only a subset of parts is instanced at a time. Second, by introducing distinct variables, the different feature types are assumed to be independent which is often not the case (see Section 5.2). To overcome these issues, this thesis proposes to represent the variables of the presented model on a hybrid (continuous/discrete) state space. This leads to the hybrid CHMs presented in Section 5.4, where continuous variables specify the configuration of a part and discrete variables represent discrete probability distributions over its type (e.g., the probability of a patch being bounded by a dashed lane marking and a road edge or two dashed lane markings).

5.3.2 Multi-Scenario Representation

So far, only the task of formulating a graphical model for a single road topology was considered. However, as discussed in Section 5.1 road scenarios show a great topological diversity. While, in general, the spatial constraints of the proposed CHM allow for a certain degree of spatial variation, as soon as these variations become too large, or different road topologies have to be represented new separate CHMs have to be specified. Therefore, heterogeneous road topologies are not represented by a single CHM but by a set of N_h CHMs $\boldsymbol{\mathcal{G}} = \{\mathcal{G}_1, \ldots, \mathcal{G}_{N_h}\}$, where each CHM is a joint probability distribution defined over a hierarchical graph $\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E}_i) \subset \boldsymbol{\mathcal{G}}$.

An illustrative example of a set of CHM is depicted in Figure 5.4, showing the compositional hierarchies of different road topologies. At the first glance, providing a separate CHM for each road topology seams unattractive, since it leads to an exponential growth of instance. However, it is reasonable to expect that many CHM contain similar parts (see Figure 1.1). In fact, the three CHMs in Figure 5.4 comprise several common parts which allows to apply part-sharing for efficient



(a) set of hierarchies (b) sharing structure

Figure 5.4: Representation of heterogeneous road topologies using sets of CHMs and part-sharing. (a) Set of CHMs, where each CHM models a specific road topology. (b) Many road topologies comprise common parts and thus can be represented by means of shared parts using a sharing structure. Performing bottom up inference in this sharing structure is computational efficient and allows to simultaneously estimate multiple road topologies.

inference.

5.3.3 Similarities and Part-Sharing

Building on the part-sharing approach introduced in Section 4.3.2, it can be extended to account for sharing common parts between different road topologies. Let us for example consider the set of hierarchies depicted in Figure 5.4. One option to perform inference in this model is to use the inference algorithm presented in Section 4.3 and apply it to each CHM contained in the set of CHMs. This approach, however, is computational unattractive, since it ignores the fact that many CHMs comprise similar parts. A better strategy is, therefore, to represent the set of CHMs by means of common parts, as depicted in Figure 5.4. The shown sharing structure clarifies the similarities between the different road topologies, showing how e.g., the two-lane road can be shared between the three different road topologies. Note that, sharing the two-lane road also requires to share its sub-tree.

The key benefit of part-sharing is that during bottom up message passing, the bottom up belief over all common parts has to be computed only once and then it can be shared between its parents. Hence, as in Section 4.3.2, bottom up message

passing is performed in the sharing structure and top down message passing in the individual CHMs, corresponding to the root nodes of the sharing structure.

Thus, part-sharing is one of the key aspects in the proposed framework, since it allows not only to perform efficient bottom up inference in a single CHM (see Section 4.3.2), but also to share complex high-level objects between different road topologies. In contrast to many recent lane and road perception approaches which only aim to detect a single road topology (see Section 2.3) this guarantees that the presented approach is scaleable to various topologies. And most important, the estimation of all road topologies in the set is supported by the rich amount of a priori knowledge contained in the construction guidelines.

5.4 Appearance Diversity and Hybrid Compositional Hierarchical Models

The hierarchical framework, developed so far already allows to estimate heterogeneous topologies and to incorporate multiple feature detection approaches. Unfortunately, the proposed model does not scale well if both the topology of roads and their appearance has to be estimated. In particular, it does not account for uncertainties arising during feature classification, as discussed in Section 5.3.1.

In order to overcome these issues, this section proposes a novel representation of lanes and roads using hybrid CHM. In this extended representation, the configuration $\boldsymbol{x}_i$ of each node is augmented with a hidden discrete variable $\boldsymbol{d}_i$ that specifies the type of a part (e.g., if a feature is a continuous lane marking, a dashed lane marking, a curbstone, etc.). Intuitively, this hybrid part representation can be understood as a form of sharing, where the configuration of a part is shared between different discrete types. This allows to not only attain a low model complexity when reasoning about various part types, but also provides an intuitive way of representing classification uncertainties. In fact, in the proposed hybrid CHM the classification uncertainties are directly represented as probability distribution $p(\boldsymbol{d}_i|\boldsymbol{b})$ over the discrete states of each part.

The following sections are structured as follows. Section 5.4.1 details the proposed hybrid CHM, including the incorporation of multiple feature sources, the specification of model parts and a discussion of its factorization properties. Finally, Section 5.4.2, presents a version of Nonparametric Belief Propagation (NBP) for efficient inference in hybrid CHMs.

5.4.1 Hybrid Compositional Hierarchical Models

A hybrid CHM $\mathcal{G}(\mathcal{E}, \mathcal{V})$ specifies the factorization of a probability distribution p over a set of variables $\mathcal{V} = \boldsymbol{x} \cup \boldsymbol{d} \cup \boldsymbol{b}$, where $\boldsymbol{x}$, $\boldsymbol{d}$ and $\boldsymbol{b}$ are disjoint. Here $\boldsymbol{x}$ denotes the continuous variables and $\boldsymbol{d}$ the discrete variables. As before a hidden variable $\boldsymbol{x}_i \subseteq \boldsymbol{x}$ defines the spatial configuration of a part on a continuous state space (see Section 4.2). Novel are the hidden discrete variables $\boldsymbol{d}_i \subseteq \boldsymbol{d}$ that are defined on the state space $\mathrm{Val}(\boldsymbol{d}_i) = \{d_{i,1} \ldots, d_{i,N_d}\}$, where each instance $d_{i,n} \in \mathrm{Val}(\boldsymbol{d}_i)$ represents a specific part type. Therefore, each part $\boldsymbol{z}_i$ is defined on a hybrid state space $\boldsymbol{z}_i = (\boldsymbol{d}_i, \boldsymbol{x}_i)$. Note that, this means, that a discrete variable $\boldsymbol{d}_i$ only exists if a sample in the continuous space is present.

Further, the previous assumptions that the features obtained by the feature detectors described in Section 5.2 are independent is dropped. Hence, the observable variables $\boldsymbol{b} = \boldsymbol{m} \cup \boldsymbol{r}$ comprise both the marking detections and the road edge detection. In analogy to the feature representation, each observable variable $\boldsymbol{b} = (x_b, y_b, \vartheta_b, \boldsymbol{c}_b)$ is defined by its location $(x_b, y_b) \in \mathbb{R}^2$, orientation $\vartheta_b \in [0; 2\pi)$ and an discrete variable $\boldsymbol{t}_b$ that denotes the discrete probability distribution over the feature classes (see Section 5.2).

The edges $\mathcal{E}$ of the graphical model encode the probabilistic dependencies. However, before detailing the factorization properties of the hybrid CHM its hybrid parts are specified.

5.4.1.1 Hybrid Model Parts

The proposed hybrid CHM comprises feature, patch, lane and road variables, each defined by a hybrid variable $\boldsymbol{z}_i = (\boldsymbol{x}_i, \boldsymbol{d}_i)$. While, as discussed above, the continuous variable $\boldsymbol{x}_i$ of each part specifies its spatial configuration, the discrete variable $\boldsymbol{d}_i$ comprise different part types which encode additional road properties.

Definition 6 (Hybrid Part)**.** *Each part $\boldsymbol{z}_i = (\boldsymbol{x}_i, \boldsymbol{d}_i)$ in the hybrid CHM comprises a discrete variable $\boldsymbol{d}_i$, whose values $d_{i,j} \in \mathrm{Val}(\boldsymbol{d}_i)$ correspond to the different lane boundary types. Furthermore, the continues variable $\boldsymbol{x}_i$ defines the spatial configuration of the part.*

For example, a boundary feature variable $\boldsymbol{z}_i^f = (\boldsymbol{x}_i^f, \boldsymbol{d}_i^f)$ in the hybrid model is specified by a continuous variable $\boldsymbol{x}_i^f = (x_i, y_i, \theta_i)$ defining its position $(x_i, y_i) \in \mathbb{R}^2$ and orientation $\theta_i \in [0, 2\pi)$ in the vehicles coordinate system. Further, $\boldsymbol{d}_i^f$ is a discrete variable defining a probability distribution over the five boundary feature classes *continuous narrow lane markings*, *continuous wide lane markings*, *dashed narrow lane markings*, *dashed wide lane markings* and *road edge*. This hybrid feature representation is illustrated in Figure 5.5. Further, this figure illustrates

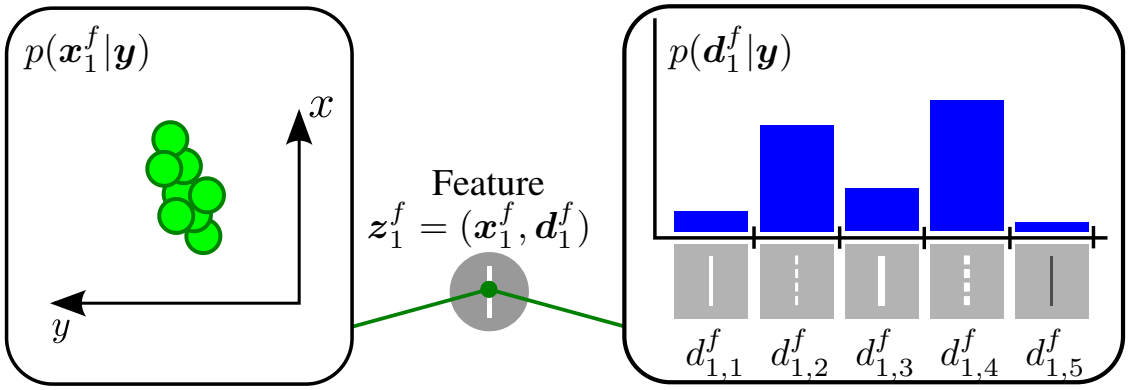


Figure 5.5: Example of a hybrid feature representation. A feature is specified by a continuous variable $\boldsymbol{x}_1^f$ and a discrete variable $\boldsymbol{d}_1^f$ which is defined on the state space $\mathrm{Val}(\boldsymbol{d}_1^f) = \{d_{1,1}^f, d_{1,2}^f, d_{1,3}^f, d_{1,4}^f, d_{1,5}^f\}$. The belief of the continuous states is approximated by a set of samples, shown by green dots. The belief over the discrete variable encodes the probability of the different feature classes, i.e., continuous narrow lane marking, dashed narrow lane marking, continuous wide lane marking, dashed wide lane marking or road edge.

that a probability distribution over the instances of the discrete variable $\boldsymbol{d}_i^f$ directly represents the uncertainty about the class of the feature.

Similarly, the other parts in the hybrid CHM comprise a multidimensional continuous variable defining their spatial configuration, as specified in Section 4.2 and a discrete variable labeling their boundary types. An overview on the different types of features, patches, lanes and roads considered in the proposed approach is given in Figure 5.6. As can be seen, the discrete variables of patches, lanes and roads encode plausible combinations of lane boundary types. Thus, each of these types incorporates a priori knowledge into the model. Comparable to the topological a priori knowledge comprised in the model, this a priori knowledge allows to restrict the estimated part types to plausible domains and thus to cope with classification uncertainties arising during low-level feature extraction.

An important assumption underlying the part types shown in Figure 5.6 is that there exists a correlation between the lane boundary type on multi-lane roads. Nonetheless, one may find that this correlation is weak, i.e., the left lane boundary type is independent of the right lane boundary type. In this case, each lane boundary can be represented using separate discrete variables. Another important restriction is that the boundary types of lanes and patches are assumed to be equivalent (see Figure 5.5), i.e., boundary type transitions are not accounted for. However, in certain scenarios such transitions may provide important information on the scene. For example, bike-lanes are often on the sidewalk and thus may be defined by a curbstone on one side and some sort of texture transition on the

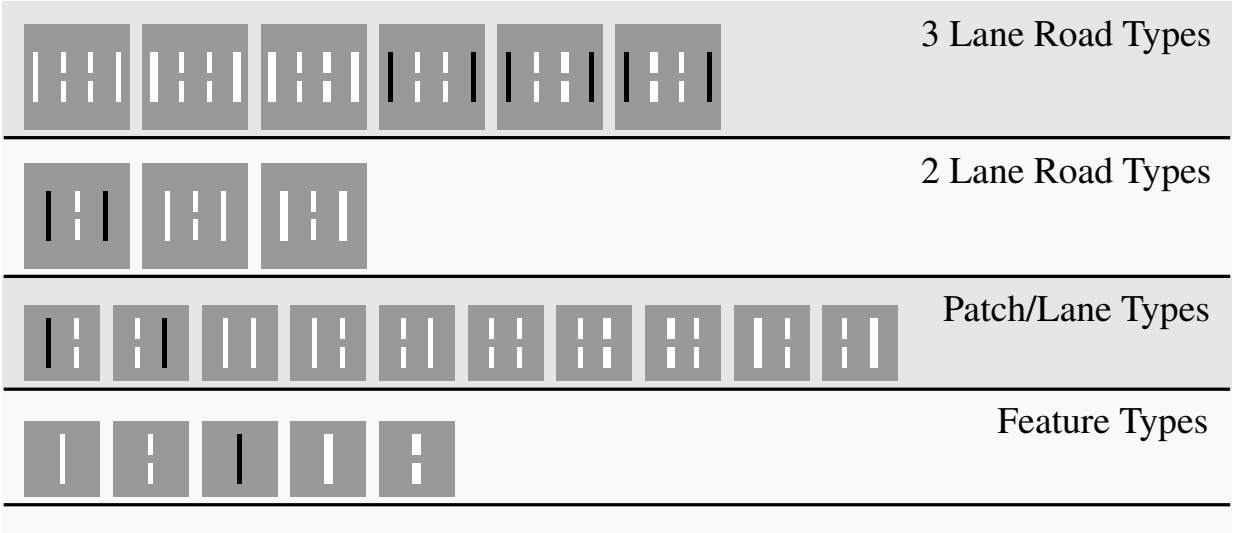


Figure 5.6: Discrete instances of roads, lanes/patches and features. The feature types continuous narrow, dashed narrow, continuous wide, dashed wide lane markings and road edge are computed during the feature extraction. The types of patches, lanes and roads are estimated during inference.

other side. However, in intersections they commonly cross the roads where they are defined by lane markings. A straight forward extension to the proposed framework would be to introduce lane boundary types which model such transitions to identify areas where vehicles and cyclists may intersect.

5.4.1.2 Observable Variables

To estimate the continuous and discrete properties of multi-lane roads, it is essential to evaluate how well the estimated results explain the feature extraction results. Towards this goal, observation potential $\phi_i(\boldsymbol{z}_i, \boldsymbol{b})$ are formulated assuming a uniform prior on all boundary types $\mathrm{Val}(\boldsymbol{t}_b)$

$$\boldsymbol{d} \sim \mathcal{U}(\{1, \ldots, |\mathrm{Val}(\boldsymbol{t}_\mathrm{b})|\}). \tag{5.1}$$

Further, assuming that the feature class is independent of the spatial feature configuration, the observation potential can be written as

$$\begin{aligned}\phi_i(\boldsymbol{d}_i, \boldsymbol{x}_i, \boldsymbol{b}_i) \propto \big(&\lambda^0 \mathcal{N}_0(\boldsymbol{x}_i; 0, \boldsymbol{\Sigma}_0) \\ &+ (1-\lambda^0) \sum_{k=1}^{N_f} \mathcal{N}(\boldsymbol{x}_i; \mu_{i,k}, \boldsymbol{\Sigma}_{i,k})\big) \sum_{f \in \Omega(x_i)} p_{d_i}(f).\end{aligned} \tag{5.2}$$

Here, $\Omega(x_i)$ denotes the area around an estimated feature configuration $\boldsymbol{x}_i$. Thus, $\sum_{f \in \Omega(x_i)} p_{d_i}(f)$ is the probability of observing a specific feature type d_i in the

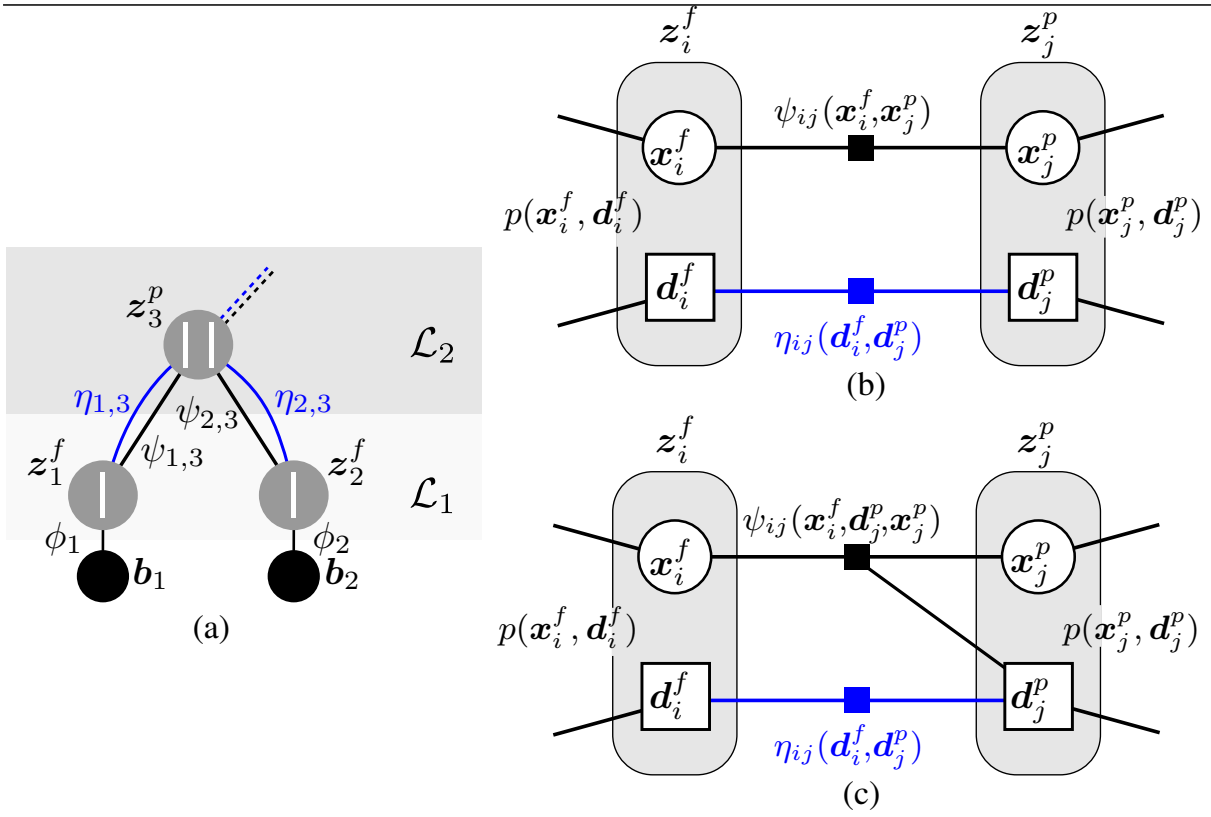


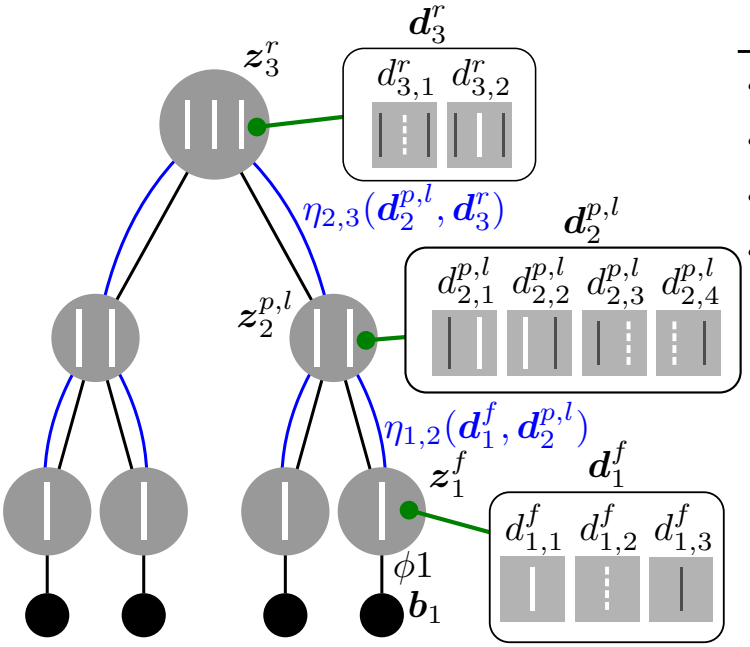
Figure 5.7: Independence structure of a hybrid CHM. (a) Hybrid CHM of a patch. (b) Factor graph showing the factorization based on the assumption that the types of neighboring parts have no influence on their spatial dependency. (c) Dropping this assumption leads to a different factorization that is used in Section 6.2.1.

area $\Omega(x_i)$, where the corresponding feature detection results f are assumed to be independent. The reason, for computing the probability of a feature type for a given area $\Omega(x_i)$ is that it reduces ambiguities regarding the feature type when using multiple feature detectors.

5.4.1.3 Factorization

Following the above discussion, the proposed hybrid CHM can be thought of as having two disjoint layers with the same hierarchical structure. Namely, one layer containing only discrete variables $\boldsymbol{d}$ and one layer containing only continuous variables $\boldsymbol{x}$.

Consider the simplified hybrid CHM of a patch shown in Figure 5.7a, where features and a patch are specified as hybrid parts $\boldsymbol{z}_i^f = (\boldsymbol{d}_i^f, \boldsymbol{x}_i^f)$ and $\boldsymbol{z}_i^p = (\boldsymbol{d}_i^p, \boldsymbol{x}_i^p)$, respectively. As before, the patch is defined by a left and a right feature. Further, the discrete variables of both patches and features may comprise different types, as shown in Figure 5.6. The actual factorization underlying this model is illustrated in Figure 5.7b. This factor graph shows the two variables $\boldsymbol{x}_i^f$ and $\boldsymbol{d}_i^f$ defining



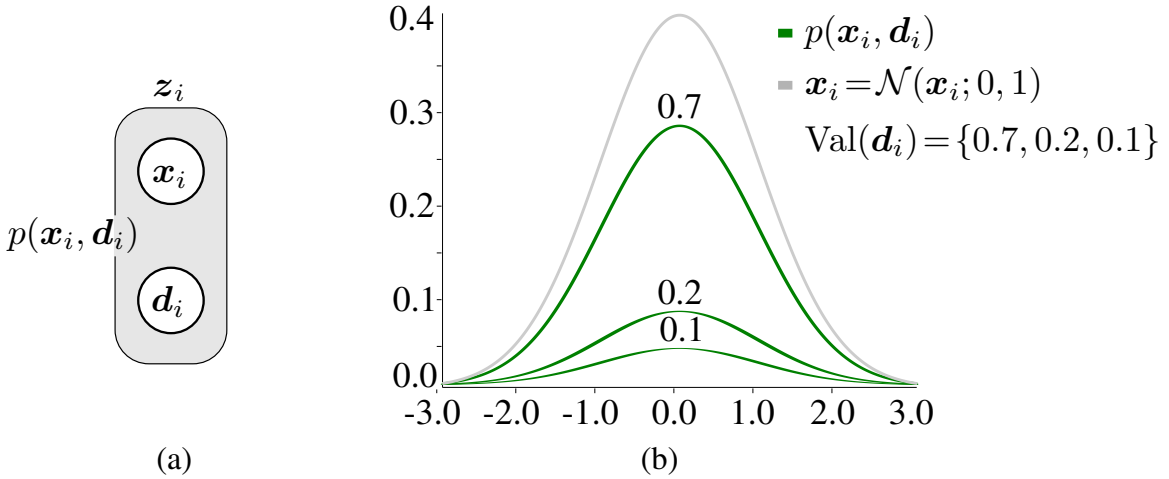
	$\boldsymbol{d}_1^f = d_{1,1}^f$	$\boldsymbol{d}_1^f = d_{1,2}^f$	$\boldsymbol{d}_1^f = d_{1,3}^f$
$\boldsymbol{d}_2^{p,l} = d_{2,1}^{p,l}$	1	0	0
$\boldsymbol{d}_2^{p,l} = d_{2,2}^{p,l}$	0	0	0,5
$\boldsymbol{d}_2^{p,l} = d_{2,3}^{p,l}$	0	1	0
$\boldsymbol{d}_2^{p,l} = d_{2,4}^{p,l}$	0	0	0,5

CPT of $\eta_{1,2}(\boldsymbol{d}_1^f, \boldsymbol{d}_2^{p,l})$

Figure 5.8: Formulation of discrete state transitions model. The probabilistic dependencies between neighboring discrete variables $\boldsymbol{d}_i$ and $\boldsymbol{d}_j$ are encoded by discrete state transitions models $\eta_{i,j}(\boldsymbol{d}_i, \boldsymbol{d}_j)$. These are specified using conditional probability tables as shown in the top right.

a feature $\boldsymbol{z}_i^f$ as well as a patch $\boldsymbol{z}_i^p = (\boldsymbol{d}_i^p, \boldsymbol{x}_i^p)$. The factors in the model express two types of probabilistic dependencies. First, factors between neighboring continuous variables express the spatial constraints $\psi_{ji}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ specified in Section 4.2. Second, factors between neighboring discrete variables represent discrete state transitions models $\eta_{ji}(\boldsymbol{d}_i, \boldsymbol{d}_j)$. An example for such a discrete state transition model is given in Figure 5.8. This figure shows the hybrid CHM of a two-lane road $\boldsymbol{z}_3^r$ with two types $d_{3,1}^r$ and $d_{3,2}^r$, defining the state space of the discrete variable $\text{Val}(\boldsymbol{d}_3^r) = \{d_{3,1}^r, d_{3,2}^r\}$. The types of all model parts are illustrated by showing boxes comprising the discrete states. As can be seen, the feature nodes $\boldsymbol{z}_i^f$ comprises three discrete states. Namely, continuous lane marking, dashed lane marking and road edge (e.g., curbstone). The patch/lane nodes $\boldsymbol{z}_i^{p,l}$ comprise four discrete states which encode plausible constellations of lane boundaries (e.g., road edge left and dashed lane marking right). The dependencies between the discrete feature and patch states are encoded by the discrete transition model $\eta_{1,2}(\boldsymbol{d}_1, \boldsymbol{d}_2)$, for which an exemplary Conditional Probability Table (CPT) is shown in Figure 5.8 on the right.

