Stixmentation - From Stixels to Objects

Dissertation
zur Erlangung des Doktorgrades
der Naturwissenschaften

vorgelegt beim Fachbereich Informatik und Mathematik
der Johann Wolfgang Goethe-Universität
in Frankfurt am Main

von
Friedrich Philipp Joachim Erich Erbs
aus Mannheim

Frankfurt 2016

vom Fachbereich Informatik und Mathematik der
Johann Wolfgang Goethe-Universität als Dissertation angenommen.

Dekan: Prof. Dr. Uwe Brinkschulte

Gutachter: Prof. Dr. Rudolf Mester und Prof. Dr. Jochen Triesch

Datum der Disputation: 2. November 2016

Abstract

Already today modern driver assistance systems contribute more and more to make individual mobility in road traffic safer and more comfortable. For this purpose, modern vehicles are equipped with a multitude of sensors and actuators which perceive, interpret and react to the environment of the vehicle. In order to reach the next set of goals along this path, for example to be able to assist the driver in increasingly complex situations or to reach a higher degree of autonomy of driver assistance systems, a detailed understanding of the vehicle environment and especially of other moving traffic participants is necessary.

It is known that motion information plays a key role for human object recognition [Spelke, 1990]. However, full 3D motion information is mostly not taken into account for Stereo Vision-based object segmentation in literature. In this thesis, novel approaches for motion-based object segmentation of stereo image sequences are proposed from which a generic environmental model is derived that contributes to a more precise analysis and understanding of the respective traffic scene. The aim of the environmental model is to yield a minimal scene description in terms of a few moving objects and stationary background such as houses, crash barriers or parking vehicles. A minimal scene description aggregates as much information as possible and it is characterized by its stability, precision and efficiency.

Instead of dense stereo and optical flow information, the proposed object segmentation builds on the so-called Stixel World, an efficient superpixel-like representation of space-time stereo data. As it turns out this step substantially increases stability of the segmentation and it reduces the computational time by several orders of magnitude, thus enabling real-time automotive use in the first place. Besides the efficient, real-time capable optimization, the object segmentation has to be able to cope with significant noise which is due to the measurement principle of the used stereo camera system. For that reason, in order to obtain an optimal solution under the given extreme conditions, the segmentation task is formulated as a Bayesian optimization problem which allows to incorporate regularizing prior knowledge and redundancies into the object segmentation. Object segmentation as it is discussed here means unsupervised segmentation since typically the number of objects in the scene and their individual object parameters are not known in advance. This information has to be estimated from the input data as well.

For inference, two approaches with their individual pros and cons are proposed, evaluated and compared. The first approach is based on dynamic programming. The key advantage of this approach is the possibility to take into account non-local priors such as shape or object size information which is impossible or which is prohibitively expen-

sive with more local, conventional graph optimization approaches such as graphcut or belief propagation.
In the first instance, the Dynamic Programming approach is limited to one-dimensional data structures, in this case to the first Stixel row. A possible extension to capture multiple Stixel rows is discussed at the end of this thesis.
Further novel contributions include a special outlier concept to handle gross stereo errors associated with so-called stereo tear-off edges. Additionally, object-object interactions are taken into account by explicitly modeling object occlusions. These extensions prove to be dramatic improvements in practice.
This first approach is compared with a second approach that is based on an alternating optimization of the Stixel segmentation and of the relevant object parameters in an expectation maximization (EM) sense. The labeling step is performed by means of the $\alpha-$expansion graphcut algorithm, the parameter estimation step is done via one-dimensional sampling and multidimensional gradient descent. By using the Stixel World and due to an efficient implementation, one step of the optimization only takes about one millisecond on a standard single CPU core. To the knowledge of the author, at the time of development there was no faster global optimization in a demonstrator car.

For both approaches, various testing scenarios have been carefully selected and allow to examine the proposed methods thoroughly under different real-world conditions with limited groundtruth at hand. As an additional innovative application, the first approach was successfully implemented in a demonstrator car that drove the so-called Bertha Benz Memorial Route from Mannheim to Pforzheim autonomously in real traffic.
At the end of this thesis, the limits of the proposed systems are discussed and a prospect on possible future work is given.

Deutsche Zusammenfassung

Einführung

Bereits heute tragen moderne Fahrerassistenzsysteme immer mehr dazu bei, die individuelle Mobilität im Straßenverkehr sicherer und komfortabler zu gestalten. Zu diesem Zweck werden in modernen Fahrzeugen eine Vielzahl von Sensoren und Aktuatoren verbaut, welche das Fahrzeugumfeld wahrnehmen, interpretieren und darauf reagieren. Um die nächsten Ziele auf diesem Weg erreichen zu können, zum Beispiel den Fahrer in immer schwierigeren Situationen unterstützen oder einen höheren Autonomiegrad der Fahrerassistenzsysteme erreichen zu können, wird ein detailliertes Verständnis des Fahrzeugumfelds und insbesondere auch anderer bewegter Verkehrsteilnehmer benötigt.

Formulierung des Segmentierungsproblems

Es ist bekannt, dass Bewegungsinformation eine Schlüsselrolle bei der Objekterkennung des Menschen spielt [Spelke, 1990]. Dennoch wird in der entsprechenden Literatur die volle 3D Bewegungsinformation für die Objektsegmentierung auf Basis von Stereobildverarbeitung meist nicht herangezogen. In der vorliegenden Arbeit werden neue Ansätze zur bewegungsbasierten Objektsegmentierung anhand von Stereo-Bildfolgen vorgestellt, womit ein generisches Fahrzeugumfeldmodell abgeleitet wird, welches zu einer genaueren Analyse und zum Verständnis der jeweiligen Verkehrsszene beiträgt. Ziel des Umfeldmodelles ist es, eine minimale Szenenbeschreibung in Form einiger weniger bewegter Objekte und des stationären Hintergrundes wie Häuser, Leitplanken oder parkender Autos, zu liefern. Eine solche minimale Szenenbeschreibung fasst so viel Information wie möglich zusammen und zeichnet sich durch ihre Stabilität, Genauigkeit und Effizienz aus. Obwohl das Umfeldmodell in der vorliegenden Arbeit auf den Daten eines Stereokamerasystems basiert, können die vorgeschlagen generischen Algorithmen prinzipiell auch für andere Sensor-Daten wie Radar, Lidar oder Ultraschall verwendet werden.

Die Stixel-Welt

Anstelle von dichter Stereo- und Bewegungsinformation über den optischen Fluss wird in der vorgestellten Objektsegmentierung auf der sogenannten Stixel-Welt aufgebaut, einer effizienten, superpixelartigen Beschreibung räumlich-zeitlicher Stereodaten. Das künstliche Wort Stixel leitet sich von den englischen Begriffen „stick “und „pixel “, also etwa stabförmiger Pixel, ab. Die Stixel-Welt wurde ursprünglich von Badino [Badino et al., 2009] vorgeschlagen und dann von Pfeiffer [Pfeiffer and Franke, 2011] wesentlich erweitert. Die stabförmigen Superpixel unterteilen eine Stereo-Disparitätskarte in

befahrbaren Freiraum und aufrechte Hindernisse. Für die Hindernisse wird eine Höhe sowie eine mittlere Tiefe mitgeschätzt. Verfolgt man die Position der Stixel über die Zeit, gelangt man zur sogenannten dynamischen Stixel-Welt [Pfeiffer and Franke, 2010]. Diese dynamischen Stixel liefern zusätzlich einen Schätzwert für die Bewegung und Geschwindigkeit der Hindernisse.
Die Stixel-Welt hat eine feste Breite im Bild von typischerweise 5-10 Pixeln und koppelt benachbarte Stixel nicht. Insofern liefert sie eine Übersegmentierung des Bildes. Nachfolgende Algorithmen, die auf den Daten der dynamischen Stixel-Welt aufbauen, wie eine Bremsfunktion oder eine Trajektorienplanung im Kontext des hochautonomen Fahrens, bevorzugen eine robustere, minimale Szenenbeschreibung aus einzelnen bewegten Objekten und einer geometrischen Beschreibung der stationären Infrastruktur. Die in dieser Arbeit untersuchte Objektsegmentierung liefert eine Interpretation der Verkehrsszene auf Basis der dynamischen Stixel-Welt und füllt damit die Lücke zwischen den reinen verrauschten Sensor-Messdaten (Stixel-Welt) auf der einen Seite und möglichen, nachgelagerten Algorithmen auf der anderen Seite.
Wie sich herausstellt, steigert der Schritt weg von dichter Stereo- und optischer Flussinformation hin zur dynamischen Stixelwelt die Stabilität der Segmentierung ungemein und reduziert die Rechenzeit um mehrere Größenordnungen, wodurch der Echtzeiteinsatz im Automobil überhaupt erst möglich wird. Der Übergang zur Stixel-Welt für die Objektsegmentierung stellt einen ersten wesentlichen Beitrag dieser Arbeit dar.
Neben einer effizienten und echtzeitfähigen Optimierung muss die Objektsegmentierung mit einem erheblichen Rauschpegel zurecht kommen, bedingt durch das Messprinzip des verwendeten Stereokamerasystems. Um unter den gegebenen, extremen Randbedingungen eine optimale Lösung zu erhalten, wird die Segmentierung als Bayes'sches Optimierungsproblem formuliert, welches es erlaubt, regularisierendes Vorwissen in die Objektsegmentierung miteinfließen zu lassen. Objektsegmentierung wie sie hier diskutiert wird meint stets unüberwachte Segmentierung, da die Zahl der Objekte in der Szene und ihre Eigenschaften typischerweise im Voraus nicht bekannt sind. Diese Informationen müssen ebenfalls aus den Eingangsdaten geschlossen werden.