Finally, assuming that discrete variables and the continuous variables are independent, the joint probability distribution $p(\boldsymbol{d}_i, \boldsymbol{x}_i)$ over a hybrid part is given by



(a) (b)

Figure 5.9: Probability distribution of a hybrid part. (a) Factor graph of a hybrid variable $\boldsymbol{z}_i = (\boldsymbol{d}_i, \boldsymbol{x}_i)$. (b) Joint probability distribution over a hybrid variable, where the distribution of the continuous variable $\boldsymbol{x}_i$ is Gaussian (grey). Each instances of the discrete variable $\boldsymbol{d}_i$ specifies a mixture component (green).

$$p(\boldsymbol{x}_i, \boldsymbol{d}_i) = p(\boldsymbol{x}_i)p(\boldsymbol{d}_i) = \sum_{k=1}^{L} \pi_i^{(k)} \mathcal{N}(\boldsymbol{x}_i; \boldsymbol{s}_i^{(k)}, \boldsymbol{\Lambda}_i) p_i(\boldsymbol{d}_i). \tag{5.3}$$

This is illustrate in Figure 5.9b, showing a probability distribution over $\boldsymbol{z}_i = (\boldsymbol{x}_i, \boldsymbol{d}_i)$. Here the belief over the continuous variable $\boldsymbol{x}_i$ is assumed to be a single Gaussian. As can be seen, each instance of the discrete variable $\boldsymbol{d}_i$ defines an individual mixture component. The weight of the mixture component is the probability of that instance. Intuitively, this means that the belief of a part $\boldsymbol{z}_i$ encodes not only the spatial distribution of the part, but also encodes the probability of a specific type at this location (see Figure 5.5). Note that, in the proposed model the discrete instances only have an influence on the mixture weights, not on the variance or the mean of the continuous distributions, as it is often the case in hybrid Bayesian networks [Kol99, Kol09, Rus10].

Given the above formulation, the joint probability distribution encoded by the hybrid CHM is given by

$$\begin{aligned} p(\boldsymbol{d}, \boldsymbol{x}|\boldsymbol{b}) \propto & \prod_{i \in \mathcal{I}_D} \phi_i(\boldsymbol{d}_i, \boldsymbol{b}_i) \prod_{i \in \mathcal{I}_X} \phi_i(\boldsymbol{x}_i, \boldsymbol{b}_i) \\ & \prod_{(i,j) \in \mathcal{E}_x} \psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j) \prod_{(i,j) \in \mathcal{E}_d} \eta_{i,j}(\boldsymbol{d}_i, \boldsymbol{d}_j), \end{aligned} \tag{5.4}$$

where $\mathcal{I}_D$ denotes the indexes of the set of cliques that are contained in $\boldsymbol{d} \cup \boldsymbol{b}$ and $\mathcal{I}_X$ the indexes of the cliques in $\boldsymbol{x} \cup \boldsymbol{b}$. Further, $\mathcal{E}_x$ and $\mathcal{E}_d$ are the sets of edges between continuous variables and discrete variables, respectively.

Assuming that all dependencies and the sensory evidence are Gaussian, Equation 5.4 clarifies that the proposed hybrid CHM induces a joint probability distribution that has the form of a mixture of Gaussian. The mixture contains one Gaussian component for each instance of the discrete variables.

The above discussion makes the important simplifying assumptions that the types of neighboring parts have no influence on their relative spatial configuration. However, this may not always be the case. In fact, Chapter 6 aims to fuse the set of lane boundary features with lane center cues, such as stopline and turn arrows towards a more holistic lane and road perception. In this case, the orientation of turn arrows and lane boundaries can be assumed to be the same, i.e., turn arrows and lane boundaries are parallel. Stoplines and lane boundaries, on the other hand, are orthogonal to each other, forming a u-shape. Hence, the spatial relation of e.g., patches and lane center cues depends on their discrete type (e.g., stopline or arrow). This, however, induces a different factorization, where the spatial constraints depend on the types of neighboring parts. This is depicted in Figure 5.7c, showing factors $\psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{d}_j)$.

Another key benefit of the proposed hybrid CHM is its rich expressive power. In fact, the number of discrete states can be chosen to be arbitrary large and thus additional feature types can easily be introduced, such as e.g., guardrails, guiding post, etc. This convenient property of the proposed hybrid models, is the key towards introducing the more involved models that allow for the estimation of lane turn directions and stopline locations in Chapter 6.

5.4.2 Inference in Hybrid Compositional Hierarchical Models

So far, inference was concerned with inferring the topology of lanes and roads using models containing only continuous variables. This section discusses inference in hybrid models, those that include both continuous and discrete variables. The goal during inference is to compute the marginal posterior distribution $p(\boldsymbol{x}_i, \boldsymbol{d}_i|\boldsymbol{b})$ or belief $b_i(\boldsymbol{x}_i, \boldsymbol{d}_i)$ of the hidden variables $\boldsymbol{z}_i = (\boldsymbol{x}_i, \boldsymbol{d}_i)$ in the model. A task that can efficiently be performed using belief propagation [Wei01, Yed01].

In belief propagation the belief over the individual variables is computed by fusing the incoming messages from its neighbors. In the hybrid model these message can be computed as

$$m_{j,i}(\boldsymbol{z}_i) = \sum_{\boldsymbol{d}_j} \int_{\boldsymbol{x}_j} \psi_{i,j}(\boldsymbol{z}_i, \boldsymbol{z}_j)\phi_i(\boldsymbol{z}_i, \boldsymbol{b}) \prod_{k \in \Upsilon(j)\setminus i} m_{k,j}(\boldsymbol{z}_j) d\boldsymbol{x}_j, \tag{5.5}$$

where the integral and the sum marginalize over the continuous variables $\boldsymbol{x}_j$ and discrete variables $\boldsymbol{d}_j$, respectively.

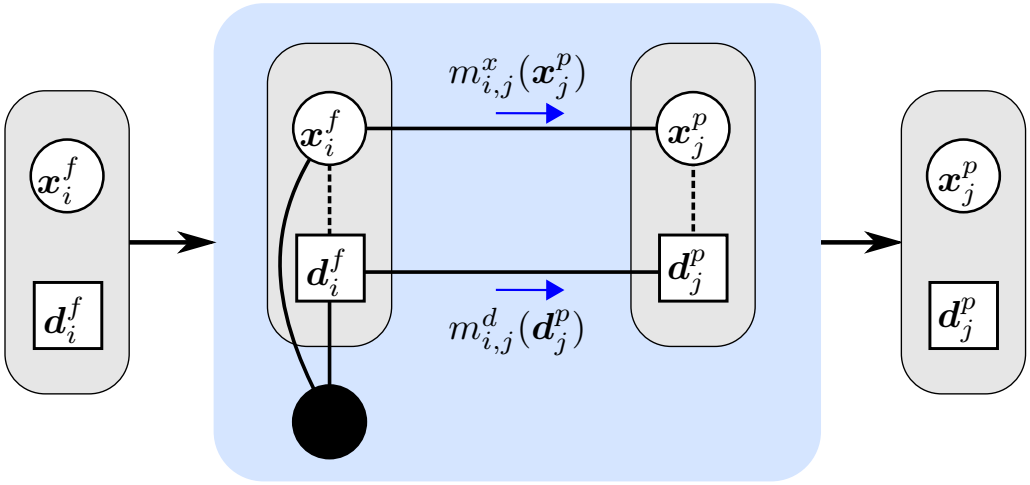


Figure 5.10: Messages passing in hybrid CHMs. Message required for performing the belief update are approximated by messages $m^x_{i,j}(\boldsymbol{x}^p_j)$ and $m^d_{i,j}(\boldsymbol{d}^p_j)$ using a factorization that decouples discrete and continuous variables.

The messages required for the belief estimation are approximated using the factorization illustrated in Figure 5.7b which decouples the discrete variables $\boldsymbol{d}_i$ and continuous variables $\boldsymbol{x}_i$. This leads to a structured approximation of the inference problem, where the message in Equation 5.5 is replaced by two separate messages

$$m^x_{j,i}(\boldsymbol{x}_i) = \int_{\boldsymbol{x}_j} \psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j)\phi_j(\boldsymbol{x}_j, \boldsymbol{b}) \prod_{k \in \Upsilon(j)\setminus i} m^x_{k,j}(\boldsymbol{x}_j) d\boldsymbol{x}_j \tag{5.6}$$

and

$$m^d_{j,i}(\boldsymbol{d}_i) = \sum_{\boldsymbol{d}_j} \eta_{i,j}(\boldsymbol{d}_i, \boldsymbol{d}_j)\phi_j(\boldsymbol{d}_j, \boldsymbol{b}) \prod_{k \in \Upsilon(j)\setminus i} m^d_{k,j}(\boldsymbol{d}_j) \tag{5.7}$$

corresponding to the factor $\psi_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and $\eta_{i,j}(\boldsymbol{d}_i, \boldsymbol{d}_j)$, respectively. The resulting message passing algorithm is illustrated in Figure 5.10. In this example, the messages $m^x_{i,j}(\boldsymbol{x}^p_j)$, predicts the spatial configuration of the patch variable based on the spatial constraint and the evidence over the continuous feature variables. Further, the messages $m^d_{i,j}(\boldsymbol{d}^p_j)$ contains the belief estimate over the discrete patch variable which is computed by multiplying the sensory evidence with the discrete state transition. This means, that the algorithm maintains independent distributions over $\boldsymbol{d}^p_j$ and $\boldsymbol{x}^p_j$. Thus, the belief $b_i(\boldsymbol{d}^p_j, \boldsymbol{x}^p_j)$ is essentially a product of a discrete distribution $b_i(\boldsymbol{d}^p_j)$ and a continuous distribution $b_i(\boldsymbol{x}^p_j)$.

A convenient property of the above algorithm is that the belief updates for discrete and continuous variables are computed separately. Hence, an exact belief update can be performed for the discrete variables (see Section 3.4.1) and

NBP [Sud03, Isa03] can be used to approximate the belief state of the continuous variables.

5.4.3 Relation to other models

The proposed hybrid CHM of roads is inspired by the rich amount of works using hybrid probabilistic models. Popular application, of hybrid models are e.g., switching linear dynamical system [Isa98b, Bar06, Bar12a] and mobile robot localization [Avo02]. Further, hybrid particle filtering is applied in [Ver4a, Ver04] to detect system faults during robot navigation and for applications of object tracking [Isa98b, BS04]. However, in contrast to their work, the proposed approach uses a structural approximation that allows to perform an exact belief update of the discrete variables and therefore only have to draw samples from the continuous variables.

Other works employ hybrid models and approximate inference for human pose estimation [Sig08, Fan05, Sud04a], where continuous state variables are augmented with discrete variables in order to handle occlusions. However, the goal in these approaches is not to estimate a distribution over the discrete variables, but to allow for the reliable estimation of self occluding object configurations.

More recently, a new class of part-based models using mixtures of parts has been proposed in the context of human pose estimation [Yan11, Dua12, Zhu12] which allow the appearance model of the human to change discretely. Similarly, to the proposed approach mixture of part models are motivated by the assumption that geometry and appearance of objects are independent. This assumption allows to reduces model complexity in the case of geometrical appearance diversity. Instead, the proposed hybrid models aim to reduce model complexity arising from variations of semantic part properties (e.g., different lane boundary types).

6 Hierarchical Approach for Holistic Lane and Road Perception at Urban Intersections

Lane and road perception in urban intersections is a challenging problem, due to the vast amount of scenarios, complex road topologies, occlusion caused by other vehicles and strongly varying lane an road cues. In recent years much effort has been devoted to tasks related to this problem, such as the detection of turn arrows, stoplines and curbstones [Enz13, Oni11, Ned09] in urban environments. Other approaches focus on recognizing the ego-lane [Dan09] or multiple lanes around the vehicle [Duc10, Kue12, Hur13], while relying on lane marking cues [Duc10, Kue12, Hur13] or a combination of lane marking cues and curbstone cues [Sei13, Dan09].

Even though these approaches show promising results, neither of them provides the level of scene understanding required for future Advance Driver Assistance Systems (ADAS) which may aim to perform driving maneuvers, such as automatic turns at intersections or automatic stopping at stoplines [Hil12, Her12]. The main bottleneck in the development of these systems is that they required not only the recognition of complex road topologies (e.g., multiple lanes including splitting and merging lanes), but also to estimate the locations of stoplines as well as the turn-direction of lanes (see Figure 6.1). Such a holistic lane and road perception, however, is beyond the reach for state-of- the-art approaches (see Chapter 2).

This chapter, builds on the hybrid compositional hierarchical model (CHM) introduced in Section 5.4, extending it for incorporating both lane boundary cues as well as lane center cues (e.g., stoplines and turn arrows). A key benefit of the proposed CHM is that it allows to model both low-level and high-level dependencies. Low-level dependencies may, for example, encode that turn arrows are located in the middle of a lane or that lane boundaries and stoplines form a u-shape. High-level dependencies, on the other hand, encode plausible constellations of stopline locations or lane turn directions on multi-lane roads. In analogy to Section 4.3.1, modeling such dependencies allows to decrease the influence of partial occlusions and clutter which is expected to improve the overall performance of the proposed approach. While these mechanism mainly rely on spatial dependencies, the presented framework also provides the possibility to model classification uncertainties which for example arise during the classification of turn arrows. Intuitively, this

provides the possibility to introduce constraints which ensure the plausibility of turn-lane direction of neighboring lanes.

As before inference is performed using Nonparametric Belief Propagation (NBP) [Sud03, Isa03], part-sharing (Section 4.3.2) and depth-first message passing (Section 4.3.3). Further, the possibility of using different message passing schedules to control the order in which low-level cues are incorporated in the perception process is discussed. Even though, in exact belief propagation the order of message passing does not matter, in the context of NBP it allows to control the way samples are distributed during inference and therefore to influence both performance and computational complexity of the proposed approach.

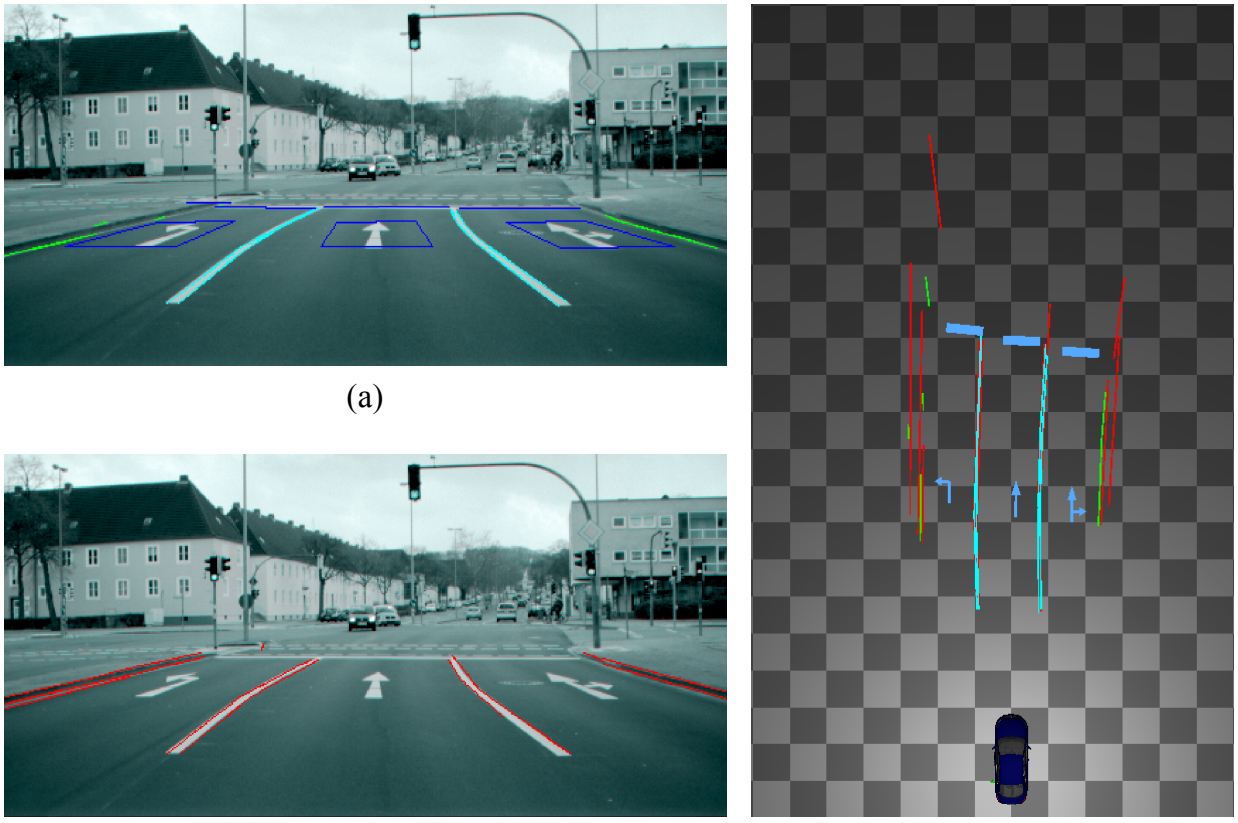
The following sections are structured as follows: Section 6.1 summarizes properties of intersection roads and specifies the set of features used in the experiments, Section 6.2 presents the proposed CHM for multi-lane intersection roads and Section 6.3 discusses different message passing schedules for multi-cue fusion.

6.1 Semantic and Cues of Intersection Roads

Even for human drivers lanes and roads at intersections are particularly complex, due to their complex topology (see Definition 5) and the various cues encoding their semantics. An example for an intersection road is given in Figure 6.1. This figure shows different lane boundary cues (e.g., lane markings and curbstones) and lane center cues, such as stoplines and turn arrows. Particularly stoplines and turn arrows encode important semantic road properties. More precisely, turn arrows specify the turn direction of the individual lanes which are important for any local navigation tasks, such as driving into a desired direction. Further, stoplines indicate the location where a vehicle must stop in case of a red traffic light or a stop sign.

In order to capture all these properties, a combination of different feature detection approaches is applied to the input images. Then, the extracted features are transformed to the vehicle coordinate system using the camera parameters. First, lane boundary features $\boldsymbol{b} = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{N_b}\}$ are extracted using the lane marking detector and the road edge detector described in Section 5.2.

Furthermore, a set of N_s stoplines $\boldsymbol{s} = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_{N_s}\}$ is detected using a modified version of the lane marking detector. Turn arrows $\boldsymbol{a} = \{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{N_a}\}$ are detected in two stage as proposed in [Ned09]. First, the lane marking detector and a geometrical arrow model are used to detect turn arrow candidates. Second, the turn arrow type is classified using a decision tree, which was automatically learned using Weka [Hal09]. After obtaining both arrows and stopline, they are combined to a set of lane center features $\boldsymbol{c} = \boldsymbol{a} \cup \boldsymbol{s}$. A single center feature $\boldsymbol{c} = (x_c, y_c, \vartheta_c, \boldsymbol{t}_c)$ is defined by its position $(x_c, y_c) \in \mathbb{R}^2$, orientation $\vartheta_c \in [0, 2\pi)$ and a discrete



(a) (b) (c)

Figure 6.1: Low-level cues used to support lane and road perception at urban intersections. (a) Lane marking features (cyan, green), stoplines (blue) and turn arrows (blue). (b) Road edge features (red). (c) Projection into the vehicle coordinate system showing the classified turn arrows.

variable $\boldsymbol{t}_c$ that denotes the discrete probability distribution over the center feature types *forward*, *left*, *right*, *forward-left*, *forward-right* and *stop*.

Sample results of applying all four feature extraction approaches to a single input image are shown in Figure 6.1.

6.2 Hybrid Models for Intersection Roads

The model used for lane and road perception at intersections is defined as a set of hybrid CHMs $\boldsymbol{\mathcal{G}} = \{\mathcal{G}_1, \ldots, \mathcal{G}_i\}$ which are specified by tree-structured graphs $\mathcal{G}_i = (\mathcal{V}, \mathcal{E})$ (see Section 5.3.2). Each graph $\mathcal{G}_i$ defines the factorization of a joint probability distribution whose variables are associated to the nodes $\mathcal{V}_i$ of the graphical model. As before the nodes $\mathcal{V} = \boldsymbol{z} \cup \boldsymbol{b} \cup \boldsymbol{c}$ are disjoint sets of hidden random variables $\boldsymbol{z}$, observable variables $\boldsymbol{b}$ and observable variables $\boldsymbol{c}$.

The hidden variables $\boldsymbol{z} = \{\boldsymbol{z}_1, \ldots \boldsymbol{z}_{N_z}\}$ represent multi-lane intersection roads

and their subparts. Each variable $\boldsymbol{z}_i = (\boldsymbol{x}_i, \boldsymbol{d}_i^b, \boldsymbol{d}_i^c)$ is defined on a hybrid state space. Here, the continuous variable $\boldsymbol{x}_i$ defines the spatial configuration of a part (see Section 4.2) and $\boldsymbol{d}_i^b$ is a discrete variable specifying the boundary types of each part (see Section 5.4.1). The additional discrete variable $\boldsymbol{d}_i^c$ represents the discrete probability distribution over the different *center types*. For example, lanes may comprise the types *forward*, *left*, *right*, *forward-left*, *forward-right* as well as five more types combining the turn direction with a stopline label. The observable variables in the model $\boldsymbol{b} = \{\boldsymbol{b}_1, \dots \boldsymbol{b}_{N_b}\}$ and $\boldsymbol{c} = \{\boldsymbol{c}_1, \dots \boldsymbol{c}_{N_c}\}$ correspond to the detected lane boundary features and the detected lane center features, respectively. In analogy to Section 5.4.1.3, edges between hidden variables $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ encode either spatial constraints $\psi_{i,j}$ or discrete transitions models $\eta_{i,j}$. Furthermore, edges between observable and hidden variables encode observation potentials ϕ_i, as defined in Equation 5.2.

Assuming that the boundary feature observation and the center feature observations are independent given $\boldsymbol{x}$ and $\boldsymbol{d}$, the joint probability distribution over all variables in the model can be written as follows

$$\begin{aligned} p(\boldsymbol{d}, \boldsymbol{x} | \boldsymbol{b}, \boldsymbol{c}) \propto & \prod_{i \in \mathcal{I}_b} \phi_i^b(\boldsymbol{z}_i, \boldsymbol{b}_i) \prod_{i \in \mathcal{I}_c} \phi_i^c(\boldsymbol{z}_i, \boldsymbol{c}_i) \\ & \prod_{(i,j) \in \mathcal{E}_{d^b}} \eta_{i,j}^b(\boldsymbol{d}_i^b, \boldsymbol{d}_i^b) \prod_{(i,j) \in \mathcal{E}_{d^c}} \eta_{i,j}^c(\boldsymbol{d}_i^c, \boldsymbol{d}_i^c) \\ & \prod_{(i,j) \in \mathcal{E}_{x^b}} \psi_{i,j}^b(\boldsymbol{x}_i, \boldsymbol{x}_j) \prod_{(i,j) \in \mathcal{E}_{x^c}} \psi_{i,j}^c(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{d}_j^c). \end{aligned} \tag{6.1}$$

Here, $\phi_i^b(\boldsymbol{z}_i, \boldsymbol{b}_i)$ and $\phi_i^c(\boldsymbol{z}_i, \boldsymbol{c}_i)$ are the observation potentials (see Equation 5.2). As in Section 5.4.1, $\psi_{i,j}^b(\boldsymbol{x}_i, \boldsymbol{x}_i)$ denotes spatial constraints and $\eta_{i,j}^b(\boldsymbol{d}_i^b, \boldsymbol{d}_i^b)$ are boundary transition models. Further, $\mathcal{I}_b$ denotes the indexes of the set of cliques that are contained in $\boldsymbol{z} \cup \boldsymbol{b}$ and $\mathcal{I}_c$ the indexes of the cliques in $\boldsymbol{z} \cup \boldsymbol{c}$. Novel are the spatial constraints $\psi_{i,j}^c(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{d}_j^c)$ which depend on discrete variables $\boldsymbol{d}_j^c$ as well as the transition models $\eta_{i,j}^c(\boldsymbol{d}_i^c, \boldsymbol{d}_i^c)$ for the center types.

An illustration of this factorization is shown in Figure 6.2, where a two-lane road with stoplines and turn arrows is decomposed into its basic observable parts.

6.2.1 Features and Patches

In analogy with the previous chapters the observable variables $\boldsymbol{b}$ and $\boldsymbol{c}$ comprise the leaves of the graphical model. Each observable node $\boldsymbol{b}_i \subseteq \boldsymbol{b}$ or $\boldsymbol{c}_i \subseteq \boldsymbol{c}$ has an associated hidden feature variable on the first level $\mathcal{L}_1$ of the model. In correspondence to the two kinds of observable variables the model comprises two kinds of hidden feature variables, namely boundary feature variables $\boldsymbol{z}_i^{f_b}$ and center feature

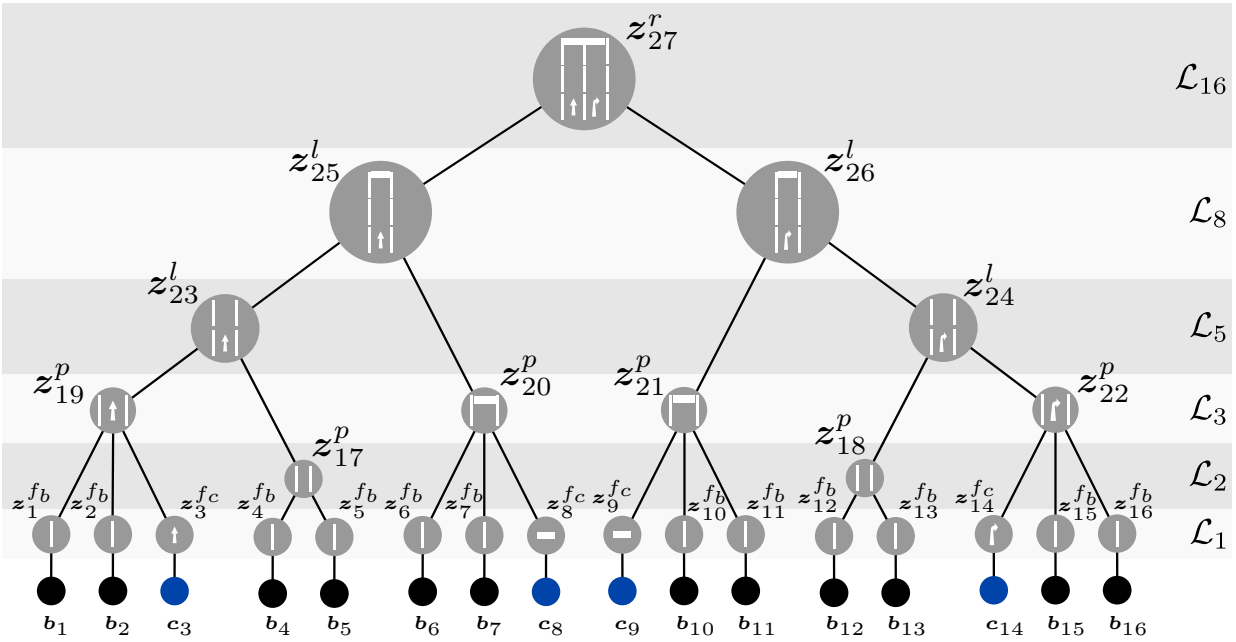


Figure 6.2: CHM of an intersection road. The root node represents a two lane road with stoplines and turn arrows. The hierarchical graph describes the recursive decomposition of this road into basic observable parts on $\mathcal{L}_1$. The most likely center type of the different parts is shown by symbols in the nodes. The additional center feature observations are highlighted in blue.

variables $\boldsymbol{z}_i^{f_b}$.

For both feature variables $\boldsymbol{z}_i^f = (\boldsymbol{x}_i^f, \boldsymbol{d}_i^f)$, the continuous variables $\boldsymbol{x}_i^f = (x_f, y_f, \vartheta_f)$ defines the position $(x_f, y_f) \in \mathbb{R}^2$ and the orientation $\vartheta_f \in [0, 2\pi)$ of a feature. The discrete variables of the two feature variables comprise different instances. Thus, the discrete variable of a boundary feature $\boldsymbol{d}_i^{f_b}$ denotes the probability distribution over boundary feature types 'continuous narrow mark', 'continuous wide mark', 'dashed narrow mark', 'dashed wide mark' and 'road edge', as described in Section 5.4.1. The discrete variable of a center feature $\boldsymbol{d}_i^{f_c}$, on the other hand, denotes the probability distribution over the center feature types 'forward', 'left', 'right', 'forward-left', 'forward-right' and 'stop' (see Section 6.1).

As in the previous chapter, both feature sources the location of a feature and its type are assumed to be independent. This allows to encode the dependencies between the observations and the hidden feature variables using observation potentials $\phi_i^b(\boldsymbol{z}_i^{f_b}, \boldsymbol{b})$ and $\phi_i^c(\boldsymbol{z}_i^{f_c}, \boldsymbol{c})$, as defined in Equation 5.2. Here, $\phi_i^b(\boldsymbol{z}_i^{f_b}, \boldsymbol{b})$ denotes the likelihood of observing a boundary feature given $\boldsymbol{z}_i^{f_b}$ and $\phi_i^c(\boldsymbol{z}_i^{f_c}, \boldsymbol{c})$ the likelihood of observing a center feature given the estimated state of $\boldsymbol{z}_i^{f_c}$.

The observable variables and the hidden feature variables constitute the first levels of the hierarchical model as depicted Figure 6.3 which shows a subset of the graphical model. This figure also shows the proposed factorization of a patch

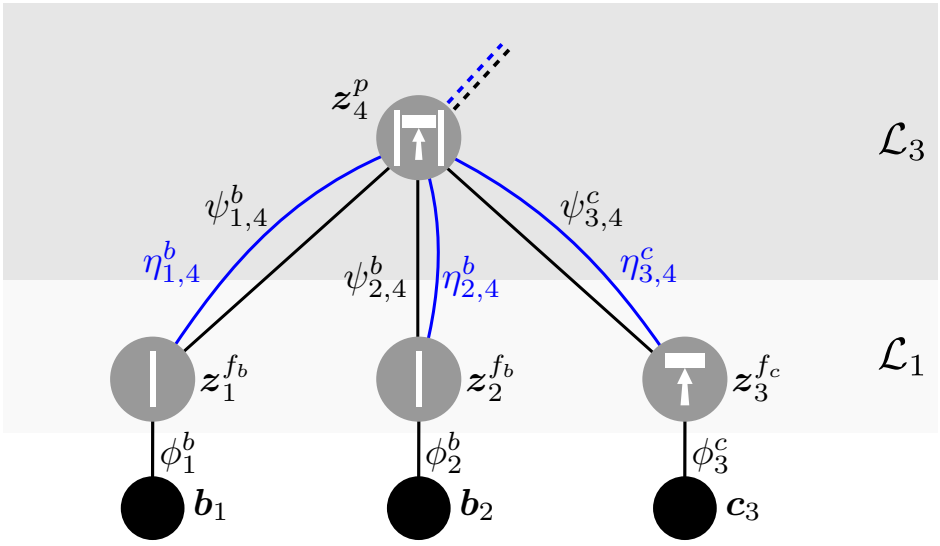


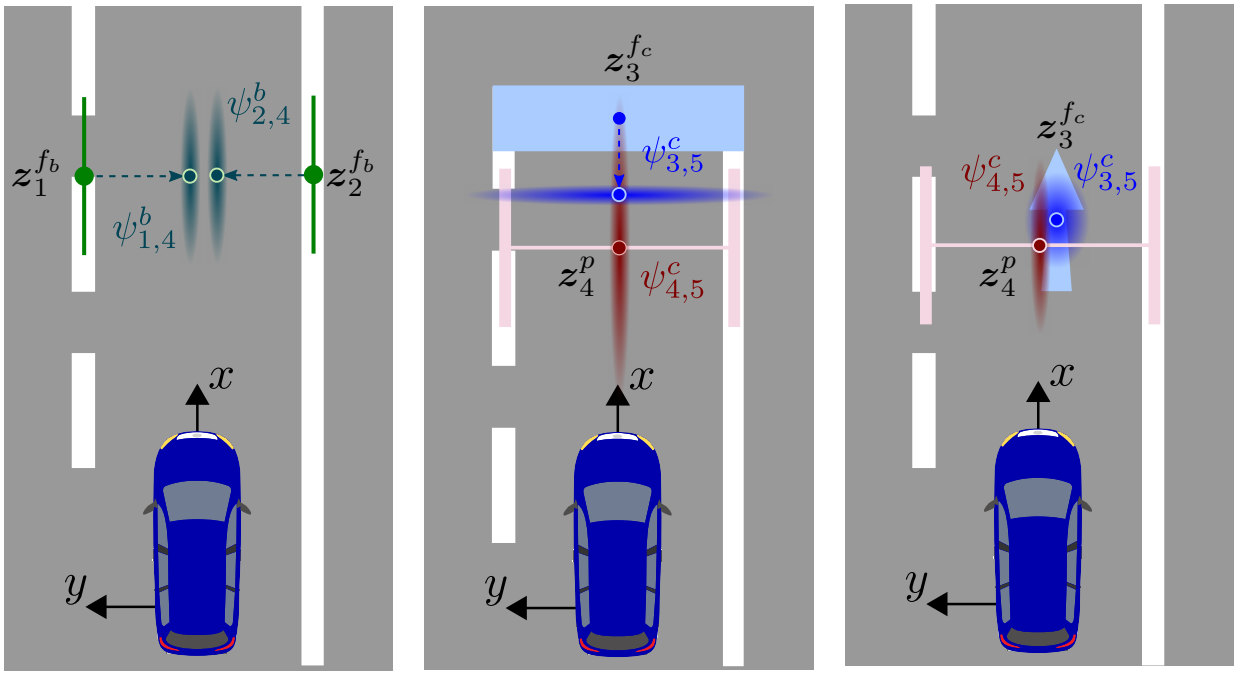
Figure 6.3: Hybrid CHM of a patch with different center types. A patch $\boldsymbol{z}_4^p$ is composed of two boundary features $\boldsymbol{z}_1^{f_b}$ and $\boldsymbol{z}_2^{f_b}$ and a center feature $\boldsymbol{z}_3^{f_c}$. The leaves of the model comprise the boundary observations $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ as well as center observations $\boldsymbol{c}_3$. The edges of the model encode spatial constraints $\psi_{i,j}$, discrete transition models $\eta_{i,j}$ and observation potentials ϕ_i.

with center type. As before patches on $\mathcal{L}_3$ represent local driveable areas and have different boundary types (see Section 5.4.1.1). Furthermore, the shown patch comprises a center feature $\boldsymbol{z}_3^{f_c}$. Generally, the boundary type of a patch and its center type are assumed to be independent.

Definition 7 (Center Patch)**.** *A center patch is defined as* $\boldsymbol{z}^p = \{\boldsymbol{x}^p, \boldsymbol{d}^{p_b}, \boldsymbol{d}^{p_c}\}$*, where* $\boldsymbol{x}^p$ *specifies its spatial configuration and* $\boldsymbol{d}^{p_b}$ *encodes the different patch boundary types, as in Definition 6. Additionally,* $\boldsymbol{d}^{p_c}$ *denotes the probability distribution over the patch center types. These center types are equivalent to the feature center types* $\boldsymbol{d}^{f_c}$*.*

As detailed in Section 5.4.1.3, the spatial dependencies between boundary features and patches are assumed to be independent of their type. Hence, their dependencies can be encoded using spatial constraints $\psi_{i,j}^b(\boldsymbol{x}_j^{f_b}, \boldsymbol{x}_i^p)$ (see Section 4.2.1) and discrete state transitions $\eta_{i,j}^b(\boldsymbol{d}_j^{f_b}, \boldsymbol{d}_i^{p_b})$ (see Section 5.4.1.3). The dependencies between center features and patches are more complex, since their spatial relation depends on their type. Therefore, these relations are defined by unnormalized spatial constraints $\psi_{i,j}^c(\boldsymbol{x}_i^{f_c}, \boldsymbol{x}_j^p, \boldsymbol{d}_j^{p_c})$ as

$$\psi_{i,j}^c(\boldsymbol{x}_i^{f_c}, \boldsymbol{x}_j^p, \boldsymbol{d}_j^{p_c}) = \begin{cases} \mathcal{N}(\boldsymbol{x}_j^p; \boldsymbol{x}_i^{f_c} + \boldsymbol{r}_{i,j}, \boldsymbol{\Sigma}_{i,j}^s) & \text{if } \boldsymbol{d}_j^{p_c} = d_i^s \text{ (stopline)} \\ \mathcal{N}(\boldsymbol{x}_j^p; \boldsymbol{x}_i^{f_c}, \boldsymbol{\Sigma}_{i,j}^a) & \text{otherwise (arrow)} \end{cases} \tag{6.2}$$



(a) (b) (c)

Figure 6.4: Spatial constraints between different features and patches. (a) Spatial dependencies $\psi^b_{i,j}$ between boundary feature and patches (see Section 4.2.1). (b) and (c) show a spatial constraint $\psi^c_{i,j}$ between a patch and a center feature that depends on the center type. (b) For a stopline $\psi^c_{i,j}$ models a large spatial uncertainty orthogonal to the stopline direction. (c) For an arrow $\psi^c_{i,j}$ models a spherical uncertainty. Spatial uncertainties are illustrated by showing Gaussian distributions, where dark colors correspond to more likely locations.

where

$$\boldsymbol{r}_{i,j} = \begin{pmatrix} -sin(\vartheta_j^{f_c}) & 0 & 0 & 0 & 0 \\ 0 & -cos(\vartheta_j^{f_c}) & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \nu_{l_p} \\ \nu_{l_p} \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{6.3}$$

is the relative configuration vector between the stopline feature and the patch center. Note that, the orientation of the center feature $\vartheta_j^{f_c} \in [0, 2\pi)$ is parallel to the patch orientation and the length $\nu_{l_p} \in \mathbb{R}^+$ is the constant patch length introduced in Section 4.2.1. The covariance matrices $\boldsymbol{\Sigma}^s_{i,j} \in \mathbb{R}^{4\times4}$ and $\boldsymbol{\Sigma}^a_{i,j} \in \mathbb{R}^{4\times4}$ encode different spatial uncertainties depending on the patch center type. This is illustrated in Figure 6.4, by showing the spatial constrains $\psi_{1,4}$ and $\psi_{2,4}$ encoding the spatial dependencies between boundary features $\boldsymbol{z}_1^{f_b}$ and $\boldsymbol{z}_2^{f_b}$ and the patch $\boldsymbol{z}_4^p$. Further, this figure shows that depending on the type of $\boldsymbol{d}_5^{p_c}$ the spatial constraint

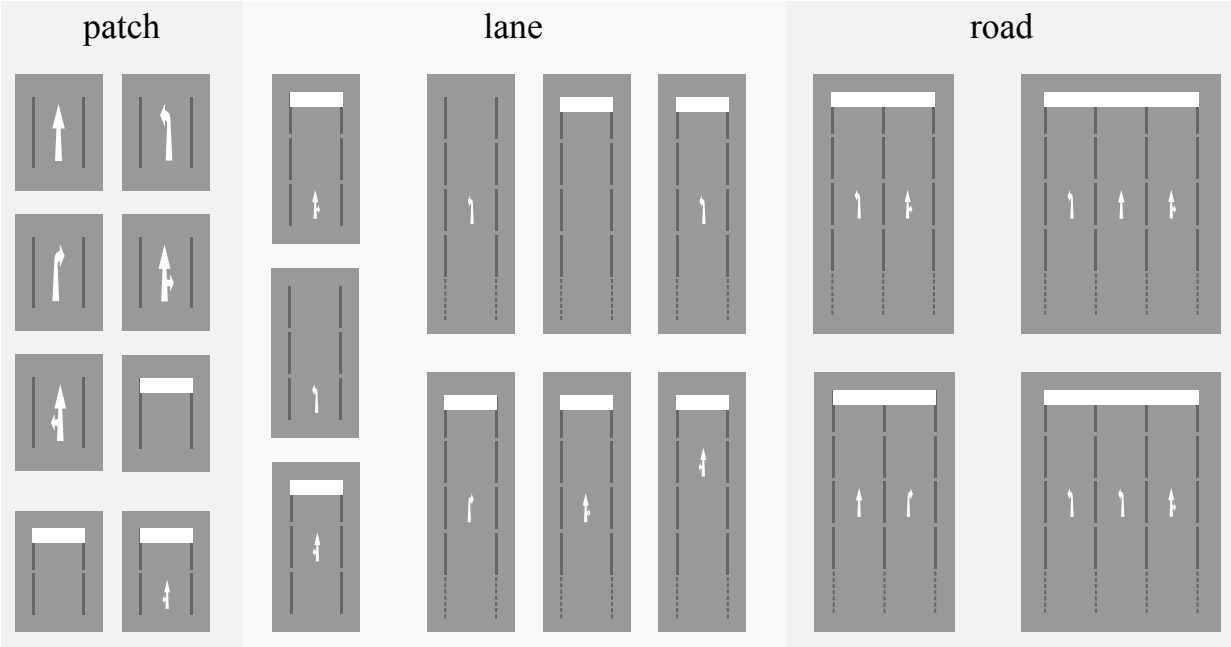


Figure 6.5: Sample patch center types, lane center types and road center types. On each lane level the model comprises a set of center types which constrain the output of the proposed approach to plausible center type sequences. Road center types define plausible constellations of lane turn directions of neighboring lanes.

$\psi^c_{3,5}$ encodes different relative configurations and spatial uncertainties.

In addition, the dependencies between the boundary types of features and patches as well as the dependencies between center feature types and patch center types have to be modeled . Assuming independence between boundary types and center types, these dependencies are encoded using the transition model $\eta^b_{i,j}(\boldsymbol{d}^b_i, \boldsymbol{d}^b_j)$ and $\eta^c_{i,j}(\boldsymbol{d}^c_i, \boldsymbol{d}^c_j)$ for the boundary types and the center types, respectively.

6.2.2 Lanes at Intersections

Building on the lane representation defined in Definition 6, it is augmented by a discrete variable $\boldsymbol{d}^{l_c}$ which defines plausible sequences of lane center types, as depicted in Figure 6.5. As the different road topologies in Section 5.3.2, these sequences are defined according to official roadway construction guidelines [ras95]. Thus, lane center types express the prior expectation on arrows and stopline constellations and are expected to improve the robustness of the proposed approach. Furthermore, the location of arrows and stoplines along the lane centerline is specified by a of N_c variables $k^{l_c}_n \in \mathbb{R}^+, n \in N_c$.

Definition 8 (Intersection Lane)**.** *An intersection lane* $\boldsymbol{z}^l$ *is defined as* $\boldsymbol{z}^l = \{\boldsymbol{x}^l, \boldsymbol{d}^{l_b}, \boldsymbol{d}^{l_c}, \boldsymbol{k}^{l_c}\}$*. As in Definition 6,* $\boldsymbol{x}^l$ *defines the lane geometry and* $\boldsymbol{d}^{l_b}$ *spec-*

ifies the lane boundary types. Further, $\boldsymbol{d}^{l_c}$ defines different lane center types and $\boldsymbol{k}^{l_c} = (k_1^{l_c}, \ldots, k_{N_c}^{l_c})$ specifies the location of the N_c center features, where each $k_n^{l_c} \in \mathbb{R}^+, n \in N_c$ is a coordinate along the lane centerline.

The spatial constrains between patches and lanes are assumed to be independent of the boundary types and are therefore defined as

$$\psi_{i,j}(\boldsymbol{x}_i^p, \boldsymbol{x}_j^l, \boldsymbol{d}_j^{p_c}) = \mathcal{N}(\boldsymbol{x}_j^l; S_{i,j}(\boldsymbol{x}_i^p), F_{i,j}(\boldsymbol{d}_j^{p_c})) \tag{6.4}$$

with

$$S_{i,j}(\boldsymbol{x}_i^p) = \begin{pmatrix} x_i \\ y_i \\ \vartheta_i \\ w_i \\ k^{l_c} \end{pmatrix} + \begin{pmatrix} \cos(\vartheta_i) & 0 & 0\ 0\ 0 \\ 0 & \sin(\vartheta_i) & 0\ 0\ 0 \\ 0 & 0 & 1\ 0\ 0 \\ 0 & 0 & 0\ 1\ 0 \\ 0 & 0 & 0\ 0\ 1 \end{pmatrix} \begin{pmatrix} \nu_{l_p} \\ \nu_{l_p} \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{6.5}$$

Here $F_{i,j}(\boldsymbol{d}_j^{p_c})$ returns the covariance matrix of the spatial relation depending on the assignment of $\boldsymbol{d}_j^{p_c}$. These covariance matrices only differ in their entries for the variances $\delta_{k^{l_c}}$ which allows to constrain the coordinate of stoplines and arrows to the desired location. For example, if a patch is not a stop-patch the variance $\delta_{k^{l_s}}$ is large, since the patch does not provide any knowledge on the location of the stopline. On the other hand, if a patch is likely to be a stop-patch, a small value for δ_{k^s} is selected.

The spatial constraints between adjacent lane variables are defined as

$$\psi_{i,j}^c(\boldsymbol{x}_i^l, \boldsymbol{x}_j^l, \boldsymbol{d}_j^{l_c}) = \mathcal{N}(\boldsymbol{x}_j^l; S_{i,j}(\boldsymbol{x}_i^l), F_{i,j}(\boldsymbol{d}_j^{l_c})), \tag{6.6}$$

where $S_{i,j}(\boldsymbol{x}_i^l)$ has a similar form as in Equation 6.5. However, they predict the value of k^{l_s} based on the constant patch length ν_{l_p}. Further, $F_{i,j}(\boldsymbol{d}_j^{l_c})$ returns the covariance matrix of the spatial relation. This function is dependent on the lane center type $\boldsymbol{d}_j^{l_c}$. The lane center type comprises the estimate of the lane center type and thus the probability that the next lane segment comprises a stopline, an arrow or no center type. In the spatial constraint these information are used to define values for the variances $\delta_{k^{l_c}}$ and the configuration uncertainties of the next lane segment. The latter of which follows the definition given in Equation 6.2.

Let us now consider the formulation of the required transition models for the boundary types. Generally, the boundary types of neighboring variables are assumed to be independent of their center type and their spatial configuration. Thus, the boundary type transitions models introduced in Section 5.4.1 are used.