Lösungsansatz über dynamische Programmierung

Die Segmentierung als Bayes'sches Optimierungsproblem lässt sich als Energieminimierungsproblem formulieren, welches einmal die Stixel verschiedenen realen Objekten beziehungsweise Objektklassen zuweist und gleichzeitig über deren Anzahl und Eigenschaften inferiert. Die vollständig in dieser Arbeit abgeleitete, generative Energiefunktion basiert im Wesentlichen auf gelernten Verteilungsdichtefunktionen der Messdaten und der Regularisierungsparameter. Für die Inferenz werden zwei Ansätze mit ihren jeweiligen Vor- und Nachteilen vorgeschlagen, evaluiert und miteinander verglichen. Der erste Ansatz basiert auf dynamischer Programmierung. Der entscheidende Vorteil dieses Ansatzes ist die Möglichkeit, nicht-lokales Vorwissen wie die Form von Objekten oder ihre Größe mit in die Segmentierung hineinzunehmen, was mit bekannten, eher lokalen Graphenoptimierungsalgorithmen wie Graphcut oder Belief Propagation nicht möglich oder sehr aufwändig wäre. Diese globalere Optimierung verbessert die Objekterkennung deutlich und stellt einen weiteren wichtigen Beitrag dieser Arbeit dar.

Der Ansatz der dynamischen Programmierung ist zunächst auf eindimensionale Datenstrukturen, hier die erste Stixel-Reihe beschränkt. In den meisten Fällen stellt dies für nachgelagerte Funktionen keine große Einschränkung dar. In der Regel beschreiben die zweite oder dritte Stixel-Reihe stationären Hintergrund wie Häuser, der für die Regelung des eigenen Fahrzeugs meist nicht unmittelbar relevant ist. Einschränkungen ergeben sich allenfalls in Verdeckungsszenarien, für Schlechtwetter-Szenarien mit vielen Phantom-Stixeln oder im Sinne eines noch holistischeren Bildverstehens. Dazu wird am Ende der Arbeit eine mögliche Erweiterung vorgestellt, um mehrere Stixel-Reihen erfassen zu können. Im Grunde genommen ist dieser Ansatz eine natürliche Erweiterung der dynamischen Programmierung auf Baumstrukturen. Um die Inferenz effizient zu gestalten, wird diese auf sogenannte „Spinnen-Strukturen“ beschränkt, das sind Unterbäume mit nur höchstens einem Verzweigungsknoten. Diese Struktur schließt die erste Stixel-Reihe auf natürliche Weise mit ein. Die vorliegende Arbeit diskutiert diesen Ansatz ausführlich und zeigt erste Ergebnisse.