Finally, the transition model $\eta_{i,j}^c(\boldsymbol{d}_i^{p_c}, \boldsymbol{d}_j^{l_c})$ and $\eta_{i,j}^c(\boldsymbol{d}_i^{l_c}, \boldsymbol{d}_j^{l_c})$ have to be specified, where these are assumed to be independent of the spatial configuration and

the boundary type of their associated parts. The transition model $\eta^c_{i,j}(\boldsymbol{d}^{l_c}_i, \boldsymbol{d}^{l_c}_j)$ between neighboring lane center types $\boldsymbol{d}^{l_c}_i$ and $\boldsymbol{d}^{l_c}_j$ models the dependency between the lane center sequences of different length and are specified as discrete conditional probabilities. Similarly, the transition models $\eta^c_{i,j}(\boldsymbol{d}^{p_c}_i, \boldsymbol{d}^{l_c}_j)$ between patches and lanes specify their conditional dependency.

6.2.3 Roads at Intersections

Similar to the lane variables, each road variable is augmented with a hidden discrete variable $\boldsymbol{d}^{r_c}_i$ which encodes plausible constellations of lane turn directions, as shown in Figure 6.5. Further, roads comprise a set of N_c variables $k^{r_s}_n \in \mathbb{R}^+, n \in N_c$ which specify the locations of the center features along the road centerline. The origin of these coordinate is the first segment of the road centerline.

Definition 9 (Intersection Road)**.** *An intersection road $\boldsymbol{z}^r = \{\boldsymbol{x}^r, \boldsymbol{d}^{r_b}, \boldsymbol{d}^{r_c}, \boldsymbol{k}^{r_c}\}$ is defined by its topology $\boldsymbol{x}^r$ and its boundary types $\boldsymbol{d}^{r_b}$, as in Definition 6. Further, $\boldsymbol{d}^{r_c}$ defines different road center types and $\boldsymbol{k}^{r_c} = (k^{r_c}_1, \ldots, k^{r_c}_{N_c})$ specifies the location of the N_c center features along the road centerline.*

The dependencies between lanes and roads are encoded using boundary type transition models as in Section 5.4.1, center type transition models $\eta^c_{i,j}(\boldsymbol{d}^{l_c}_i, \boldsymbol{d}^{r_c}_j)$ and spatial constraints $\psi_{i,j}(\boldsymbol{x}^l_i, \boldsymbol{x}^r_i, \boldsymbol{d}^{r_c}_j)$ which encode the spatial relation between lanes and roads. These spatial constraints are based on the definition given in Equation 4.7. However, they are extended to include the additional one dimensional center feature variables. Therefore, the spatial constraints between a road and its i-th lane is defined as

$$\psi_{i,j}(\boldsymbol{x}^l_i, \boldsymbol{k}^{l_c}_i, \boldsymbol{x}^r_j, \boldsymbol{k}^{r_c}_j) = \psi_{i,j}(\boldsymbol{x}^s_i, \boldsymbol{x}^s_j)\psi_{i,j}(w_i, w_{j,i})\psi_{i,j}(\boldsymbol{k}^{l_c}_i, \boldsymbol{k}^{r_c}_j), \tag{6.7}$$

where $\psi_{i,j}(\boldsymbol{x}^s_i, \boldsymbol{x}^s_j)$ and $\psi_{i,j}(w_i, w_{j,i})$ are defined according to Equation 4.8 and Equation 4.9, respectively. Furthermore,

$$\psi_{i,j}(\boldsymbol{k}^{l_c}_i, \boldsymbol{k}^{r_c}_j) = \sum_{l=1}^{N_c} \mathcal{N}\left(k^{l_c}_{i,l}; k^{r_c}_{j,l}, \delta_{i,l}\right), \tag{6.8}$$

encodes the spatial dependency of the center features of lanes and roads. Here, $\delta_{i,l}$ models the spatial uncertainty regarding the location of the center features.

Furthermore, the proposed model comprises center type transitions $\eta^c_{i,j}(\boldsymbol{d}^{l_c}_i, \boldsymbol{d}^{r_c}_j)$ that encode the dependency between the lane and road center types by means of conditional probabilities. Let us now take a closer look at these

road center types. At the first glance the road center types seem very complex. In fact, one could also encode road types which simply define the turn direction of each lane. This would simplify both the road variables and the specification of the transition model $\eta^c_{i,j}(\boldsymbol{d}^{l_c}_i, \boldsymbol{d}^{r_c}_j)$. However, this simplified road model does not contain any knowledge on the relative location of arrows along the lanes of a road. The proposed model, on the other hand, explicitly models plausible locations and types of arrows on neighboring lanes, e.g., arrows are parallel. This brings along two main benefits. First, by constraining arrows to specific locations, the influence of clutter or false positives on the overall result is limited. Second, it allows to identify missing evidence and thus to initiate a top down search for, e.g., missing arrow features, as detailed in Section 6.3.2.

A CHM for a single intersection road is shown in Figure 6.2, where the spatial constraint $\psi_{i,j}(\boldsymbol{x}^l_i, \boldsymbol{k}^{l_c}_i, \boldsymbol{x}^r_j, \boldsymbol{k}^{r_c}_j)$ encodes the dependencies between lanes and roads, including assumptions on the relative location of arrows and stoplines. Not explicitely shown are, the correlation functions $\eta^c_{i,j}(\boldsymbol{d}^{l_c}_i, \boldsymbol{d}^{r_c}_i)$ and $\eta^b_{i,j}(\boldsymbol{d}^{l_b}_i, \boldsymbol{d}^{r_b}_i)$ encoding the dependencies between road types and lane types which ensure the plausibility of lane turn direction constellations on the road level.

6.3 Inference of the Road Topology and Semantic

This section, introduces a novel inference algorithm for lane and road perception for urban intersection roads. Towards this goal, the inference algorithm presented in Section 5.4.2 is extend to incorporate the center types on the different levels of the proposed hierarchical model.

6.3.1 Message Passing

The goal of inference is to estimate the most likely instance of the hybrid CHM. This task is solved using a version of NBP [Isa03, Sud04b] with nearest neighbor product sampling [Spe13] for the continuous valued message products and Belief Propagation (BP) [Wei01, Yed01] for the message products only involving discrete variables. In this case the message update equations for the continuous part is

$$m^x_{j,i}(\boldsymbol{x}_i) = \sum_{\boldsymbol{d}^c_j} \int_{\boldsymbol{x}_j} \psi^c_{i,j}(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{d}^c_j) \prod_{k \in \Upsilon(j) \setminus i} m^x_{kj}(\boldsymbol{x}_j) d\boldsymbol{x}_j. \tag{6.9}$$

For the discrete part, the boundary messages

$$m^c_{j,i}(\boldsymbol{d}^c_i) = \sum_{\boldsymbol{d}^c_j} \eta^c_{i,j}(\boldsymbol{d}^c_i, \boldsymbol{d}^c_j) \prod_{k \in \Upsilon(j) \setminus i} m^c_{kj}(\boldsymbol{d}^c_j) \tag{6.10}$$

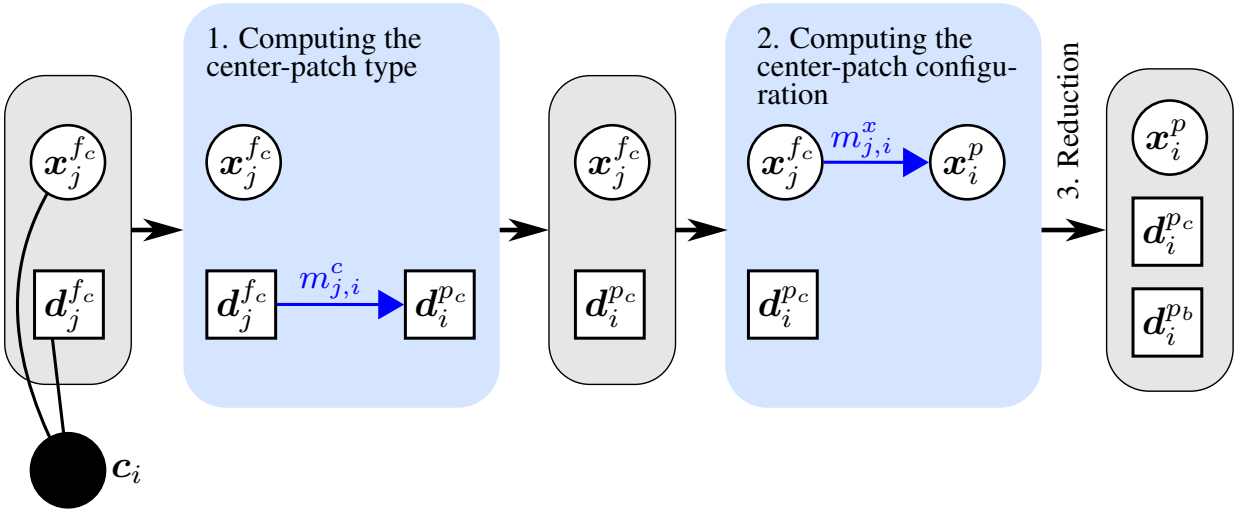


Figure 6.6: Messages passing algorithm for approximate inference in the proposed hybrid CHM. At first, the discrete patch type $\boldsymbol{d}_i^{p_c}$ is computed. Then, the center-patch configuration $\boldsymbol{x}_i^p$ is computed. Finally, the mixture components contained in the nonparametric belief estimate of $\boldsymbol{x}_i^p$ are reduced, for computational efficiency. Not shown are the messages from the boundary features which are required to update the boundary variable $\boldsymbol{d}_i^{p_b}$ and the patch configuration $\boldsymbol{x}_i^p$.

as well as the center messages

$$m_{j,i}^b(\boldsymbol{d}_i^b) = \sum_{\boldsymbol{d}_j^b} \eta_{i,j}^b(\boldsymbol{d}_i^b, \boldsymbol{d}_j^b) \prod_{k \in \Upsilon(j) \setminus i} m_{k,j}^b(\boldsymbol{d}_j^b) \tag{6.11}$$

have to be computed, where Equation 6.10 and 6.11 are defined as in Section 5.4.2. The main difference to the previous discussion is that the message update for the continuous variables in Equation 6.9 requires to marginalize over $\boldsymbol{d}_j^c$. In this case, each instance of $\boldsymbol{d}_j^c$ produces a different mixture component as defined in Equation 6.2. During message passing this leads to the problem that the messages contain an enormous amount of mixture components. In order to keep inference traceable, some kind of approximation has to be found.

Towards this goal, an approach that is inspired by the Interacting Multiple Model (IMM) algorithm [Kol09] is used, as depicted in Figure 6.6. This figure shows that the belief over the patch variables is computed in several steps. First, the belief over the patch center types $b_i(\boldsymbol{d}_i^{p_c})$ is computed. Subsequently, $b_i(\boldsymbol{x}_i^p)$ is computed, where each instance of $\boldsymbol{d}_i^{p_c}$ produces a different mixture component, as defined in Equation 6.9. The final and most important part of the proposed inference algorithm is to reduce the mixture to a smaller mixture with K components. Towards this goal, the K components are retained with the largest mixture weight. This approach is particularly suitable for the proposed approach, since the mixture weight of each component represents both, its spatial plausibility and the proba-

bility of the discrete instance that produced it (see Equation 5.3).

Due to illustrative simplicity Figure 6.6 does not show, messages $m^x_{k,i}$ and $m^x_{n,i}$ which are send from the two boundary feature variables and predict the location of the patch (see Figure 6.3). Further, messages $m^b_{k,i}$ and $m^b_{n,i}$ predicting the patch boundary type are not depicted.

The above expositions mainly focus on the task of computing patches. This example was chosen, since it clarifies how inference is performed if the spatial relations depend an the instances of the center type variable (see Equation 6.2). Computing lanes and roads requires the application of the same inference algorithm, whereas the message passing schedule follows the depth first message passing algorithm presented in Section 4.3.3.

6.3.2 Message Passing Schedule in Multi-Cue Models

Message passing in the proposed framework is based on the depth-first message passing algorithm, detailed in Section 4.3.3. This message passing algorithm requires to specify a fixed message passing schedule, to guaranty non-degenerative message products. The hybrid model, provides various possibilities to specify this message passing schedule. In the following, some of these possibilities are discussed alongside their expected influence on the overall recognition results.

Multi-Cue Patch Recognition: Let us first consider the task of recognizing a patch with center type, using the CHM depicted in Figure 6.3. In this CHM, message passing can be performed in the two principal ways depicted in Figure 6.7. First, as shown in Figure 6.7a, message passing can be initiated by propagating messages from the boundary feature nodes $\boldsymbol{z}_1^{f_b}$ and $\boldsymbol{z}_2^{f_b}$ to the patch node $\boldsymbol{z}_4^p$. Subsequently, an expectation-based search for low-level evidence can be performed by propagating messages back and forth to node $\boldsymbol{z}_3^{f_c}$. This latter step is required, since it can not assume that there was already a message send from $\boldsymbol{z}_3^{f_c}$ to $\boldsymbol{z}_4^p$ (see Section 4.3.3). Finally, by fusing all incoming messages at node $\boldsymbol{z}_4^p$, its bottom up belief state can be computed according to Equation 4.11.

Second, as shown in Figure 6.7b, message passing can be initiated by predicting a patch hypothesis $\boldsymbol{z}_4^p$ based on the local belief state of node $\boldsymbol{z}_3^{f_c}$ and subsequently fuse it with the evidence at $\boldsymbol{z}_1^{f_b}$ and $\boldsymbol{z}_2^{f_b}$.

Both of the above message passing schedules have individual strength. The message passing schedule depicted in Figure 6.7a is particularly suitable if the observations $\boldsymbol{c}$ contain many false positives, since samples for product approximation are selected based on the predicted patch configuration. Hence, features at $\boldsymbol{z}_3^{f_c}$ which are not similar to a patch sample at $\boldsymbol{z}_4^p$ do not have to be processed and do not influence the recognition results (see Section 4.3.3).

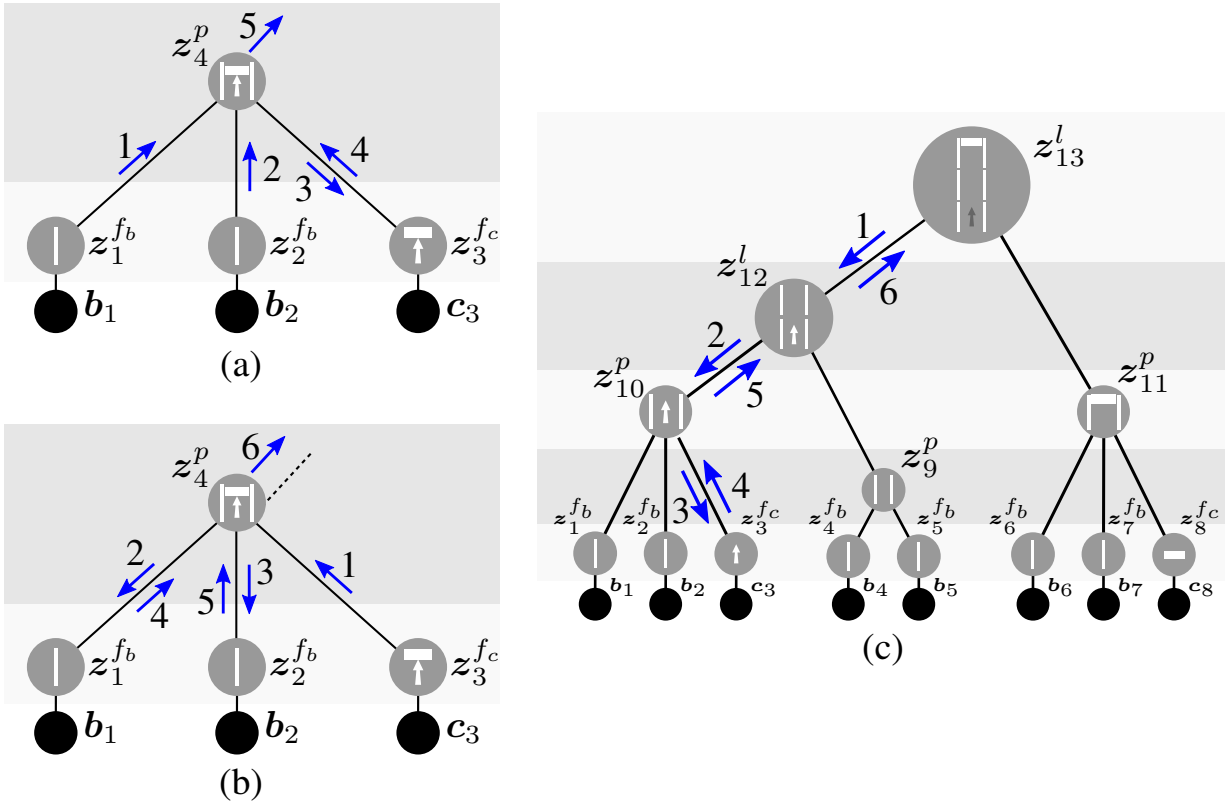


Figure 6.7: Different message passing schedules for multi-cue lane perception. (a) Recognition of a center feature $z_3^{f_c}$ based on patch hypothesis z_4^p. (b) Recognition of a patch hypothesis based on an observed center feature $z_3^{f_c}$. (c) An observed stopline at $z_8^{f_c}$ can be used to initialize the recognition of an arrow at $z_3^{f_c}$, using the structure of the proposed model.

On the other hand, the message passing schedule shown in Figure 6.7b, is expected to be suitable if the results obtained by the center feature detector comprise less false positives than the results of the boundary feature detector. In this case, it is computational unattractive to perform message passing as shown in Figure 6.7a, since for the many patch hypotheses, the bottom up/top down sweep has to be performed. Another advantage of the message passing schedule in Figure 6.7b is that it allows to generate patch hypotheses even if no lane boundary features are detected and thus may allow for recognizing lanes even in scenarios, where only arrows and stoplines are present.

Multi-Cue Lane and Road Recognition: Another key motivation for the proposed CHM, is that it allows to use the dependencies between arrows and stoplines to improve the performance of arrow and stopline recognition. This is shown in Figure 6.7c, where the arrow may not have been detected during the bottom up sweep. However, if a stopline is recognized during feature extraction (see Sec-

tion 6.1), a search for the potentially missing arrow can be initialized using the lane center types (see Figure 6.5).

Further, Section 6.2.3 explains how roads are used to encode plausible constellations of lane turn directions. During message passing this allows to more precisely predict the turn directions of neighboring lanes. Similarly to Section 4.3.3, messages send from the road node to the lane nodes are used to propose lanes, including their spatial configuration, stopline positions and lane turn directions. Subsequently, either the propose lane hypotheses can be evaluated, as detailed in Section 4.3.3 or used to initialize a search for low-level feature, supporting e.g., stopline locations or turn arrow types.

7 Experimental Evaluation

This chapter presents a set of quantitative and qualitative results of applying the proposed framework to challenging real world scenarios. These experiments aim to demonstrate that the approach presented in this thesis:

- can robustly infer complex road topologies (e.g., splitting and merging lanes) from low-level observations in real-time.
- can simultaneously infer a richer amount of topological and semantic road properties.
- allows to benefit from topological and semantic a priori knowledge.
- can incorporate different vision approaches and thus be applied to a wide domain, i.e., highways, rural roads and urban intersections.
- allows to improve detection results by introducing sensor specific inference schedules.

7.1 Experimental Setup and Dataset

The dataset used for the experiments performed in this thesis comprises 47 sequences with a duration of 20 to 40 seconds, featuring multi-lane highways, highway ramps, rural roads, two-lane intersection roads and three-lane intersection roads. Each sequence captures a realistic moment of driving, where most of the data were captured during low traffic density, i.e., the road is often completely visible. Figure 7.1 depicts a selection of sequences of the dataset, showing the variability of road topology and appearance.

Further the dataset comprises images with a resolution of 1024×512 pixel and a field of view of about $45°$. Additionally, the dataset comprises 3D LIDAR scans, high-accuracy (<10 cm) DGPS measurements and IMU accelerations and velocities from a combined DGPS and IMU system.

Since manual labeling of the dataset is impractical, a novel automatic labeling approach is used. The fundamental idea of this approach is to assign the results to one of the four categories in the confusion matrix (i.e., false positive (FP), false negative (FN), true positive (TP) and true negative (TN)) by aligning the detection results to a geo-referenced ground truth database [Kna10b, Hom11]. This ground truth comprises detailed information on roads, such as number, location and extend

Figure 7.1: Sample images from the evaluation dataset. This figure shows 9 images out of the 47 sequences used to evaluate the proposed method. Shown are frames of highways, rural roads and urban roads which differ significantly in topological complexity and appearance.

of lanes, the locations of stoplines, and the turn directions of lanes (see Figure 7.2). The overall spatial accuracy of the database is approximately 10 cm. During the labeling, for each input image, the ground truth data is transformed in the vehicle coordinate system using the DGPS+IMU data comprised in the dataset. Then, the labeling results are obtained by aligning ground truth and detection results. More details on this alignment process and how errors in both DGPS locations and the ground truth database are considered in the evaluation are presented in Section 7.2.

During the experiments, the lane marking detector and the road edge detector are considered, as presented in Section 5.2. Further, the stopline and turn arrow detectors are employed, as presented in Section 6.1. All detection results are projected in the vehicle coordinate system, assuming a planar ground plane.

Even though the dataset was captured at low traffic density, it may comprise other vehicles occluding the field of view. This leads to the issue that parts of the lane and road cues are occluded. The ground truth, however, is not affected by such occlusion. Thus, the ground truth and the detection results are not comparable. In order to cope with this issue, the 3D LIDAR data of the dataset is used to compute the actual field of view which is called dynamic detection range. Subsequently, both feature extraction results and ground truth data are restricted to this dynamic detection range.

All experiments presented in the following sections are performed on a Win-

dows(TM) laptop with Intel(R) Core(TM) i5-3427U CPU with 1.80 GHz and 8.00 GB computer memory. All algorithms are implemented in C++ and executed in the Automotive Data and Time-Triggered Framework (ADTF) [Gmb13]. For all experiments on the execution time, the profiling tool of ADTF is used.

7.2 Evaluation Metric and Data Annotation

For analyzing the results of lane detection approaches multiple evaluation metrics have been proposed. In general, existing evaluation metrics can be categorized into two different approaches: metrics that directly operate in the perspective image domain [Alv08, Wu11, Shi12, Guo12a] and metrics that are applied in the vehicle coordinate system. The following sections focus on the latter approach, since the results of the proposed approach are obtained in the vehicle coordinate system, as required by most vehicle control applications.

Corresponding to the large number of works in the field of lane detection a large number of metrics for evaluating their results have been proposed. For examples, to quantify the results of lane boundary detection, traditionally the spatial deviation between detected lane-boundaries and ground truth lane boundaries is evaluated [Ser08, Zha12, Hur13]. Further, in [Zha12, Gop12, Guo12b, Lin11] a flexible margin is applied during the matching of detections and ground truth which allows to compute true positive rates and false positive rates. Another important measure for ADAS is the occupied lane length which is evaluated in [Gum11]. Additionally, evaluation measures that focus on the width of the driving corridor have been used in [Kuh11, Kue12]. Finally, in [Fri13] the authors propose two evaluation measures for the vehicle coordinate system. First, a pixel-based evaluation for applications requiring a very detailed lane shape detection. Second, a behavior-based evaluation metric that generates different driving corridor hypotheses based on a single track model. The confidence of each hypothesis is evaluated by aligning it to the detection results which allows for a task dependent abstraction of the original data inputs. Finally, the corridor hypothesis with the highest confidence is compared to the ground truth.

All the above evaluation metrics have their advantages. However, each of them only covers a subset of the results obtained by the proposed holistic lane and road perception approach. More precisely, the above approaches mainly focus on evaluating the results of lane and road topology detection. During the evaluation, however, the performance of stopline recognition and lane turn direction estimation has to be quantified. Further, and most important, the influence of the contextual knowledge that each level of the Compositional Hierarchical Model (CHM) introduces in the perception process has to be quantified, as it is the key benefit of the proposed approach. In order to evaluate all these aspects, the evaluation is per-

formed using an overlap-based evaluation metric on the patch level, as detailed in the next section.