Notwendige Erweiterungen

Die probabilistische Modellierung des Segmentierungsproblems und die effiziente Inferenz sind wesentliche Bestandteile der vorgestellten Algorithmen. In realen, anspruchsvollen Verkehrsszenarien treten jedoch weitere Schwierigkeiten auf, welche eine Erweiterung des Ansatzes notwendig machen.
Ein wichtiger Beitrag der Arbeit ist ein besonderes Ausreißerkonzept, welches erlaubt, mit groben Stereorekonstruktionsfehlern, sogenannten Stereoabrisskanten, umzugehen. Stereoabrisskanten entstehen typischerweise an Objektgrenzen durch ein Verschmieren der Tiefeninformation bei zu starker Glättung von global optimierenden Stereoalgorithmen wie SGM [Hirschmuller, 2005]. Die daraus resultierenden Ausreißer-Stixel erhöhen die Objektdimensionen signifikant auf unplausible Werte und können, wenn sie sich im eigenen Fahrkanal befinden, zu Notbremsungen führen. Da es das Ziel ist, zu wirklich jedem Stixel eine Aussage treffen zu können, wird eine Klassifikation dieser Ausreißer-Stixel notwendig.
Wie oben beschrieben ermöglicht es der Ansatz der dynamischen Programmierung, Objektwissen wie deren Größe in die Optimierung miteinfließen zu lassen. Obwohl sich ein genereller Objektdimensions-Prior als enorme Verbesserung erweist, treten insbesondere für Verdeckungsszenarien Abweichungen von dessen Modellannahmen auf. Um solche Objekt-Objekt Interaktionen richtig behandeln zu können, wird es nötig, Objektverdeckungen explizit zu modellieren. Die globale Optimierung versucht, über das Vorhandensein von Verdeckungs-Konfigurationen mitzuinferieren, um so zu realisitischeren Szenenbeschreibungen zu gelangen. Auch diese Erweiterung erweist sich in der Praxis als enorme Verbesserung.

Lösungsansatz über EM-Graphcut

Der Ansatz basierend auf dynamischer Programmierung wird mit einem weiteren Ansatz verglichen, der auf einer alternierenden Optimierung der Stixel-Segmentierung und der relevanten Objektparameter beruht im Sinne eines Erwartungswert-Maximierungs-

Algorithmus (EM). Der Klassenzuweisungsschritt wird mit Hilfe des $\alpha-$Expansion Graphcut Algorithmus durchgeführt, hierfür werden die Zahl und Parameter der einzelnen Objektklassen als bekannt angenommen. Die Schätzung der Objektparameter erfolgt über eindimensionale Abtastung und mehrdimensionalen Gradientenabstieg. Die Zahl der bewegten Objekte in der Verkehrsszene ergibt sich aus der Optimierung und einer statistischen Modellauswahl.
Neben dieser Lösung, die vollständig auf Kameradaten beruht, kann die Schätzung der Objektparameter auch über einen anderen Sensor erfolgen. Alternativ wird daher die Initialisierung der Objektparameter über Radar-Daten diskutiert.
Durch Verwendung der Stixel-Welt und einer effizienten Implementierung benötigt ein einzelner Schritt der Optimierung lediglich etwa 1 ms auf einem einzelnen, standardmäßigen CPU-Kern. Nach Wissen des Autors gab es zum Zeitpunkt der Entwicklung keine schnellere Graphcut-Optimierung in einem Versuchsträger.

Experimentelle Ergebnisse

Für beide Ansätze wurden Testszenarien entwickelt, welche es erlauben, die vorgestellten Algorithmen gründlich in verschiedenen realen Verkehrsszenen zu testen, selbst ohne riesige Mengen an Referenzdaten.
Zunächst werden die Segmentierungsergebnisse beider Ansätze gegen eine große Zahl von mehreren tausend manuell annotierten Bildern als Refernzergebnisse verglichen. Es wird gezeigt, dass der Ansatz über dynamische Programmierung insbesondere bei der Erkennung bewegter Objekte für große Entfernungen, wo die Eingangsdaten schwach sind und starke Regularisierung der Ergebnisse notwendig ist, klare Vorteile bietet. Wie erwartet liefert der Ansatz über dynamische Programmierung auch in Verdeckungsszenarien und für unsichere Stixel am Rand von Objekten bessere Ergebnisse. Trotz dieser deutlich höheren Erkennungsleistung zeigt der Ansatz über dynamische Programmierung lediglich eine leicht erhöhte Falschalarmrate, so dass man insgesamt von einem deutlichen Fortschritt sprechen kann.
Der Ansatz der dynamischen Programmierung wird anschließend weiter untersucht. Zunächst wurde ein Objektverfolgungs-Testszenario entwickelt, in der die Objektsegmentierung die Geschwindigkeit eines vorausfahrenden Fahrzeugs schätzt, welches seine eigene Geschwindigkeit - über die fahrzeuginterne Inertialsensorik gemessen - aufzeichnet. Das vorausfahrende Fahrzeug fährt hierbei eine Serpentinenstrecke und moduliert seine Eigengeschwindigkeit stark durch starkes Beschleunigen und Abbremsen. Diese Inertialsensorikdaten werden als annähernd korrekte Referenzdaten angesehen. Es wird anschließend die Geschwindigkeitsschätzung der Objektsegmentierung verglichen mit diesen Referenzdaten.
Es zeigt sich in diesem Testszenario, dass die vorgeschlagene Objektsegmentierung das vorausfahrende Fahrzeug zum einen problemlos ohne Abbrüche verfolgen kann, ohne falsche Phantom-Objekte aufzusetzten und weiter zu verfolgen. Zum anderen ist trotz hoher relativer Objektdynamiken die geschätzte Geschwindigkeit des Objekts sehr nah an der tatsächlichen Objektgeschwindigkeit, im Mittel berträgt die Abweichung weniger als 1 m/s. Zusammen mit der oben beschriebenen Stabilität der Segmentierung stellt dies eine wesentliche Anforderung für das hochautonome Fahren dar.