7.2.1 Evaluation Metric

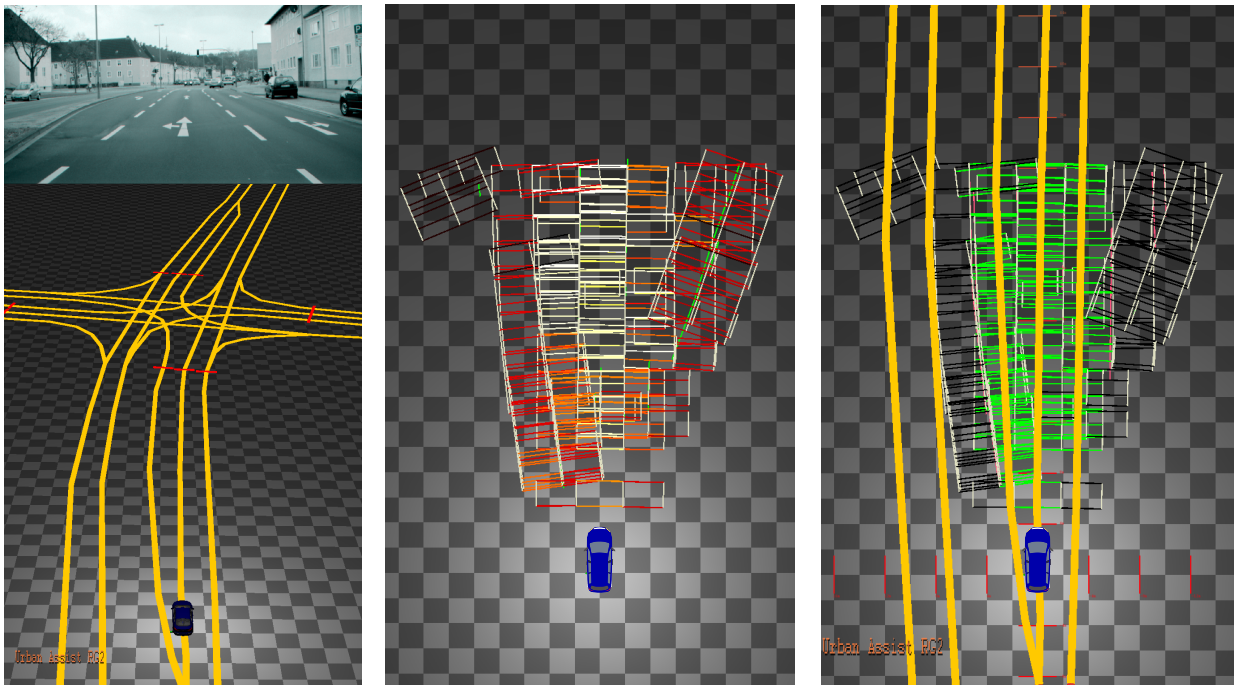
The fundamental task of the overlap-based evaluation metric is to fit the results of the proposed approach to the ground truth and to judge if a detection is a positive example or negative example, i.e., the evaluation is treated as a binary classification problem. During the experiments, this labeling is performed on the patch-level, since it is particularly suitable to evaluate the influence of contextual information provided by the lane and road levels on the overall recognition performance. Additionally, labeling on the patch-level allows to evaluate not only the results of lane and road detection, but also to quantify the results of stopline and lane turn direction recognition.

One of the key advantages of the outlined labeling method is that given a ground truth database it allows to perform the evaluation without exhaustive labeling of individual input images.

To perform the labeling, it is necessary to transform the ground truth data from the Universal Transverse Mercator (UTM) coordinate systems [KAW13] to the coordinate frame of the ego-vehicles. This transformation is based on the estimated GPS location of the ego-vehicle and hence can not be performed without errors. In addition, it has to be assumed that the ground truth has small inaccuracies. Therefore, the labeling allows for small variations during the association of ground truth data and detection results. More precisely, during the association, each detected patch sample is convolved with a multivariate Gaussian distribution which models inaccuracies in the xy-plane as well as orientation and width errors. Subsequently, positive or negative labels are assigned to the detection depending on if a ground truth element is found in the 1σ-range of the patch or not. During the experiments, the parameters of the covariance matrix are fixed to $\delta_x = \delta_y$ =0.3 m for the xy-variance, to δ_θ =0.1 rad for the angular variance and to δ_w =0.2 m for the variance of the width.

Given the labeling results, the weight of each patch sample or hypothesis can be used to assign it to one of the following four categories: True Positives (TP) are examples correctly labeled as positives, False Positives (FP) refer to negative examples incorrectly labeled as positives, True Negatives (TN) correspond to negatives correctly labeled as negatives and False Negatives (FN) refer to positive examples incorrectly labeled as negatives. This allows to quantify the performance of the proposed approach using precision and recall. These two metrics are defined as

$$Precision = \frac{TP}{TP + FP} \tag{7.1}$$



(a) (b) (c)

Figure 7.2: Evaluation framework. (a) Ground truth road network database [Kna10b, Hom11]. (b) Nonparametric approximation of the joint probability distribution over all patch variables. The importance weights of patch-hypotheses are depicted, showing different colors on the temperature scale, where low weights correspond to dark red and high weights to white. (c) Labeling results, showing positively labeled patch samples (green) and negatively labeled patch samples (black).

$$Recall = \frac{TP}{TP + FN} \tag{7.2}$$

where the recall measures the fraction of positive examples that are correctly labeled and precision measures the fraction of positive examples which are truly positive. Finally, the weight of the hypothesis is used as a confidence measure to draw Precision-Recall (PR) curves [Sok06, Gou05]. In the PR space, the Recall is plotted on the x-axis and the precision on the y-axis.

Sample results of the proposed labeling process are depicted in Figure 7.2. It can be seen that the ground truth data comprises detailed topological information on lanes, roads and stopline locations. Information on lane turn directions are implicitly encoded by the structure of the road network. Further, detection results on the patch level are depicted in Figure 7.2, showing the nonparametric estimation of the marginal posterior distribution over all patch nodes which equals the product of all belief states over the patch variables in the proposed CHM. The unnormalized weights of the patch hypothesis are illustrated by showing colors on the temper-

ature scale, where black corresponds to low weights and white to high weights. The labeling results are depicted by showing green patches and black patches for positive labels and negative labels, respectively.

7.3 Road Detection Performance

This section presents the evaluation of the lane and road perception performance of the proposed hierarchical framework in road scenes with varying topology. Towards this goal, the proposed framework is applied to the database, where in particular the performance of lane and road topology estimation on highways, rural roads and urban roads is evaluated.

The following section are structured as follows: Section 7.3.1 specifies the employed CHM, the used inference algorithm and model assumptions, Section 7.3.2 evaluates the overall performance of applying the proposed framework to the database, Section 7.3.3 evaluates the benefits of depth-first message passing over standard message passing (breadth-first), Section 7.3.4 evaluates the benefit of using multiple low-level detectors and Section 7.3.5 investigates different message passing schedules for road detection.

7.3.1 Model Specification

During the experiments, a model is used that is composed of a set of hybrid CHMs (see Section 5.3.2). Each of these CHMs corresponds to one of the following road types: (1) two-lane rural road, (2) two-lane highway road, (3) three-lane highway road, (4) two-lane urban roads and (5) three-lane urban roads. For each of these scenarios a parallel lane configuration on the road-level is assumed. However, lanes are not restricted to be parallel. The spatial constraints are generally defined as proposed in Section 4.2. While the lane width is estimated during inference, the initial parameter ν_{w_p} of the spatial constraint in Equation 4.3 are specified to 3.5 m, 4.0 m and 3.2 m for rural roads, highway and urban roads, respectively. The patch length which defines the segmentation of the lane centerline is fixed to $\nu_{l_p} = 1$ m (see Section 4.2.2).

We assign a weighted sample to each detected low-level feature using the observation potential presented in Section 4.1.2. Subsequently, the belief over the patch nodes is computed using belief sharing as detailed in Section 4.3.2. Lane detection is performed using depth-first message passing with 25 individual sweeps (see Section 4.3.3). Consequently, the belief of each lane node is approximated by a maximum number of 25 weighted samples. During message passing, the lane hypotheses are allowed to grow recursively and therefore the model is not fixed to a specific length. However, a lane hypothesis is terminated if for six consecutive

lane elements no local evidence is present. Furthermore, lane hypotheses are restricted to a maximal length of 80 m on highways and on rural roads. In urban environments the maximum lane length is fixed to 40 m.

After computing the lane hypotheses, they are shared between different road-types, as proposed in Section 5.3.2. In the following, the weight of the road hypothesis is computed as proposed in Section 4.3.3, where on each level the acceptance rate is used to decide if an additional top down step is necessary or not. Since the number of plausible road hypotheses is expected to be relatively small, the belief over road nodes is approximated by only 5 samples.

In order to bound hypotheses to plausible domains, it has to be decided, when a hypothesis should be assigned to the outlier process of the observation potential in Equation 4.1. Towards this goal, a threshold that assigns those hypotheses to the outlier process which exceed the 3σ range of the observation potential is introduced.

7.3.2 Road Recognition Performance

A key benefit of the proposed framework is that each level of the CHM incorporates a priori scene knowledge comprised in the probabilistic constraints (see Section 5.3.1). The importance of this knowledge is evaluated by evaluating the posterior marginal distribution $p(\boldsymbol{x}_i^p|\boldsymbol{b})$ or belief $b_i(\boldsymbol{x}_i^p)$ over the patch locations at different stages of message passing. Therefore, three different message passing algorithms are used.

Algorithm 1 The bottom up belief $b_i^-(\boldsymbol{x}_i^p)$ over the patch nodes is computed by fusing messages received from their associated feature nodes $\boldsymbol{x}_j^f, j \in \Xi(i)$.

Algorithm 2 The bottom up belief of both the lane nodes $b_i^-(\boldsymbol{x}_i^l)$ and patch nodes $b_i^-(\boldsymbol{x}_i^p)$ is computed. Then, the belief of the patches $b_i^{\setminus r}(\boldsymbol{x}_i^p)$ is computed by propagating messages down from the lanes to the patches. Here $\setminus r$ denotes that the road level of the model is not processed.

Algorithm 3 The belief $b_i(\boldsymbol{x}_i^p)$ is computed by passing message from the leaves (feature nodes) to the root (road nodes) and back down.

Each of the above algorithms incorporates a different amount of a priori scene knowledge in the detection results. More precisely, the bottom up belief state $b_i^-(\boldsymbol{x}_i^p)$ only comprises knowledge on local evidence and the expected lane width. The belief $b_i^{\setminus r}(\boldsymbol{x}_i^p)$ represents the belief over patches in the context of lanes and hence contains knowledge on the longitudinal lane model. Finally, the belief $b_i(\boldsymbol{x}_i^p)$ represents the belief over patches in the context of both lanes and roads which imposes constraints on both the longitudinal and the lateral road model.

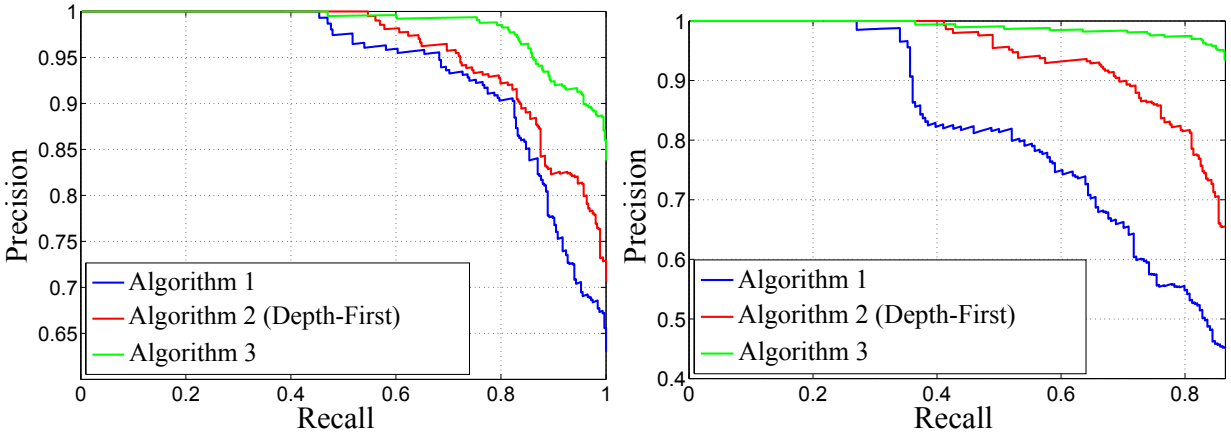


Figure 7.3: Precision-Recall curves for highway scenarios (a) and for rural scenarios (b). Results are computed using different algorithms and evaluated against the ground truth. For details see text.

Figure 7.3 shows, the results of applying the above algorithms to the highways and rural test frames of the dataset, while relying only on lane marking cues. As can be seen, the performance of the proposed approach increases drastically, as additional contextual information are incorporated. This can be explained by the fact that Algorithm 1 performs inference over a relatively small area. Accordingly, the results strongly rely on the presence of local evidence,i.e., missing, occluded or damage lane markings have a significant influence on the recognition performance.

Algorithm 2, on the other hand, fuses evidence from a larger area. Accordingly, the results are not as affected by missing local evidence as in Algorithm 1. Further, Algorithm 2 benefits from proposed approach for occlusion handling, as presented in Section 4.3.1. In fact, the bottom up belief of the patches $b_i^-(\boldsymbol{x}_i^p)$ is likely to contain many false-negatives, due to the absence of both the left and/or the right boundary feature. However, during lane detection, the inference algorithm allows for up to 6 missing patches which allows to detect partly occluded lanes.

Finally, the results obtained by Algorithm 3 (see Figure 7.3) clarify the importance of the road levels which impose constraints on the lateral lane structure. These constraint ensure the overall compatibility of neighboring lanes, patches and features.

Sample results of applying the different algorithms to the dataset are depicted in Figure 7.4 . This figure shows lane detection results in challenging scenarios with partly occluded lanes (left), splitting lanes (middle) and complex intersection topology (right). Note that lane hypotheses outside the lane markings are supported by marking features on one side and by the outlier process on the other side (see Section 4.3.2). The existence of these hypotheses is allowed for, since they

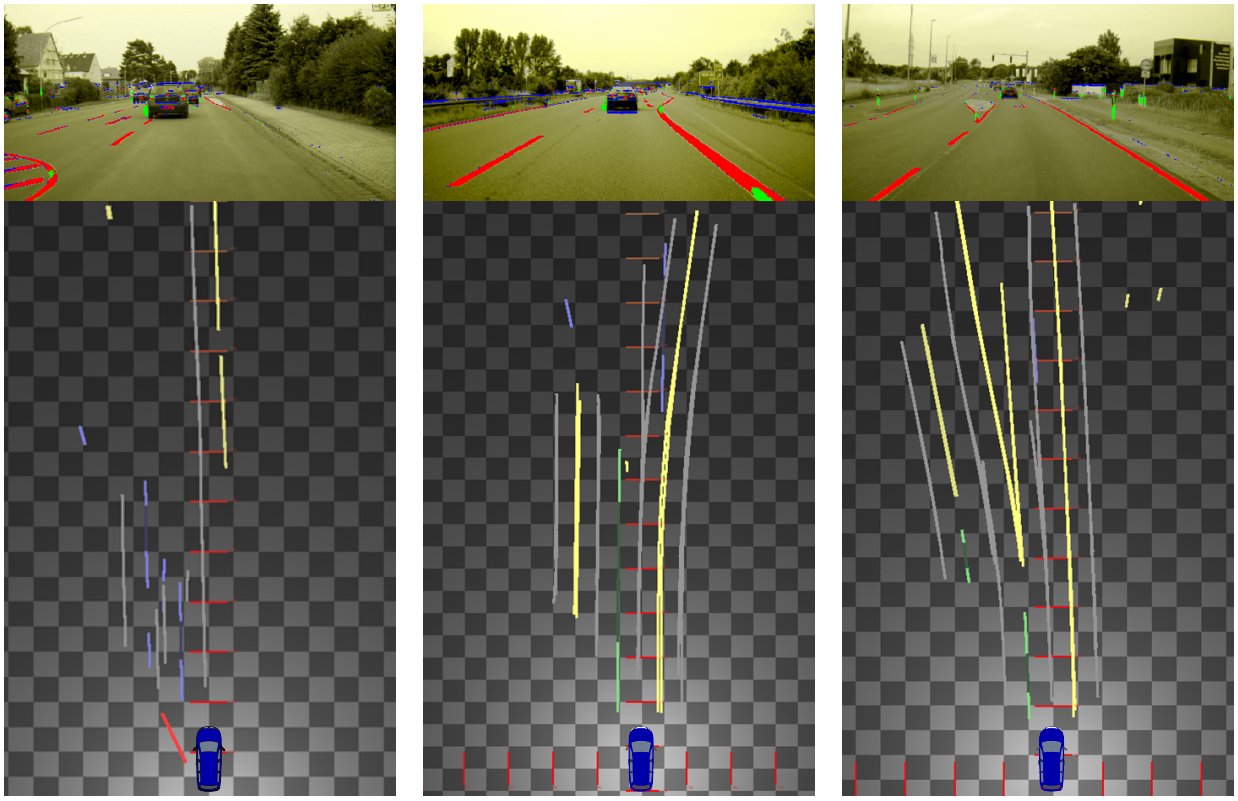


Figure 7.4: Sample results of lane topology estimation. (Top) Detected lane marking features (red). (Bottom) Lane marking features in the BEV (yellow, purple, green) and results of lane hypotheses generation (grey).

may become plausible in the context of roads. However, given the lack of low-level evidence their corresponding weight is relatively small (see Equation 4.15).

7.3.3 Depth-First and Breadth-First Message Passing

A key aspects of the proposed framework is depth-first message passing for lane detection [Töp13] presented in Section 4.3.3. Depth-first message passing is expected to require significantly lower run time than breadth-first message passing, while achieving a comparable performance.

To test this hypothesis, lane and road perception is performed by applying both depth-first message passing and breadth-first message passing to the highway and rural test scenarios of the dataset. To avoid an exponential growth of the lane hypotheses using breadth-first message passing, a resampling is performed after computing the belief update on the lane-levels. This optional step of Nonparametric Belief Propagation (NBP) is used to limit the number of lane samples to 150.

It can be seen in Figure 7.5 that depth-first message passing out-performs standard breadth-first message passing over the complete range of confidence, while

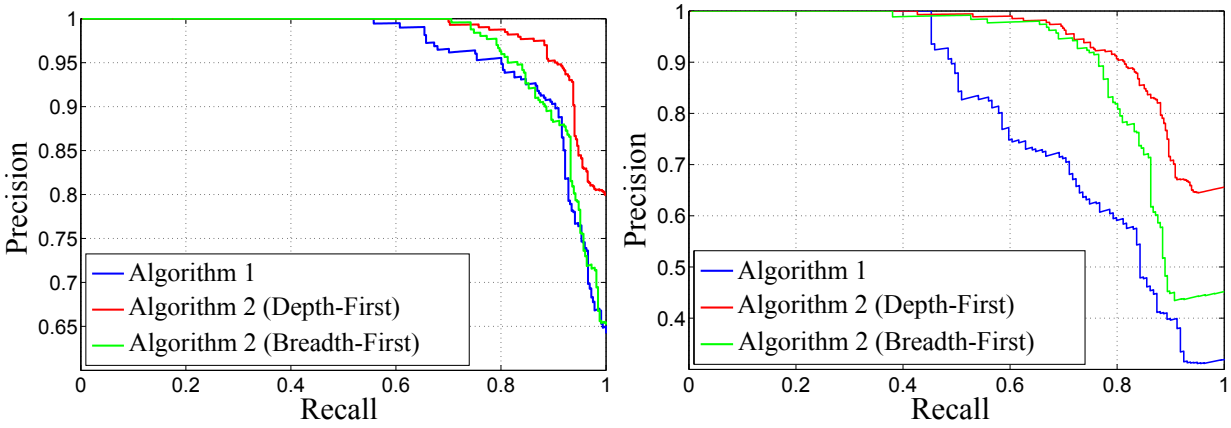


Figure 7.5: Results of lane detection for highway scenarios (a) and for rural scenarios (b) using depth-first message passing (red) and breadth-first message passing (green). In both scenarios depth-first message passing (25 samples) shows more promising results than breadth-first message passing (150 samples).

	Alg. 1	Alg. 2 (BF)	Alg. 2 (DF)	Alg. 3
time (ms)	1.47	75.29	4.41	5.38
RMS (m)	0.12	0.25	0.20	0.23

Table 7.1: Runtime and location errors of lane detection using depth-first and breadth-first message passing. Depth-first message passing allows for real-time performance, since patches, lanes and roads can be computed within a single sensor frame (frame rate 20fps).

using a significantly reduced number of only 25 samples. The reason for this major improvement is that applying depth-first message passing increases the probability to first propagate those low-level hypotheses which are likely to be part of valid high-level hypotheses. Consequently, during messages passing, less invalid hypotheses than using breadth-first message passing are propagated. In addition, Table 7.1 shows that applying depth-first message passing significantly reduces the required runtime, while achieving a similar geometrical accuracy.

Illustrative results of applying both breadth-first message passing and depth-first message passing are depicted in Figure 7.6, showing the large amount of hypotheses computed during breadth-first message passing (left), and the few likely hypotheses computed during depth-first message passing (right).

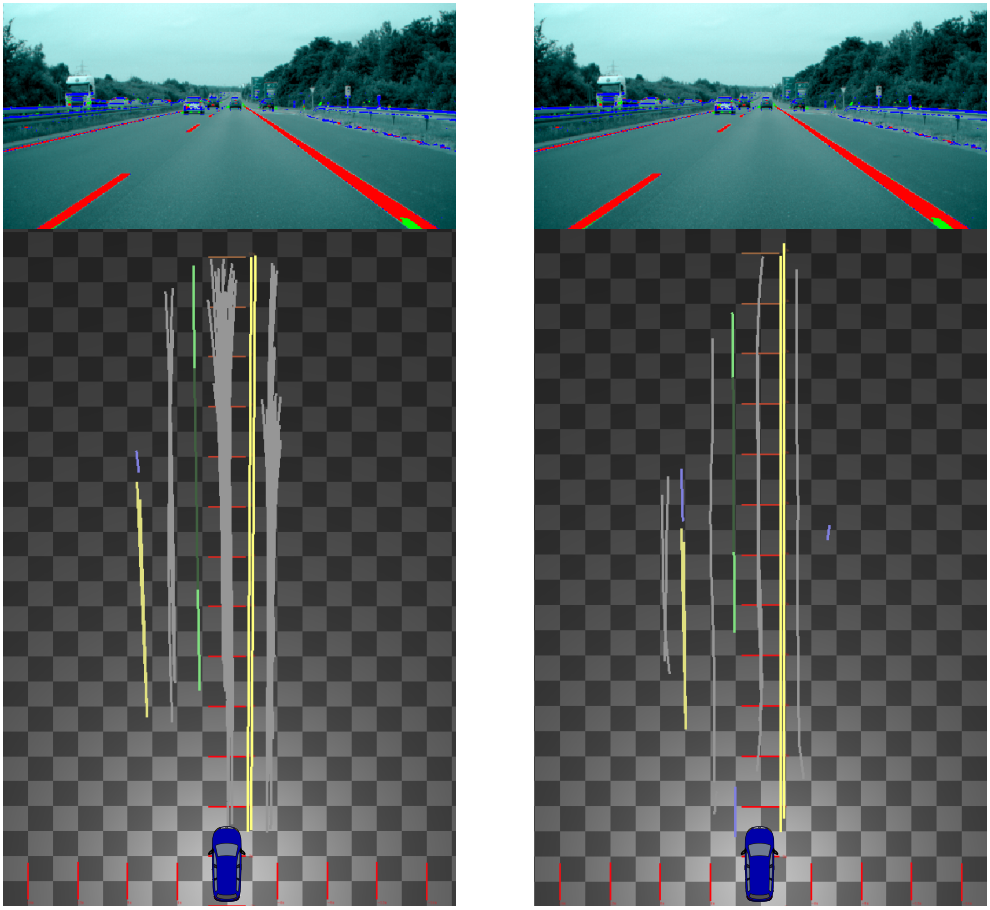
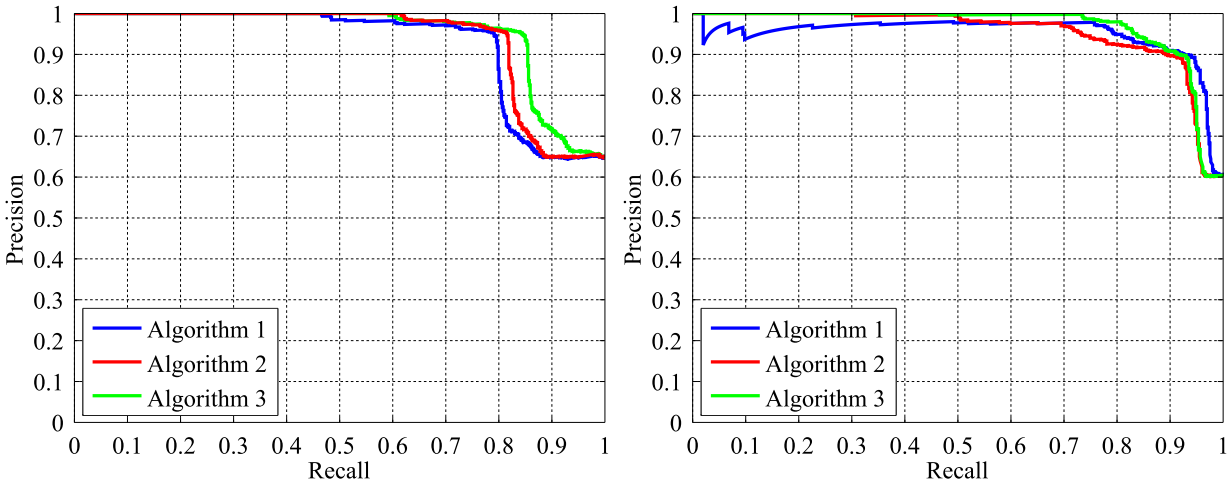


Figure 7.6: Results of lane topology estimation. (Top) Detected lane marking features (red). (Bottom) Results using breadth-first message passing (left) and depth-first message passing (right). Lane hypotheses are shown in grey.