Um den tatsächlichen Wert der Objektsegmentierung noch klarer herauszuarbeiten, wurde in einem dritten Testszenario mit besonderem Hinblick auf das hochautonome Fahren die Anzahl der möglichen Notbremsungen aufgrund sehr kleiner Kollisionszeiten (TTC) aufgrund von falsch geschätzter Objektkinematik evaluiert. Sehr kleine Kollisionszeiten (TTC $<$ 1s), die in den eingefahrenen Szenarien nicht vorkamen, weisen deutlich auf Fehler in der Geschwindigkeitsschätzung hin. Hierzu wurden die Ergebnisse der Objektsegmentierung mit denen der ursprünglichen dynamischen Stixel-Welt bezüglich vermeintlich sehr niedriger Kollisionszeiten verglichen. Für die Analyse wurde der tatsächlich gefahrene Fahrkorridor des eigenen Fahrzeugs aus den aufgezeichneten Inertialsensorikdaten rekonstruiert. Dies stellt sozusagen den Planungskorridor des eigenen Fahrzeugs dar. Dieser Pfad wird geschnitten mit der prädizierten Position der Stixel, entsprechend ihrer geschätzten Geschwindigkeit. Hierbei kann es zu vermeintlichen Kollisionen kommen. Zum Vergleich der beiden Algorithmen ist zu beachten, dass die Objektsegmentierung - im Gegensatz zur Stixel-Welt - sehr stark regularisiert, was Fehlmessungen prinzipigell unterdrückt. Allerdings birgt diese Regularisierung auch die Gefahr von Überregularisierung und des damit verbundenen Unterdrückens von Information. Insofern ist der Vergleich der beiden Algorithmen in diesem Testszenario sinnvoll und wichtig. Es stellt sich die Frage, wie viel Information bereits in den Daten steckt und was aufgrund der Objektsegmentierung interpretiert wird. Die Ergebnisse zeigen, dass die vorgestellte Objektsegmentierung die Zahl der Falschalarme aufgrund sehr niedriger Kollisionszeiten wesentlich um einen Faktor 70 reduziert. So wird gezeigt, dass die vermeintliche Überregularisierung in der Praxis nicht auftritt und die Objektsegmentierung einen wichtigen Baustein für ein detailliertes Szenenverständnis für das hochautonome Fahren darstellt.

Das Bertha-Benz Projekt

Es stellt sich die Frage, wie weit die vorgeschlagene Objektsegmentierung trägt. Ziel des sogenannten Bertha-Benz Projekts [Ziegler et al., 2014] war es nachzuweisen, dass es bereits 2013 mit seriennaher Sensorik möglich war, dass ein autonomes Fahrzeug auch äußerst komplexe Situationen im realen Stadtverkehr sicher und ohne menschliche Eingriffe berherrschen kann. Dazu wurde die geschichtsträchtige Strecke von Mannheim nach Pforzheim gewählt, auf der Bertha Benz und ihre Söhne im August 1888 die erste Überlandfahrt mit einem Automobil überhaupt antraten. Zu ihrem 125-jährigen Jubiläum wurde diese Strecke wiederholt, dieses Mal allerdings vollständig autonom. Das Bertha-Benz Projekt war bis dahin wohl das schwierigste und anspruchsvollste Vorhaben mit einem hochautonomen Fahrzeug. Die Arbeit und Entwicklung an diesem Projekt bildete einen wichtigen Teil dieser Arbeit. Die vorgestellte Objektsegmentierung war ein zentraler Baustein für dieses Projekt und ermöglichte es den Beteiligten, ein weiteres Kapitel der Automobilgeschichte zu schreiben.