7.3.4 Multi-Cue Road Detection

A key benefit of the proposed hierarchical framework is that it allows for the incorporation of multiple lane and road boundary cues and fuses them in an intelligent way. Particularly, in semi or unstructured urban environments using multiple cues is expected to lead to an increased performance. This section presents the results of evaluating the benefit of incorporation multiple lane and road boundary cues based on the urban scenarios contained in the database. In this experiment, the recognition performance achieved using only lane marking cues is compared to the results obtained using both lane marking and the road edge cues.

The results of these experiments are depicted in Figure 7.7, showing that, as expected, the additional usage of road edge cues improves the recognition performance, since in many urban scenarios lane markings are not reliable. It can be seen that using both cues, a precision of about 90% up to a recall of 90-95% can be obtained, while precision drops drastically for a recall higher than 80-85% us-



(a) single cue setup (b) multi-cue setup

Figure 7.7: Results of urban multi-cue lane detection. (a) When relying on lane marking features the precision drops rapidly for a recall higher than 80-85%. (b) Using both lane marking and road edge features allows to obtain a precision higher than 90% up to a recall of 90-95%. In (b) the precision decreases slowly for a recall between 70-95%, since the second detector also causes additional false positives.

ing only lane marking cues. However, using multiple low-level cues also leads to additional computational complexity. In fact, the average computational time for the single cue setup is 5.38 ms, while for the multi-cue setup processing requires 7.63 ms.

Figure 7.8 shows the detection results for a multi-lane urban road using the two lane boundary cues.

7.3.5 Message Passing Schedule for Road Detection

In the previous section, inference is performed according to the depth-first message passing schedule (Section 4.3.3) and roads are detected by proposing and evaluating additional lane hypotheses based on low-level observations. This approach is expected to be ideal in means of robustness. However, the search for corresponding low-level evidence is computational expensive. One possibility to bootstrap inference, is to always accept lane hypotheses and thus avoid the search for low-level evidence on the feature level which is comparable to the lane and road perception approach presented in [Hur13].

Figure 7.9 summarizes the results obtained by applying both schedules to the urban scenarios of the dataset. The results show that the additional top down step which searches for corresponding low-level evidence, increases the recognition performance. This result can be explained by the fact that terminating the top

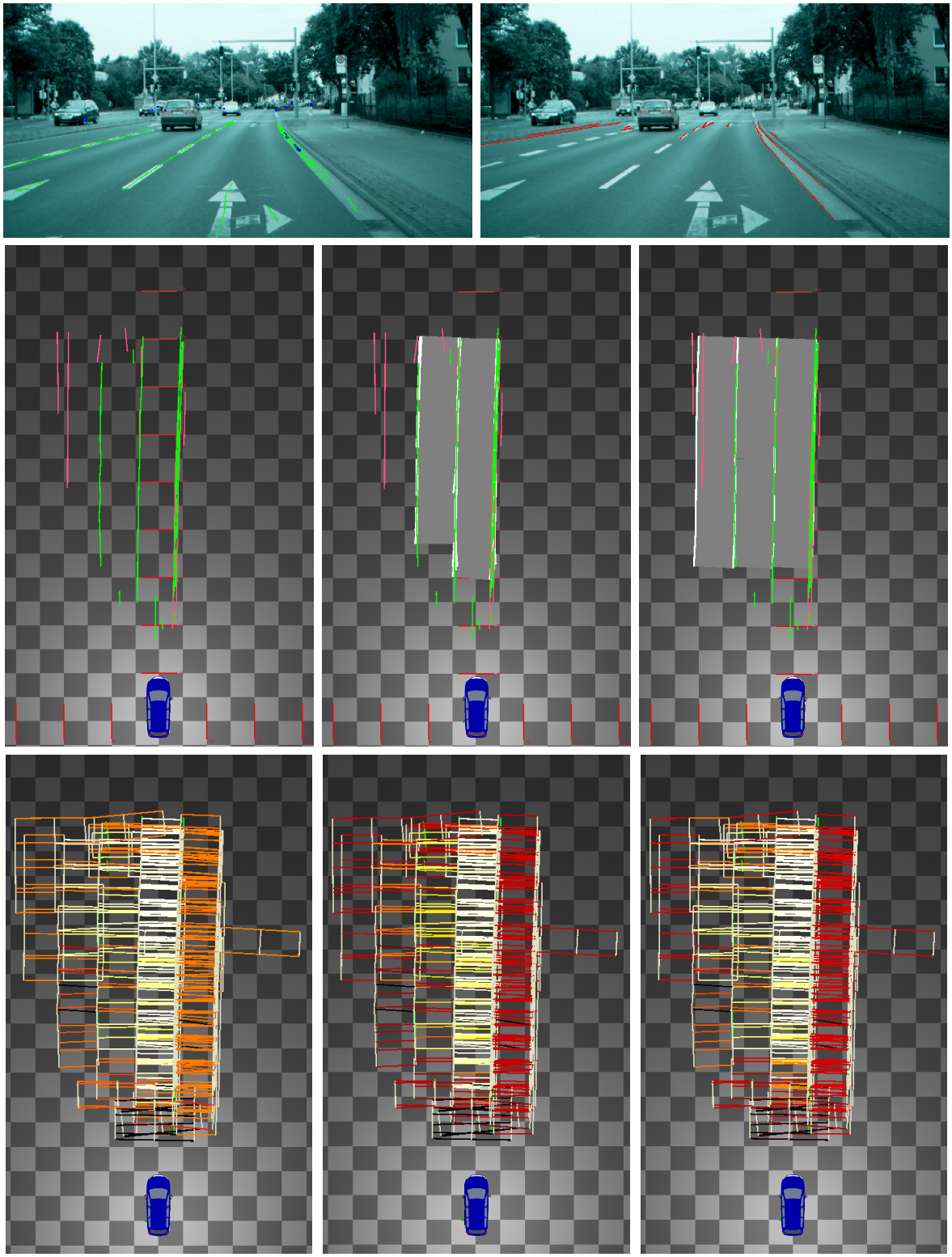


Figure 7.8: Results of multi-cue lane detection in urban scenarios. (Top) Feature extraction result in the image domain. (Middle) Low-level features in the vehicle coordinate frame, lane detection results and road hypotheses (left to right). (Bottom) Joint probability distribution at different stages of inference: Algorithm 1, Algorithm 2 and Algorithm 3 (left to right).

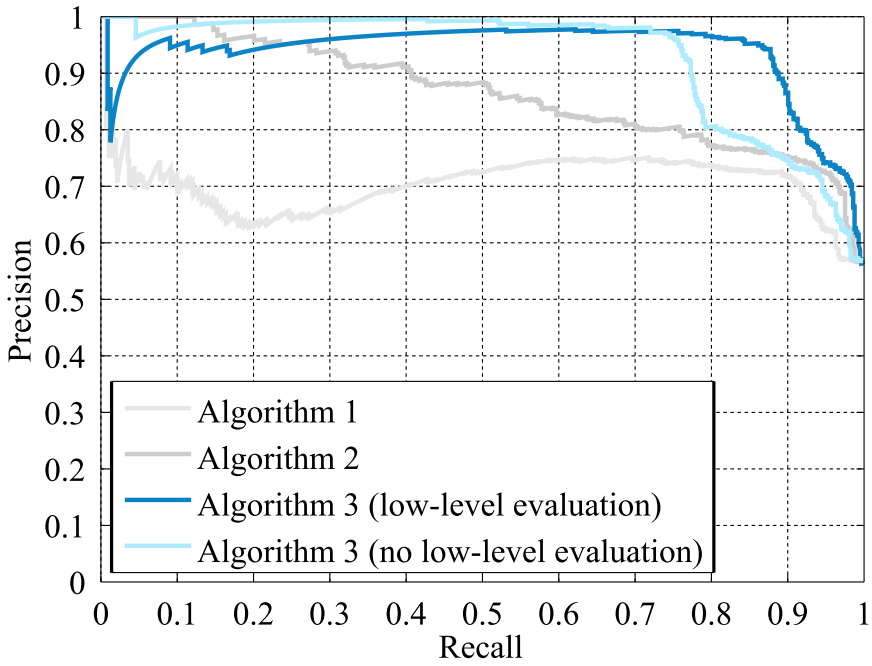


Figure 7.9: Results of road detection using different message passing schedules. (light-blue) Results of road detection, where top down validation is terminated on the lane-level. (blue) Results of performing complete top down validation including low-level features.

down search on the lane-level implies that the belief over the lane nodes in the CHM is well approximated. However, during the experiments many lane samples converge to the lane of the ego-vehicle since it is supported by the majority of the sensory evidence. An example is shown in Figure 7.8, where most of the 25 lane samples correspond to the ego-lane or its direct neighbor, while the most left lane is not represented. The proposed approach overcomes this issue by performing the complete top down search for local evidence based on the structure of the proposed CHM.

The drawback of the latter approach is its additional computational complexity. However, the additional time required for the evaluation of each road sample including the low-level features is only 0.2 ms. Considering that often a small number of road samples is sufficient to obtain reliable results the total computational complexity still fulfills real-time requirements (see Table 7.1).

7.4 Lane and Road Perception at Intersections

This section presents the results of evaluating the performance of stopline detection and lane turn direction estimation of the hybrid compositional hierarchical framework proposed in Chapter 6. The following experiments are based on the

hybrid CHM described in Section 6.2, where both the model parameters and the inference algorithm follows the specification given in Section 7.1. The exception is the patch detection which is performed as shown in Figure 6.7. The feature set used for the following experiments comprises lane marking and road edge features used to detect the road topology as well as detections of stoplines and turn arrows (see Section 6.1). Further, the experiments are limited to 20 sequences of intersection approaches with an individual length of 15 to 40 seconds. The sequences stem from 16 intersections roads with varying lane structure and different combinations of lane turn directions.

7.4.1 Hierarchical Stopline Detection

In Section 7.3, it is shown that the spatial constraints imposed by the different hierarchical levels of the model increase the robustness of lane and road perception. Similarly, the constraints on the location of stoplines are expected to improve the performance of stopline detection (see Section 6.2). To verify this hypothesis, the following experiment is performed.

We begin with computing three intermediate stopline detection results, each incorporating a different amount of constraints in the detection process.

Algorithm 4 The bottom up belief $b_i^{-}(\boldsymbol{x}_i^p, \boldsymbol{d}_i^{p_c} = d_i^{stop})$ of the stop-patches which comprises the local evidence of boundary features and the stopline features is computed.

Algorithm 5 The belief $b_i^{\setminus r}(\boldsymbol{x}_i^p, \boldsymbol{d}_i^{p_c} = d_i^{stop})$ over the patch nodes is computed without processing the road nodes.

Algorithm 6 The belief $b_i(\boldsymbol{x}_i^p, \boldsymbol{d}_i^{p_c} = d_i^{stop})$ is computed including the road nodes.

Here $\boldsymbol{d}_i^{p_c} = d_i^{stop}$ specifies that only the stop-patches are considered in the experiment. In this case, the importance weight of a patch is scaled by the stopline probability. This means that the importance weight of a patch sample encodes both its spatial plausibility in respect to the low-level features and the probability of a stopline at the patch location. After computing the importance weights, the ground truth and the overlap criterion introduced in Section 7.2 are used to label the detected patches. Note that, since all patch samples comprises a discrete center type variable, each detected patch is considered as a potential stop-patch.

Figure 7.10 depicts PR curves summarizing the results of the conducted experiment. As evidenced by this figure, the spatial and semantic constraints increase the detection performance significantly. With bottom up stopline detection (Algorithm 4), a precision close to 100% for a recall up to 50-55% is obtained. However, precision drops rapidly above this level of recall. More promising are the results

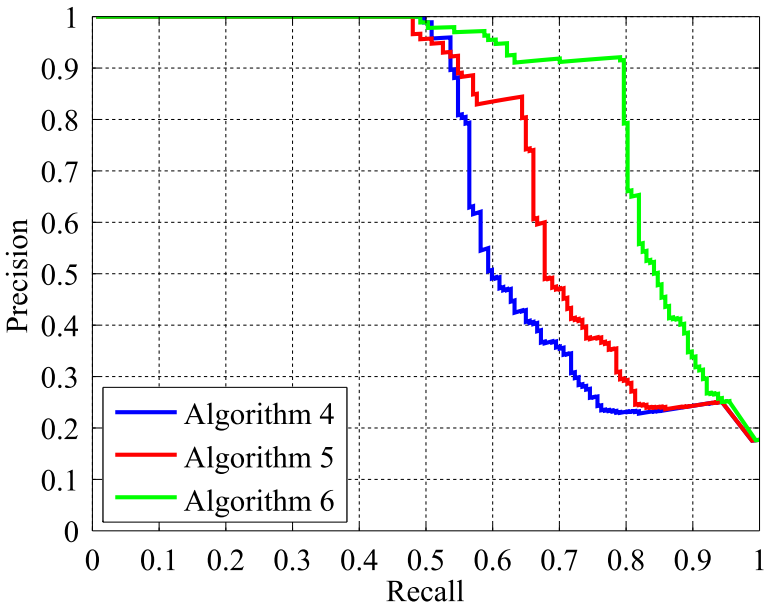


Figure 7.10: Stopline detection performance at different stages of message passing. (blue) Bottom up belief state over patch nodes $b_i^{-}(\boldsymbol{x}_i^p, \boldsymbol{d}_i^{p_c} = d_i^{stop})$. (red) Belief $b_i^{\setminus r}(\boldsymbol{x}_i^p, \boldsymbol{d}_i^{p_c} = d_i^{stop})$ over the patch nodes without processing the road levels. (green) Belief $b_i(\boldsymbol{x}_i^p, \boldsymbol{d}_i^{p_c} = d_i^{stop})$ after completing message passing.

of Algorithm 5 and Algorithm 6. In fact, Algorithm 5 allows to obtain a precision higher than 80% for a recall up to 65%. This suggests that constraining the stop-patches to be part of lanes efficiently reduces the number of false positives. An additional performance gain can be obtained by processing the road levels (Algorithm 6). This clarifies the importance of the road-level constraints which constrain stoplines to be part of neighboring lanes and thus allow to maintain a precision of higher than 90% up to a recall of 80%. Further, Figure 7.10 shows how each level of the hierarchy improves the recall which is achieved by performing the expectation-based search for low-level stopline features at each level introduced in Section 6.3.2.

The benefits of including the knowledge of the different levels into the detection process is also illustrated in Figure 7.11. This figure shows the results of lane and road detection (middle) based on low-level features (middle-left) extracted from the visual input (top). The bottom row illustrates how the knowledge of the different levels allows to constrain the stop-patch hypotheses to the actual stopline locations which is shown by high weighted samples close to the stoplines (bottom) and the well aligned stoplines on the road level.

Message Passing Schedule: Performing sequential message passing in a multi-cue model complicates the choice of the message passing schedule. In the pro-

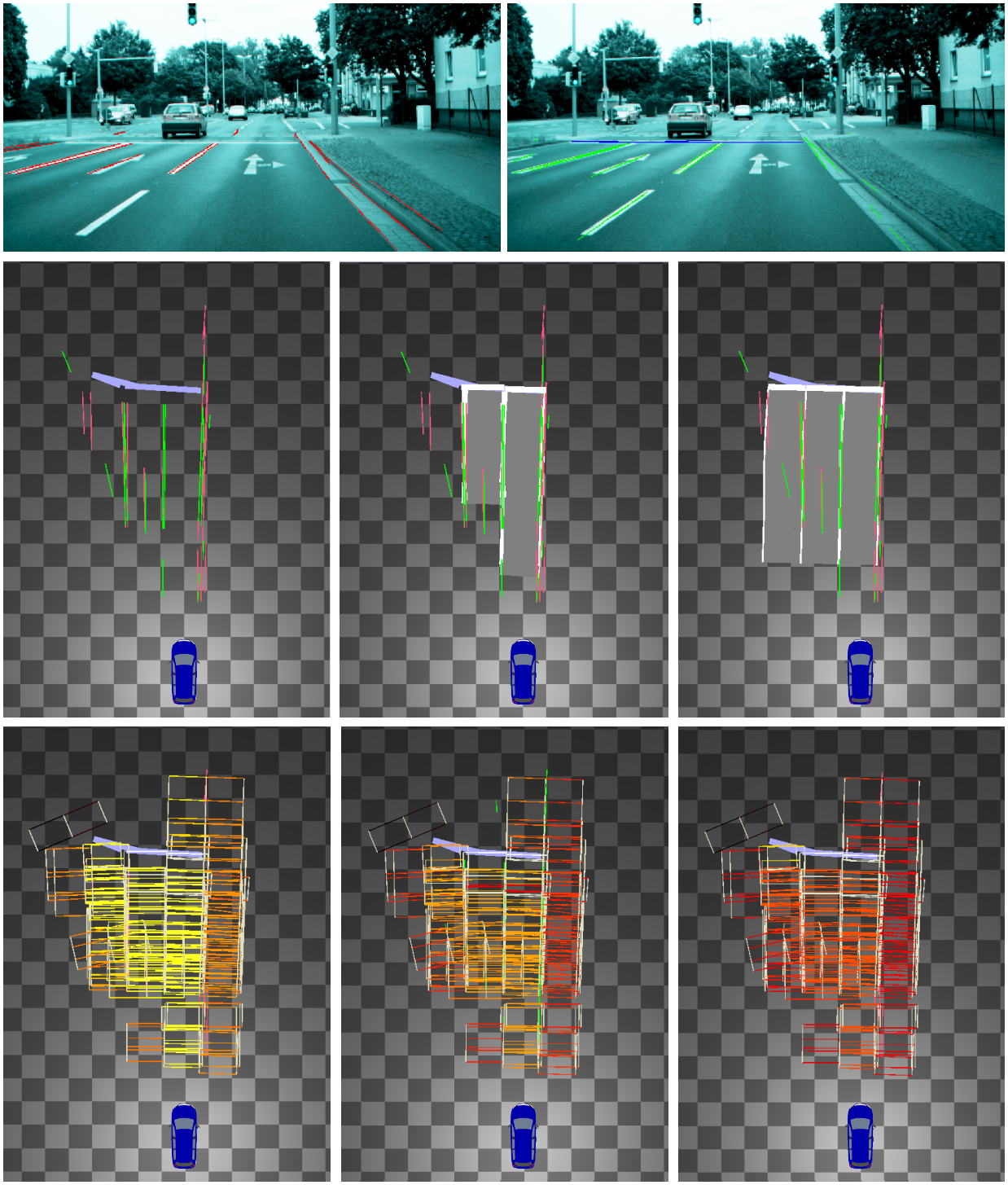
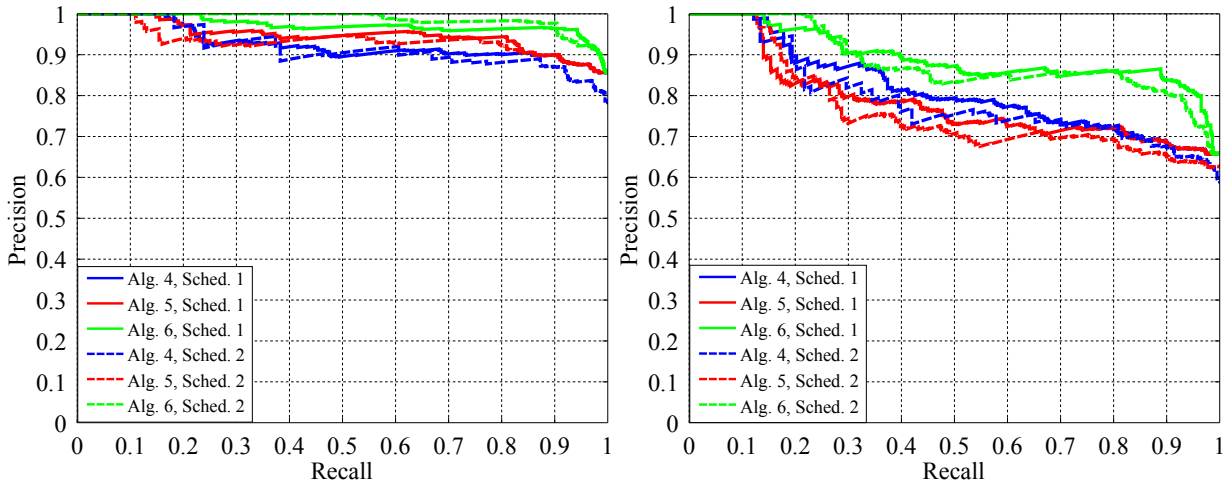


Figure 7.11: Stopline detection results with varying amount of topological knowledge. (top) Results of road edge detection (red), lane marking detection (green) and stopline detection (blue). (middle) Feature detection results in the vehicle coordinate frame, results on the lane level and results on the road level. (bottom) belief approximation results over all patch variables obtained by Alg. 4, Alg. 5 and Alg. 6. Note that the distance to the closest stopline is approximately 25 m.



(a) 2 FP/Frame

(b) 5 FP/Frame

(c) 10 FP/Frame

(d) 20 FP/Frame

Figure 7.12: Results of the hierarchical stopline recognition for different message passing schedules and varying stopline detection performances. As the number of false positives per frame increases Schedule 1 dominates the PR curves.

posed CHM for intersection roads this gives rise to the two following message passing schedules for stop-patch detection presented in Section 6.3.2.

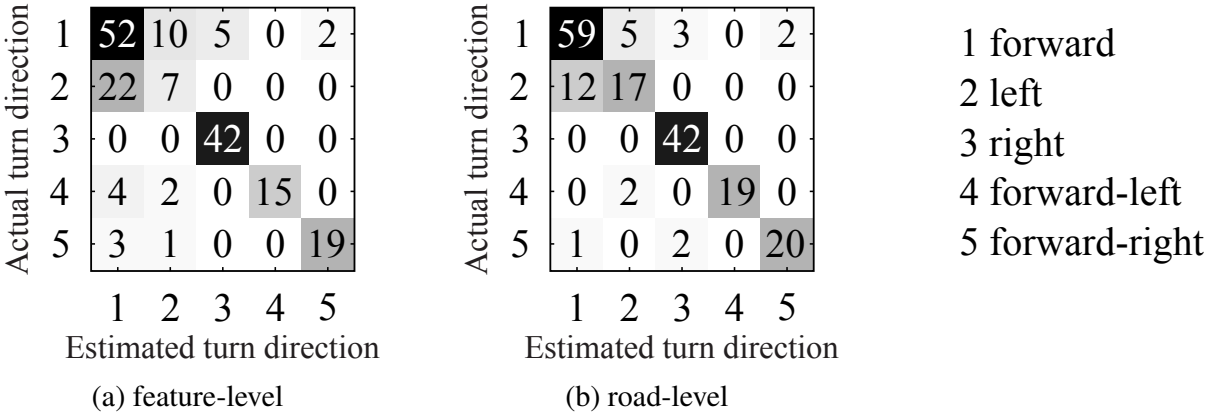
Schedule 1 First stop-patch hypotheses are proposed based on previously detected patches. Then, a top down/bottom up sweep is performed, in which corresponding evidence provided by the stopline detector (see Figure 6.7a) is incorporated

Schedule 2 Stop-patch hypotheses are initialized based on low-level stopline observations. Subsequently, the stop-patch hypotheses are aligned to the low-level lane boundary evidence as depicted in Figure 6.7b.

Generally, it can be expect that the optimal message passing schedule strongly depends on the performance of the employed low-level feature extraction approaches. More precisely, Schedule 2 is expected to be optimal if the stopline detector is very reliable, since it limits the influence of false positive in the lane boundary detection results on the stopline detection performance. In contrast, Schedule 1 is expected to be optimal if the stopline detection results comprise a high number of false positive detections.

This hypothesis is verified by performing the following experiment. To simulate different low-level detection performances, false positives detections are added to the stopline detection results. These additional features are created by randomly creating stopline features in the field of view and adding a Gaussian random noise to each feature parameter. Then, Schedule 1 and Schedule 2 are applied to the input data and their results are compared.

The results of this experiment are depicted in Figure 7.12, showing PR curves. As expected, Schedule 2 leads to comparable lower performance as the false detection rate of the stopline detector increases. On the other hand Schedule 1 allows to maintain a relatively good performance for all setups. For some of the settings the results on the lane-level are counter-intuitive, for example in Figure 7.12d the precision for Algorithm 5 drops rapidly for a recall higher than 10%, while Algorithm 4 retains a relatively high precision. These results may be explained by the presence of many false positive detections which are well aligned to the actual course of lanes in the scene. Even though the acceptance rate introduced in Section 4.3.3 allows to model such spatial uncertainties, if false positives are well aligned to the course of the lane the distinction between true positives and false positives becomes challenging. However, it can be seen in Figure 7.12d that even in such challenging scenarios the additional constraints imposed by the road-level allow to obtain a comparable good performance.



(a) feature-level (b) road-level

Figure 7.13: Turn direction confusion matrices. (a) Confusion matrix for the results of turn arrow recognition, with estimated turn direction and true turn direction at the x- and y-axis, respectively. (b) Results of lane turn direction recognition using the proposed framework.

7.4.2 Hierarchical Estimation of Lane-Turn Directions

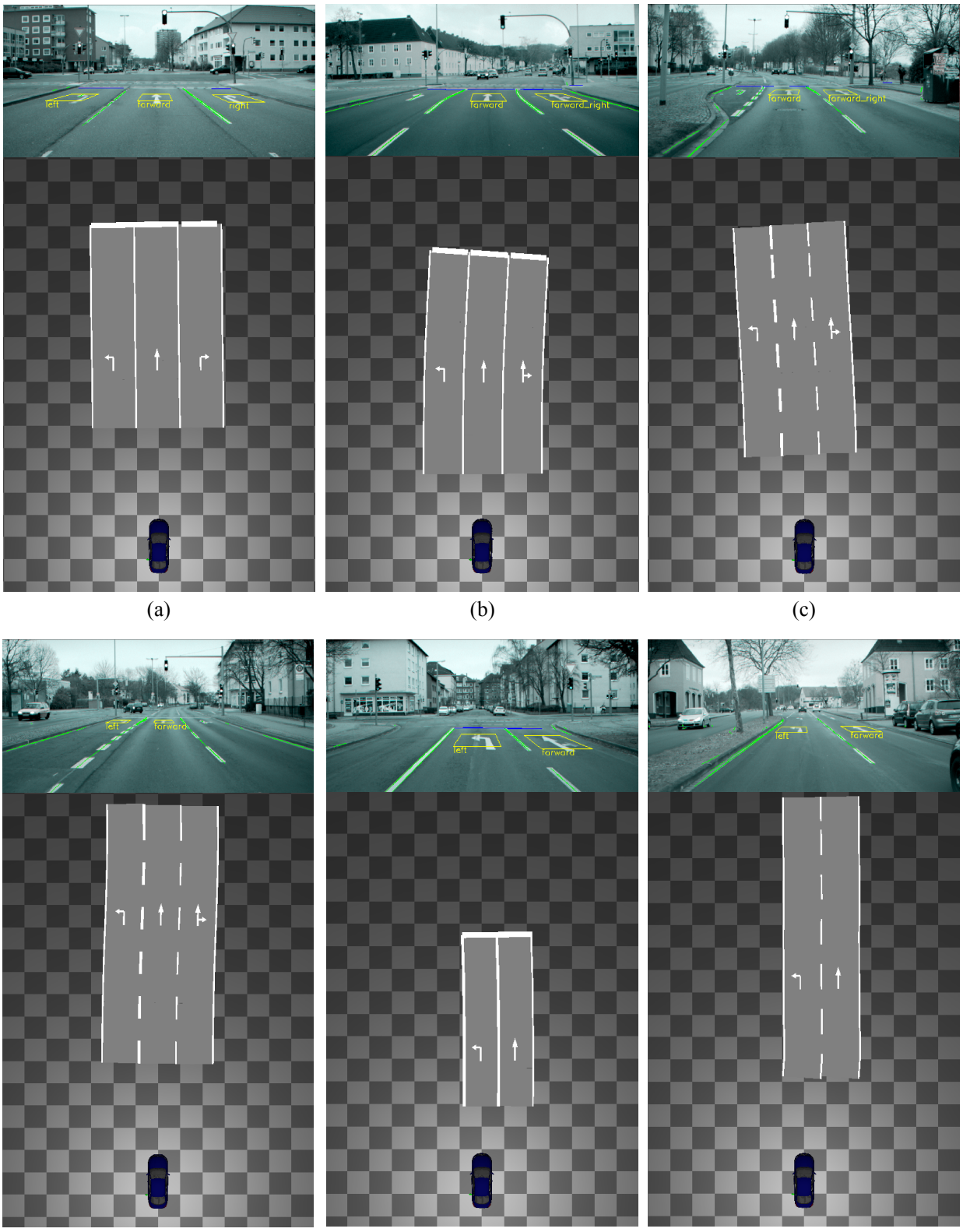
In this section, the performance of lane turn direction estimation is evaluated. Towards this goal, the detection results on the feature-level are compared with the results obtained by performing a complete bottom up/top down message passing procedure in the proposed framework. The major difference to the previous section is that not a predefined type is of interest but the maximum a posteriori probability (MAP) estimate of the turn direction

$$d_i^{f_c} = \arg\max_{\boldsymbol{d}_i^{fc}} p(\boldsymbol{d}_i^{f_c}|\boldsymbol{c}). \tag{7.3}$$

Hence, only the most likely lane turn direction is compared with the ground truth. The performance of the proposed approach is then quantified using confusion matrices.

The results of the experiments are depicted in Figure 7.13. As evidenced by this figure, a large performance gain can be achieved by combining the lane-level detection results on the road-level, since they restrict the results to plausible lane turn direction constellations. Note that, the effect of the patch-level and the lane-level constraints is not explicitly investigated. This is mainly due to the fact that neither lanes nor patches impose constraints on the turn directions.

Figure 7.14 show exemplary result of urban multi-lane road recognition on the road-level. As Figure 7.14a shows, the proposed framework allows to jointly recognize the road topology, the turn direction of individual lanes, the location of stoplines and the type of lane boundaries. Further, Figure 7.14b, c and d indicate that by combining arrow detections of neighboring lanes on the road level the



(a) (b) (c)

(d) (e) (f)

Figure 7.14: Sample results of joint road topology, lane turn direction and stopline recognition.

Module	Average Runtime/Frame
Lane marking detection	1.08 ms
Road edge detection	1.78 ms
Stopline detection	2.67 ms
Turn arrow detection	3.20 ms
Patch detection (Section 4.3.1)	2.47 ms
Lane detection (Section 4.3.3)	4.92 ms
Road detection (Section 5.3.2)	7.63 ms
Patch detection (Section 6.2.1)	3.41 ms
Lane detection (Section 6.2.2)	6.87 ms
Road detection (Section 6.2.3)	9.05 ms

Table 7.2: Running times per frame. The table show average running times of the individual modules of the C++ implementation on an Intel(R) Core(TM) i5-3427U CPU with 1.80 GHz using the ADTF framework. The first part of the table shows the time needed for computing the feature extraction. The second part lists the timings for computing patches, 25 lane samples and 5 road samples, using the CHM presented in Chapter 4. The third part shows the runtime for the hybrid CHM for intersection roads.

proposed framework obtains reliable results even if single arrows have not been detected. Figure 7.14e shows a scene where the classification result on the feature level also lead to wrong results on the road level. In this case, the turn direction constrains imposed by the proposed model do not have a positive effect on the results, since the detected turn directions are plausible.

7.5 Runtime

In this section the computational complexity of the proposed approach is evaluated. Towards this goal, the running time of the C++ implementation of the proposed framework is measured, while applying it to the intersection scenarios of the database. While the results listed in Table 7.1 already indicate that the proposed approach runs in real-time (<50.00 ms) this evaluation provides an overview on the runtime demands of the various modules of the implementation. Table 7.2 lists the average running times of the various procedures of the algorithm, separated into feature extraction as well as patch, lane and road detection. In order to evaluate the additional runtime demands of the hybrid CHM for intersection roads as presented

in Chapter 6, its runtime is compared with the CHM presented in Chapter 4.

8 Conclusion

8.1 Contributions

This thesis presents a novel hierarchical probabilistic framework that outperforms existing algorithms (in terms of computational efficiency and scalability) for reliably perception of lanes and roads in real-time and in the presence of uncertainty (including partial occlusion and error prone sensory evidence). This thesis presented novel algorithms to address these issues:

- The hierarchical representation of roads and lanes allows to infer complex lane and road topologies from low-level sensory evidence. The presented compositional hierarchical model provides not only a unified probabilistic formulation of the addressed lane and road perception task, but also allows to incorporate a priori scene knowledge which increases the robustness of the proposed approach in the presence of clutter and partial occlusions. Thereby, the proposed road model is compositional and generic in the sense that it does not impose any hard constraints on the lane geometry as imposed by e.g., clothoids or splines. Instead, a priori expectations on the lane geometry are expressed through weak probabilistic constraints, and lanes are assembled from a large number of lane patches.

- The hybrid hierarchical framework allows to simultaneously infer complex non-linear road topologies and lane semantics (e.g., turn direction and stopline positions). Whereas it provides a unified probabilistic formulation that provides an enormous expressive power.

- The proposed framework accounts for the topological diversity of target scenarios by formulating a hierarchical multi-scenario model. A key component of this extended model is part-sharing [Zhu10, Spe11] which allows to benefit from the similarities of different road topologies leading to the much desired computational efficiency.

- Sequential depth-first message passing for lane and road perception [Töp13, Spe13] takes into account the inherent structure of the lane and road perception problem. Thus it allows for both real-time computations and the detection of lanes and roads in areas of low belief.

Most importantly, this thesis includes the first application of hybrid compositional hierarchical models for holistic lane and road perception at urban intersections including the detection of non-linear road topologies, lane turn direction and stopline locations. One of the key innovation of this hybrid framework is that it formulates the task of lane and road perception as the problem of performing inference in a probabilistic graphical model. This allows to not only avoid early decision, but also to fuse the evidence of multiple low-level sensory sources in an intelligent way. Thus, it is possible to obtain robust results in challenging real world scenarios, where a single sensory source is not reliable.

For all the above aspects, it is shown that both qualitatively and quantitatively the proposed hierarchical framework archives reliable results, while attaining real-time performance.

8.2 Future Directions

While the previous chapter present promising results, there are a number of possible improvements and promising lines of future research.

8.2.1 Tracking

This thesis focuses on the task of inferring high-level scene knowledge from low-level detection for a single frame, as it is the key benefit of the proposed framework. However, the presented model can easily be extended to incorporate temporal consistency. The most direct way of extending the model is to chain the random variables of the proposed model along time which is a relatively standard undirected variant of the Hidden Markov Model (HMM) of order one [Bis06, BS04, Thr05], as shown in Figure 8.1. The resulting spatiotemporal model is expected to increase the robustness of the results by incorporating temporal consistency. The additional spatiotemporal constraints $\psi^t(\boldsymbol{x}_{i,t}, \boldsymbol{x}_{i,t-1})$ encode the dependencies between random variables at time $t-1$ and t. Then, these constraints can be used to construct additional message sent from node $\boldsymbol{x}_{i,t-1}$ to node $\boldsymbol{x}_{i,t}$ which correspond to the prediction

$$p(\boldsymbol{x}_{i,t}|y_{1:t-1}) = \int \psi^T(\boldsymbol{x}_{i,t}, \boldsymbol{x}_{i,t-1}) p(\boldsymbol{x}_{i,t-1}|y_{1:t-1}) d\boldsymbol{x}_{i,t-1} \tag{8.1}$$

where the spatiotemporal constrain $\psi^T(\boldsymbol{x}_{i,t}, \boldsymbol{x}_{i,t-1})$ models the motion of the mobile platform. A convenient choice is to use a Gaussian model

$$\psi^T(\boldsymbol{x}_{i,t}, \boldsymbol{x}_{i,t-1}) = \mathcal{N}(\boldsymbol{x}_{i,t}, \varsigma(\boldsymbol{x}_{i,t-1}), \Lambda_{i,t})) \tag{8.2}$$

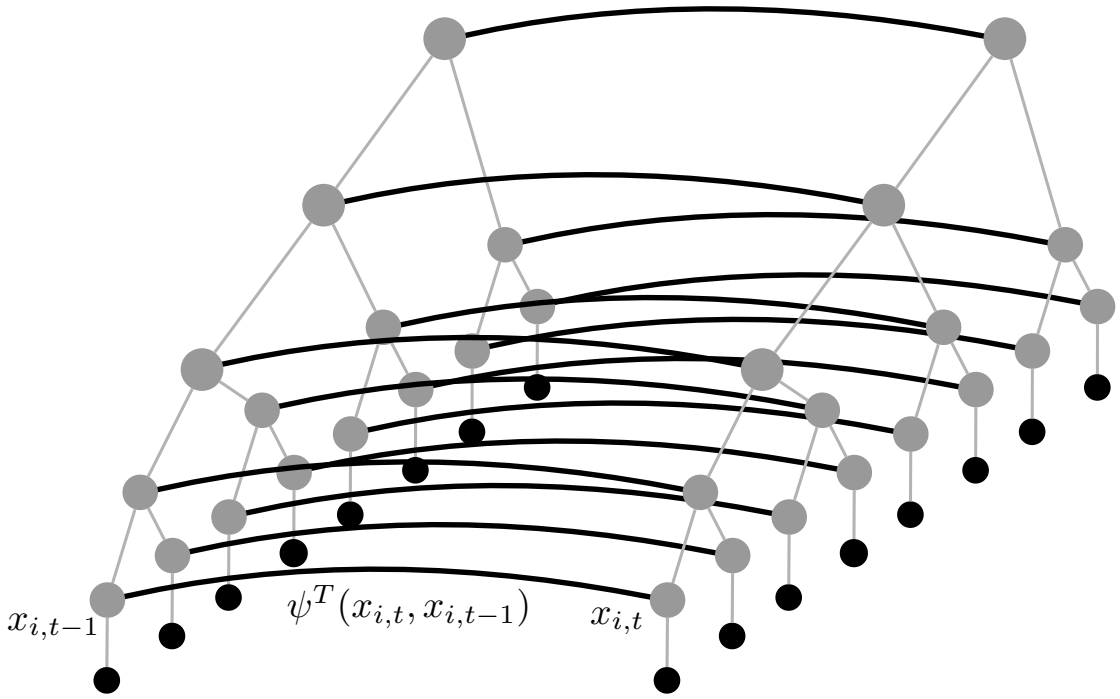


Figure 8.1: HMM of a compositional hierarchical lane model. The constraint $\psi^T(\boldsymbol{x}_{i,t}, \boldsymbol{x}_{i,t-1})$ models the spatiotemporal dependency between variables at time $t-1$ and t.

where the function $\varsigma(\boldsymbol{x}_{i,t-1})$ returns the predicted location of part $\boldsymbol{x}_i$ at time t based on the ego motion, and the covariance matrix $\Lambda_{i,t}$ models the motion uncertainty.

The above prediction can then be used as a proposal function and tracking can be performed as in the standard particle filter [Isa98a]. Alternatively, the prediction could be used as an additional input message for the belief update of node $\boldsymbol{x}_{i,t}$, as proposed in [Spe13, Sig08, Sud04b]. Particularly, interesting is the question of how to perform inference in the spatiotemporal model, since it contains loops. Accordingly, a specific message passing schedule has to be developed, and its convergence has to be proved. Further, tracking in the proposed model results in multi-target tracking, as each comprised variables may comprise several hypotheses for a specific object category. Maintaining, this multi-modality in the context of nonparametric inference is a critical future challenge.

8.2.2 Observable High-Level Nodes

This thesis is concerned with the task of road topology estimation based on low-level features, such as lane markings and road edges. However, all levels of the proposed CHM can be observable, in principal. For example, patches could be observed directly using object detections. Similarly, lanes could be observed directly

by vehicle tracks [Wei12, Siv13]. Towards this goal, additional observation potentials have to be introduced which model the dependencies of object detection and patches as well as of lanes and vehicle tracks. The key benefit of incorporating information on other vehicles is that it allows to detect activities in the scene [Gei14] including the detection of e.g., lane changes or of vehicles stopping at a stopline. Additionally, incorporating vehicle tracks would allow to detect lanes and roads even in scenarios, when roads are covered by snow or are completely unmarked.

However, vehicles driving on the road often occlude lane and road cues. Hence, the resulting graphical model includes self occluding components, and thus implicate a more complex independence structure than assumed in this thesis. A heuristic approach to address this issue is to guarantee the independence of lane and road cues, and vehicle detections using the dynamic detection range, which is introduced in Section 7.1. Another approach is to allow for loops in the graphical model [Sig08] or to explicitly model the presence of occlusion, as proposed in [Sig08, Sud04b].

8.2.3 Electronic Horizon and Localization

The vision-based lane and road perception approach presented in this thesis provides reliable lane and road perception results in the vehicles surrounding. However, the results are limited to the field of view of the used sensors. Further, the performance of vision-based lane and road perception approaches drastically decreases with increasing distance to the host vehicle. State-of-the-art navigation maps, on the other, hand may contain less detailed topological and semantic information. Their advantage, however, is that they comprise information on lanes and roads fare beyond the sensory field of view, and that they are not affected by clutter or occlusions.

The hierarchical framework proposed in this thesis provides several possibilities to incorporate navigation maps. First, if the navigation map contains detailed lane information, the proposed framework allows to fuse these information with the lane estimation results. Second, if the navigation map contains road-level information, they could be used as observations of roads. Subsequently, these coarse information could either be fused with the results obtained form the low-level cues or evaluated in a top down manner in order to detect inaccuracies in the map data (see Section 4.3.3). Moreover, navigation maps could be used to initiate hypothesis in a coarse to fine hierarchy [Spe13] to further bootstrap inference. More precisely, navigation maps typically contain information on road centerlines and intersections. These information could be used to initiate specific road or even intersection types, and thus restrict the hypothesis space which has to be processed during inference. For example, a navigation map may provide the information

that an intersection has only three roads. Even if the geometric information is expected to be inaccurate, it allows to restrict the number of intersection types (i.e., intersection with three roads), and thus to start inference with plausible parameters. Then, more accurate information on e.g., lanes, stopline positions and lane turn directions could be obtained from the sensory evidence, during inference. A principal challenge of the above approaches is that they require to know the exact position of the host vehicle on the navigation map. In fact, since the position on the map is estimated using GPS, position errors of approximately $10-15\,m$ have to be expected [Kap05, Töp10].

Therefore, an interesting extension to the proposed framework is to use it for the task of precise localization. The initial erroneous global location provided by a GPS receiver can be used to extract e.g., lane hypotheses from the navigation map, and to transform them into the vehicles coordinate system. In order to cope with the GPS-related uncertainties during this transformation, different hypotheses for each lane can be generated by sampling locations according to the GPS uncertainty. In the following, the importance weight of each of these hypothesis can be computed using the structure of the model (see Section 4.3.3). While many existing localization approaches rely on coarse topological scene knowledge [Hom12, Kon12, Wei11], using the rich amount of topological and semantic knowledge obtained by the proposed approach (e.g., road topology and stopline positions) is expected to lead to more accurate and robust results.

Bibliography

[Ada11] C. Adam, R. Schubert, N. Mattern and G. Wanielik: *Probabilistic road estimation and lane association using radar detections*. In *International Conference on Information Fusion (FUSION)*, pp. 1–8, IEEE, 2011.

[Alv08] J. M. Alvarez and A. Lopez: *Novel index for objective evaluation of road detection algorithms*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 815–820, IEEE, 2008.

[Alv12] J. M. Alvarez, T. Gevers, Y. LeCun and A. M. Lopez: *Road scene segmentation from a single image*. In *European Conference on Computer Vision (ECCV)*, pp. 376–389. Springer, 2012.

[Apo03] N. Apostoloff and A. Zelinsky: *Robust vision based lane tracking using multiple cues and particle filtering*. In *Intelligent Vehicles Symposium (IV)*, pp. 558–563, IEEE, 2003.

[Apo04] N. Apostoloff: *Vision based lane tracking using multiple cues and particle filtering*. PhD thesis, Australian National University, 2004.

[Aru02] M. S. Arulampalam, S. Maskell, N. Gordon and T. Clapp: „A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking“. *IEEE Transactions on Signal Processing* **50** (2), pp. 174–188, 2002.

[Avo02] D. Avots, E. Lim, R. Thibaux and S. Thrun: *A probabilistic technique for simultaneous localization and door state estimation with mobile robots in dynamic environments*. In *International Conference on Intelligent Robots and Systems*, vol. 1, pp. 521–526, IEEE, 2002.

[Ban05] A. Banerjee, I. S. Dhillon, J. Ghosh and S. Sra: *Clustering on the unit hypersphere using von Mises-Fisher distributions*. In *Journal of Machine Learning Research*, pp. 1345–1382, 2005.

[Bar06] D. Barber: „Expectation correction for smoothed inference in switching linear dynamical systems“. *The Journal of Machine Learning Research* **7**, pp. 2515–2540, 2006.

[Bar12a] D. Barber: *Bayesian reasoning and machine learning*. Cambridge University Press, 2012.

[Bar12b] A. Bartels, M.-M. Meinecke and S. Steinmeyer: *Lane Change Assistance*. In *Handbook of Intelligent Vehicles*, pp. 729–757. Springer, 2012.

[Beh96] R. Behringer: *Visuelle Erkennung und Interpretation des Fahrspurverlaufes durch Rechnersehen für ein autonomes Straßenfahrzeug*. PhD thesis, Universitüt der Bundeswehr München, 1996.

[Bil06] S. Bileschi: *StreetScenes: Towards scene understanding in still images*. Techn. Rep., DTIC Document, 2006.

[Bis06] C. Bishop et al.: *Pattern recognition and machine learning*, vol. 4. Springer New York, 2006.

[Bor09] A. Borkar, M. Hayes and M. T. Smith: *Robust lane detection and tracking with RANSAC and Kalman filter*. In *International Conference on Image Processing (ICIP)*, pp. 3261–3264, IEEE, 2009.

[Bro06] A. Broggi and S. Cattani: „An agent based evolutionary approach to path detection for off-road vehicle guidance“. *Pattern Recognition Letters* **27** (11), pp. 1164–1173, 2006.

[Bro13] A. Broggi, P. Grisleri and P. Zani: *Sensors Technologies for Intelligent Vehicles Perception Systems: A Comparsion between Vision and 3D-LIDAR*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 887–892, IEEE, 2013.

[BS04] Y. Bar-Shalom, X. R. Li and T. Kirubarajan: *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[Car06] F. Caron, E. Duflos, D. Pomorski and P. Vanheeghe: „GPS/IMU data fusion using multisensor Kalman filtering: introduction of contextual aspects“. *Information Fusion* **7** (2), pp. 221–230, 2006.

[Cha97] P. Charbonnier, F. Diebolt, Y. Guillard and F. Peyret: *Road markings recognition using image processing*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 912–917, IEEE, 1997.

[Cli90] P. Clifford: „Markov random fields in statistics“. *Disorder in physical systems* pp. 19–32, 1990.

[Dan09] R. Danescu and S. Nedevschi: „Probabilistic lane tracking in difficult road scenarios using stereovision“. *Intelligent Vehicles Symposium (IV)* **10** (2), pp. 272–282, 2009.

[Dan10] R. Danescu and S. Nedevschi: *Detection and classification of painted road objects for intersection assistance applications*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 433–438, IEEE, 2010.

[Dic87] E. D. Dickmanns and A. Zapp: *A curvature-based scheme for improving road vehicle guidance by computer vision*. In *Cambridge Symposium on Intelligent Robotics Systems*, pp. 161–168, International Society for Optics and Photonics, 1987.

[Dic88] E. D. Dickmanns and A. Zapp: *Autonomous High Speed Road Vehicle Guidance by Computer Vision*. In *10th International Federation of Automatic Control World Congress.*, vol. 1, 1988.

[Dic92] E. D. Dickmanns and B. D. Mysliwetz: „Recursive 3-D road and relative ego-state recognition“. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* **14** (2), pp. 199–213, 1992.

[Die05a] R. Diestel: *Graph Theory*. Springer, 2005.

[Die05b] K. Dietmayer, N. Kaempchen, K. Fuerstenberg, J. Kibbel, W. Justus and R. Schulz: *Roadway detection and lane detection using multilayer laserscanner*. In *Advanced Microsystems for Automotive Applications (AMAA)*, pp. 197–213. Springer, 2005.

[Dou01] A. Doucet, N. De Freitas, N. Gordon et al.: *Sequential Monte Carlo methods in practice*, vol. 1. Springer New York, 2001.

[Dua12] K. Duan, D. Batra and D. Crandall: *A Multi-layer Composite Model for Human Pose Estimation*. In *Proceedings of the British Machine Vision Conference*, pp. 116.1–116.11, BMVA Press, 2012.

[Duc10] C. Duchow: *Videobasierte Wahrnehmung markierter Kreuzungen mit lokalem Markierungstest und Bayes' scher Modellierung*, vol. 16. KIT Scientific Publishing, 2010.

[Eli06] G. Elidan: *Residual belief propagation: Informed scheduling for asynchronous message passing*. In *Twenty-second Conference on Uncertainty in Artificial Intelligence (UAI)*, 2006.

[Enz13] M. Enzweiler, P. Greiner, C. Knoeppel and U. Franke: *Towards multi-cue urban curb recognition*. In *Intelligent Vehicles Symposium (IV)*, pp. 902–907, IEEE, 2013.

[Ess09] A. Ess, T. Müller, H. Grabner and L. Van Gool: *Segmentation-based urban traffic scene understanding*. In *British Machine Vision Conference*, 2009.

[Fan05] C. Fanti, L. Zelnik-Manor and P. Perona: *Hybrid models for human motion recognition*. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 1166–1173, IEEE, 2005.

[Fel05] P. F. Felzenszwalb and D. P. Huttenlocher: „Pictorial structures for object recognition“. *International Journal of Computer Vision* **61** (1), pp. 55–79, 2005.

[Fis73] M. A. Fischler and R. A. Elschlager: „The representation and matching of pictorial structures“. *IEEE Transactions on Computers* **100** (1), pp. 67–92, 1973.

[Fis81] M. A. Fischler and R. C. Bolles: „Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography“. *Communications of the ACM* **24** (6), pp. 381–395, 1981.

[Fra07] U. Franke, H. Loose and C. Knoppel: *Lane recognition on country roads*. In *Intelligent Vehicles Symposium (IV)*, pp. 99–104, IEEE, 2007.

[Fre02] B. J. Frey: *Extending factor graphs so as to unify directed and undirected graphical models*. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pp. 257–264, Morgan Kaufmann Publishers Inc., 2002.

[Fri13] J. Fritsch, T. Kühnl and A. Geiger: *A New Performance Measure and Evaluation Benchmark for Road Detection Algorithms*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 428–432, IEEE, 2013.

[Gei11] A. Geiger, M. Lauer and R. Urtasun: *A generative model for 3d urban scene understanding from movable platforms*. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1945–1952, IEEE, 2011.

[Gei14] A. Geiger, M. Lauer, C. Wojek, C. Stiller and R. Urtasun: „3D Traffic Scene Understanding from Movable Platforms“. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 2014.

[Gib91] A. Gibbons: *Algorithmic Graph Theory*. Cambridge University Press, 1991.

[Gmb13] A. E. V. GmbH: *Automotive Data and Time-Triggered Framework (ADTF)*. `http://www.audi-electronics-venture.de/aev/brand/en/services/developmenttools/adtf.html`, September 2013.

[Gop12] R. Gopalan, T. Hong, M. Shneier and R. Chellappa: „A learning approach towards detection and tracking of lane markings“. *International Conference on Intelligent Transportation Systems (ITSC)* **13** (3), pp. 1088–1098, 2012.

[Gor93] N. J. Gordon, D. J. Salmond and A. F. Smith: *Novel approach to nonlinear/non-Gaussian Bayesian state estimation*. In *IEE Proceedings F (Radar and Signal Processing)*, vol. 140, pp. 107–113, IET, 1993.

[Gou05] C. Goutte and E. Gaussier: *A probabilistic interpretation of precision, recall and F-score, with implication for evaluation*. In *Advances in Information Retrieval*, pp. 345–359. Springer, 2005.

[Gum11] T. Gumpp, D. Nienhuser and J. Zollner: *Lane confidence fusion for visual occupancy estimation*. In *Intelligent Vehicles Symposium (IV)*, pp. 1043–1048, IEEE, 2011.

[Guo12a] C. Guo, S. Mita and D. McAllester: „Robust Road Detection and Tracking in Challenging Scenarios Based on Markov Random Fields With Unsupervised Learning“. *Transaction on Intelligent Transportation Systems* **13** (3), pp. 1338–1354, 2012.

[Guo12b] C. Guo, T. Yamabe and S. Mita: *Robust road boundary estimation for intelligent vehicles in challenging scenarios based on a semantic graph*. In *Intelligent Vehicles Symposium (IV)*, pp. 37–44, IEEE, 2012.

[Hal09] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten: „The WEKA data mining software: an update“. *ACM SIGKDD explorations newsletter* **11** (1), pp. 10–18, 2009.

[Her12] S. Herrmann and F. Schroven: *Situation analysis for driver assistance systems at urban intersections*. In *International Conference on Vehicular Electronics and Safety*, pp. 151–156, IEEE, 2012.

[Hil12] A. B. Hillel, R. Lerner, D. Levi and G. Raz: „Recent progress in road and lane detection: a survey“. *Machine Vision and Applications* pp. 1–19, 2012.

[Hom11] K. Homeier and L. Wolf: *RoadGraph: High level sensor data fusion between objects and street network*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1380–1385, IEEE, 2011.

[Hom12] F. Homm: *Fahrzeugeigenlokalisation im Kontext hochautomatisierter Fahrfunktionen*. PhD thesis, Technische Universität München, 2012.

[Hua09] A. S. Huang, D. Moore, M. Antone, E. Olson and S. Teller: „Finding multiple lanes in urban road networks with vision and lidar“. *Autonomous Robots* **26** (2-3), pp. 103–122, 2009.

[Hur13] J. Hur, S.-N. Kang and S. S.-W.: *Multi-Lane Detection in Urban Driving Environments Using Conditional Random Fields*. In *Intelligent Vehicles Symposium (IV)*, pp. 1297–1302, IEEE, 2013.

[Ihl03] A. Ihler, E. Sudderth, W. Freeman and A. Willsky: „Efficient multiscale sampling from products of Gaussian mixtures“. *Advances in Neural Information Processing Systems (NIPS)* **16**, pp. 1–8, 2003.

[Isa98a] M. Isard and A. Blake: „Condensation-Conditional density propagation for visual tracking“. *International Journal of Computer Vision* **29** (1), pp. 5–28, 1998.

[Isa98b] M. Isard and A. Blake: *A mixed-state condensation tracker with automatic model-switching*. In *International Conference on Computer Vision*, pp. 107–112, IEEE, 1998.

[Isa03] M. Isard: *PAMPAS: Real-valued graphical models for computer vision*. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. I–613, IEEE, 2003.

[Jor01] M. I. Jordan and T. J. Sejnowski: *Graphical models: Foundations of neural computation*. MIT Press, 2001.

[Jor04] M. I. Jordan: „Graphical models“. *Statistical Science (Special Issue on Bayesian Statistics)* pp. 140–155, 2004.

[Kab08] G. Kable: *VW's new self-parking system*. `http://www.autocar.co.uk/car-news/motoring/vws-new-self-parking-system`, April 2008.

[Kan95] K. Kanazawa, D. Koller and S. Russell: *Stochastic simulation algorithms for dynamic probabilistic networks*. In *Proceedings of the Eleventh conference on Uncertainty in Artificial Intelligence*, pp. 346–351, Morgan Kaufmann Publishers Inc., 1995.

[Kap05] E. D. Kaplan and C. J. Hegarty: *Understanding GPS: principles and applications*. Artech house, 2005.

[KAW13] K. KAWASE: „Concise Derivation of Extensive Coordinate Conversion Formulae in the Gauss-Krüger Projection". *Bulletin of the Geospatial Information Authority of Japan* **60**, 2013.

[Kim08] Z. Kim: „Robust lane detection and tracking in challenging scenarios". *Transaction on Intelligent Transportation Systems* **9** (1), pp. 16–26, 2008.

[Klu95a] K. Kluge and S. Lakshmanan: *A deformable-template approach to lane detection*. In *Intelligent Vehicles Symposium (IV)*, pp. 54–59, IEEE, 1995.

[Klu95b] K. Kluge and C. Thorpe: „The YARF system for vision-based road following". *Mathematical and Computer Modelling* **22** (4), pp. 213–233, 1995.

[Kna10a] J. Knaup, S. Herrmann, M.-M. Meinecke and M. A. Obojski: *Progress of Intersection Safety System Development: Volkswagen Within the INTERSAFE-2 Project*. In *Advanced Microsystems for Automotive Applications 2010*, pp. 241–251. Springer, 2010.

[Kna10b] J. Knaup and K. Homeier: *RoadGraph-Graph based environmental modelling and function independent situation analysis for driver assistance systems*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 428–432, IEEE, 2010.

[Kol99] D. Koller, U. Lerner and D. Angelov: *A general algorithm for approximate inference and its application to hybrid Bayes nets*. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pp. 324–333, Morgan Kaufmann Publishers Inc., 1999.

[Kol09] D. Koller and N. Friedman: *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[Kon10] M. Konrad, M. Szczot and K. Dietmayer: *Road course estimation in occupancy grids*. In *Intelligent Vehicles Symposium (IV)*, pp. 412–417, IEEE, 2010.

[Kon11] M. Konrad, M. Szczot, F. Schule and K. Dietmayer: *Generic grid mapping for road course estimation*. In *Intelligent Vehicles Symposium (IV)*, pp. 851–856, IEEE, 2011.

[Kon12] M. Konrad, D. Nuss and K. Dietmayer: *Localization in digital maps for road course estimation using grid maps*. In *Intelligent Vehicles Symposium (IV)*, pp. 87–92, IEEE, 2012.

[Ksc01] F. R. Kschischang, B. J. Frey and H.-A. Loeliger: „Factor graphs and the sum-product algorithm“. *Transactions on Information Theory* **47** (2), pp. 498–519, 2001.

[Kue10] D. Kuettel, M. Breitenstein, L. Van Gool and V. Ferrari: *What's going on? discovering spatio-temporal dependencies in dynamic scenes*. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1951–1958, IEEE, 2010.

[Kue11] T. Kuehnl, F. Kummert and J. Fritsch: *Monocular road segmentation using slow feature analysis*. In *Intelligent Vehicles Symposium (IV)*, pp. 800–806, IEEE, 2011.

[Kue12] T. Kuehnl, F. Kummert and J. Fritsch: *Spatial ray features for real-time ego-lane extraction*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 288–293, IEEE, 2012.

[Kuh11] T. Kuhnl, F. Kummert and J. Fritsch: *Monocular road segmentation using slow feature analysis*. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pp. 800–806, 2011.

[Lin11] A. Linarth and E. Angelopoulou: *On feature templates for Particle Filter based lane detection*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1721–1726, IEEE, 2011.

[Loo10] H. Loose and U. Franke: *B-spline-based road model for 3d lane recognition*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 91–98, IEEE, 2010.

[Mac98] D. J. MacKay: *Introduction to monte carlo methods*. In *Learning in graphical models*, pp. 175–204. Springer, 1998.

[Mal91] H. A. Mallot, H. H. Bülthoff, J. Little and S. Bohrer: „Inverse perspective mapping simplifies optical flow computation and obstacle detection“. *Biological Cybernetics* **64** (3), pp. 177–185, 1991.

[Man13] M. Manz: *Modellbasierte visuelle Wahrnehmung zur autonomen Fahrzeugführung*. PhD thesis, Universität der Bundeswehr München, 2013.

[Mar01] G. Marsden, M. McDonald and M. Brackstone: „Towards an understanding of adaptive cruise control“. *Transportation Research Part C: Emerging Technologies* **9** (1), pp. 33–51, 2001.

[McC06] J. C. McCall and M. M. Trivedi: „Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation“. *Intelligent Vehicles Symposium (IV)* **7** (1), pp. 20–37, 2006.

[Met49] N. Metropolis and S. Ulam: „The monte carlo method“. *Journal of the American statistical association* **44** (247), pp. 335–341, 1949.

[Moh01] A. Mohan, C. Papageorgiou and T. Poggio: „Example-based object detection in images by components“. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* **23** (4), pp. 349–361, 2001.

[Mur03] K. Murphy, A. Torralba and W. Freeman: „Using the forest to see the trees: a graphical model relating features, objects and scenes“. *Advances in Neural Information Processing Systems (NIPS)* **16**, 2003.

[Mur06] K. P. Murphy: *Machine Learning A Probabilistic Approach*. MIT Press, 2006.

[Mus01] C. Musso, N. Oudjane and F. Le Gland: *Improving regularised particle filters*. In *Sequential Monte Carlo methods in practice*, pp. 247–271. Springer, 2001.

[Ned09] S. Nedevschi, M. Tiberiu, R. Danescu, F. Oniga and S. Bota: *Onboard stereo sensor for intersection driving assistance architecture and specification*. In *5th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pp. 409–416, IEEE, 2009.

[Oni11] F. Oniga and S. Nedevschi: *Curb detection for driving assistance systems: A cubic spline-based approach*. In *Intelligent Vehicles Symposium (IV)*, pp. 945–950, IEEE, 2011.

[Par62] E. Parzen: „On estimation of a probability density function and mode“. *The annals of mathematical statistics* **33** (3), pp. 1065–1076, 1962.

[Pea88] J. Pearl: *Probabilistic Reasoning in Intelligent Systems Networks of Plausible Inference*. Morgan Kaufmann Pub, 1988.

[Pom95] D. Pomerleau: *RALPH: Rapidly adapting lateral position handler*. In *Intelligent Vehicles Symposium (IV)*, pp. 506–511, IEEE, 1995.

[Rab89] L. R. Rabiner: „A tutorial on hidden Markov models and selected applications in speech recognition“. *Proceedings of the IEEE* **77** (2), pp. 257–286, 1989.

[ras95] *Richtlinien für die Anlage von Straßen - Linienführung*. Techn. Rep., Forschungsgesellschaft für Straßen- und Verkehrswesen, 1995.

[Ras05] C. Rasmussen and T. Korah: *On-vehicle and aerial texture analysis for vision-based desert road following*. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 66–66, IEEE, 2005.

[Rau12] S. Rauch, T. Schaller, A. Savkin and P. Hecker: *Hochgenaue Fahrzeugeigenlokalisierung und kollektives Erlernen hochgenauer digitaler Karten*. In *Automatisierungs-, Assistenzsysteme und eingebettete Systeme für Transportmittel*, 2012.

[Rob04] C. P. Robert and G. Casella: *Monte Carlo Statistical Methods*, vol. 319. Springer-Verlag, 2004.

[Rus10] S. J. Russell, P. Norvig, E. Davis, S. J. Russell and S. J. Russell: *Artificial intelligence: a modern approach*, vol. 2. Prentice hall Englewood Cliffs, 2010.

[Sch03] D. A. Schwartz: *Clothoid road geometry unsuitable for sensor fusion clothoid parameter sloshing*. In *Intelligent Vehicles Symposium (IV)*, pp. 484–488, IEEE, 2003.

[Sei13] A. Seibert, H. Haehnel, A. Tewes and R. Rojas: *Towards multi-cue urban curb recognition*. In *Intelligent Vehicles Symposium (IV)*, pp. 902–907, IEEE, 2013.

[Ser08] M. Serfling, R. Schweiger and W. Ritter: *Road course estimation in a night vision application using a digital map, a camera sensor and a prototypical imaging radar system*. In *Intelligent Vehicles Symposium (IV)*, pp. 810–815, IEEE, 2008.

[Shi12] P. Y. Shinzato, V. Grassi, F. S. Osório and D. F. Wolf: *Fast visual road recognition and horizon detection using multiple artificial neural networks*. In *Intelligent Vehicles Symposium (IV)*, pp. 1090–1095, IEEE, 2012.

[Sie11] J. Siegemund, U. Franke and W. Forstner: *A temporal filter approach for detection and reconstruction of curbs and road surfaces based on conditional random fields*. In *Intelligent Vehicles Symposium (IV)*, pp. 637–642, IEEE, 2011.

[Sig04] L. Sigal, S. Bhatia, S. Roth, M. Black and M. Isard: *Tracking loose-limbed people*. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. I–421 – I–428 Vol.1, IEEE, june-2 july 2004.

[Sig08] L. Sigal: *Continuous-state Graphical Models for Object Localization, Pose Estimation and Tracking*. PhD thesis, Brown University, 2008.

[Sil86] B. W. Silverman: *Density estimation for statistics and data analysis*, vol. 26. Chapman & Hall, 1986.

[Siv13] S. Sivaraman and M. M. Trivedi: „Integrated Lane and Vehicle Detection, Localization, and Tracking: A Synergistic Approach“. *Transaction on Intelligent Transportation Systems* **14** (2), 2013.

[Sok06] M. Sokolova, N. Japkowicz and S. Szpakowicz: *Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation*. In *Advances in Artificial Intelligence*, pp. 1015–1021. Springer, 2006.

[Spe11] J. Spehr, D. Rosebrock, D. Mossau, R. Auer, S. Brosig and F. Wahl: *Hierarchical scene understanding for intelligent vehicles*. In *Intelligent Vehicles Symposium (IV)*, pp. 1142–1147, IEEE, 2011.

[Spe13] J. Spehr: *On Hierarchical Models for Visual Recognition and Learning of Objects, Scenes, and Activities*. PhD thesis, Technical University Braunschweig, 2013.

[Ste03] T. A. Stephenson: *Conditional Gaussian Mixtures*. Techn. Rep., Technical Report, IDIAP-RR 03-11, 2003.

[Stu09] P. Sturgess, K. Alahari, L. Ladicky and P. Torr: „Combining appearance and structure from motion features for road scene understanding“2009.

[Sud03] E. B. Sudderth, A. T. Ihler, W. T. Freeman and A. S. Willsky: *Nonparametric Belief Propagation*. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 605–612, 2003.

[Sud04a] E. Sudderth, M. Mandel, W. Freeman and A. Willsky: „Distributed occlusion reasoning for tracking with nonparametric belief propagation“. *Advances in Neural Information Processing Systems (NIPS)* **17**, pp. 1369–1376, 2004.

[Sud04b] E. Sudderth, M. Mandel, W. Freeman and A. Willsky: „Distributed occlusion reasoning for tracking with nonparametric belief propagation“. *Advances in Neural Information Processing Systems (NIPS)* **17**, pp. 1369–1376, 2004.

[Sud06] E. B. Sudderth: *Graphical Models for Visual Object Recognition and Tracking*. PhD thesis, Massachusetts Institute of Technology, 2006.

[Thr05] S. Thrun, W. Burgard, D. Fox et al.: *Probabilistic robotics*, vol. 1. MIT Press, 2005.

[Töp10] D. Töpfer: *Konzeption und Implementierung einer Stopp- und Ampelassistenz*. Master's thesis, Technische Universität Braunschweig, 2010.

[Töp13] D. Töpfer, J. Spehr, J. Effertz and C. Stiller: *Efficient Scene Understanding for Intelligent Vehicles Using a Part-Based Road Representation*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 65–70, IEEE, 2013.

[Tor10] A. Torralba, K. P. Murphy and W. Freeman: „Using the forest to see the trees: exploiting context for visual object detection and localization“. *Communications of the ACM* **53** (3), pp. 107–114, 2010.

[Urm08] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer et al.: „Autonomous driving in urban environments: Boss and the urban challenge“. *Journal of Field Robotics* **25** (8), pp. 425–466, 2008.

[Vei08] T. Veit, J. Tarel, P. Nicolle and P. Charbonnier: *Evaluation of road marking feature extraction*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 174–181, IEEE, 2008.

[Ver4a] V. Verma, G. Gordon, R. Simmons and S. Thrun: „Real-time fault diagnosis [robot fault diagnosis]“. *Robotics & Automation Magazine* **11** (2), pp. 56–66, verma2004a.

[Ver04] V. Verma: *Tractable particle filters for robot fault diagnosis*. PhD thesis, Stanford University, 2004.

[Wai08] M. J. Wainwright and M. I. Jordan: „Graphical models, exponential families, and variational inference“. *Foundations and Trends® in Machine Learning* **1** (1-2), pp. 1–305, 2008.

[Wan04] Y. Wang, E. K. Teoh and D. Shen: „Lane detection and tracking using B-Snake“. *Image and Vision Computing* **22** (4), pp. 269–280, 2004.

[Wan09] X. Wang, X. Ma and W. Grimson: „Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models“. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* **31** (3), pp. 539–555, 2009.

[Wan13] B. Wang and V. Fremont: *Fast road detection from color images*. In *Intelligent Vehicles Symposium (IV)*, pp. 1209–1214, IEEE, 2013.

[Wei01] Y. Weiss and W. Freeman: „Correctness of belief propagation in gaussian graphical models of arbitrary topology“. *Neural Computation* **13** (10), pp. 2173–2200, 2001.

[Wei11] T.-T. Weiss: *Hochgenaue Positionierung und Kartographie mit Laserscannern für Fahrerassistenzsysteme*. PhD thesis, Universität Ulm, 2011.

[Wei12] T. Weiherer, E. Bouzouraa and U. Hofmann: *A generic map based environment representation for driver assistance systems applied to detect convoy tracks*. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 691–696, IEEE, 2012.

[Win05] M. G. Wing, A. Eklund and L. D. Kellogg: „Consumer-grade global positioning system (GPS) accuracy and reliability“. *Journal of Forestry* **103** (4), pp. 169–173, 2005.

[Woj08] C. Wojek and B. Schiele: „A dynamic conditional random field model for joint labeling of object and scene classes“. *European Conference on Computer Vision (ECCV)* pp. 733–747, 2008.

[Woj10] C. Wojek, S. Roth, K. Schindler and B. Schiele: „Monocular 3d scene modeling and inference: Understanding multi-object traffic scenes“. *European Conference on Computer Vision (ECCV)* pp. 467–481, 2010.

[Wu11] Q. Wu, W. Zhang and B. V. Kumar: *Example-based clear path detection assisted by vanishing point estimation*. In *International Conference on Robotics and Automation (ICRA)*, pp. 1615–1620, IEEE, 2011.

[Xie03] B. Xie, D. Comaniciu, V. Ramesh, M. Simon and T. Boult: *Component fusion for face detection in the presence of heteroscedastic noise*. In *Annual Conference of the German Society for Pattern Recognition (DAGM)*, pp. 434–441. Springer, 2003.

[Yan11] Y. Yang and D. Ramanan: *Articulated pose estimation with flexible mixtures-of-parts*. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1385–1392, IEEE, 2011.

[Yed00] J. Yedidia: „An idiosyncratic journey beyond mean field theory“. *Advanced mean field methods: Theory and practice* pp. 21–36, 2000.

[Yed01] J. S. Yedidia, W. T. Freeman, Y. Weiss et al.: „Generalized belief propagation“. *Advances in Neural Information Processing Systems (NIPS)* pp. 689–695, 2001.

[Yed02] J. S. Yedidia, W. T. Freeman and Y. Weiss: „Understanding belief propagation and its generalizations“. *Exploring Artificial Intelligence in the New Millennium* **8**, pp. 236–239, 2002.

[Zha12] K. Zhao, M. Meuter, C. Nunn, D. Muller, S. Muller-Schneiders and J. Pauli: *A novel multi-lane detection and tracking system*. In *Intelligent Vehicles Symposium (IV)*, pp. 1084–1089, IEEE, 2012.

[Zhu10] L. L. Zhu, Y. Chen, A. Torralba, W. Freeman and A. Yuille: „Part and appearance sharing: Recursive compositional models for multi-view multi-object detection“. *Conference on Computer Vision and Pattern Recognition (CVPR)* pp. 1919–1926, 2010.

[Zhu12] X. Zhu and D. Ramanan: *Face detection, pose estimation, and landmark localization in the wild*. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2879–2886, IEEE, 2012.