Contents

Notation

Symbol	Description
	Camera-related Parameters
b	stereo base line in meters
f	principal distance in meters
f_x	scaled principal distance in x-direction in pixels
f_y	scaled principal distance in y-direction in pixels
d	stereo disparity in pixels
$\mathcal{D}$	disparity image
e_{P_i}	epipolar line of all points P_i
u	horizontal pixel coordinate
v	vertical pixel coordinate
a_{ij}	elements of 3×4 extrinsic camera parameter matrix
σ_u	uncertainty (standard deviation) in u-direction in pixels
σ_d	disparity uncertainty (standard deviation) in pixels
δu	optical flow along the horizontal pixel coordinate in pixels
δv	optical flow along the vertical pixel coordinate in pixels
u_{hor}	horizontal image coordinate of vanishing point in pixels
v_{hor}	vertical image coordinate of vanishing point in pixels
W	image width in pixels
H	image height in pixels
$\mathcal{W}$	image window
$\mathcal{I}$	image sequence
I	single image
$I(v,u)$	image intensity as grey value at pixel (v,u)
	Kalman filter Elements
t	time index
Δt	discrete time interval between two images in seconds
ψ	yaw angle (rotation about height axis) in ego vehicle coordinate system in rad
$\dot{\psi}$	yaw rate [rad/s]
V_{ego}	absolute ego velocity in driving direction [m/s]
a	object acceleration in $m \cdot s^{-2}$
X	lateral world coordinate in meters
Y	height-related world coordinate in meters
Z	longitudinal world coordinate in meters
P_i	3D world point

$\vec{x}$	3D state position vector
$\dot{X}$	velocity in X-direction [m/s]
$\dot{Y}$	velocity in Y-direction [m/s]
$\dot{Z}$	velocity in Z-direction [m/s]
$\vec{V}$	3D velocity vector
$\|\vec{V}\|$	length of $\vec{V}$
$\vec{h}$	vector of N height measurements
$\vec{X}$	6D state vector consisting of 3D position and 3D velocity
$\mathbf{A}$	state transition matrix
$\mathbf{H}$	measurement matrix
$\mathbf{B}$	control input vector
$\vec{\omega}$	process noise vector
$\mathbf{Q}$	process noise covariance matrix
$\vec{\gamma}$	measurement noise vector
$\mathbf{R}$	measurement noise covariance matrix
$\mathbf{R}_y(\psi)$	rotation matrix around Y-axis about Euler angle ψ
$\mathbf{0}_{n\times m}$	$n \times m$ zero matrix
$\mathbf{\Sigma}$	state covariance matrix
F	number of Kalman filter hypotheses per Stixel
f_i	filter index for Stixel i
	Segmentation-related Elements
$\mathbb{L}$	set of all possible labelings
$\mathbf{L}$	one particular labeling of all N Stixels
$p(\mathbf{L})$	probabiltiy of $\mathbf{L}$
$E(\mathbf{L})$	energy of $\mathbf{L}$
$Q(\mathbf{L})$	another probabiltiy distribution depending on $\mathbf{L}$
$\mathcal{J}$	set of possible object classes
L	one specific class from $\mathcal{J}$
J	cardinality of $\mathcal{J}$
M	number of moving objects per image
B	number of stationary objects per image
$\mathcal{F}$	number of all objects in an image
$\mathbb{Z}$	set of all possible measurements for N Stixels
$\mathcal{Z}$	one realization of $\mathbb{Z}$ for N Stixels
$\mathcal{M}$	statistical map knowledge
$m_{X,Z}$	occupancy variable for cell centered at X and Z
$\mathcal{R}$	set of Radar objects
	Graph-related Elements
$\mathcal{G}$	graph representation of random variables
$\mathcal{C}$	maximal clique in a graph
Ω	partition sum
T_i	tree rooted at node i
$\Pi(i)$	parent node of tree node i

$ne(i)$	neighbor nodes of node i
ν	integer value denoting the order of a Markov chain
$\mathcal{C}_i$	set of all possible cuts for a tree rooted at node i
C_i^*	the best (MAP) cut for a tree rooted at node i
$\lvert C^* \rvert$	cost of the minimum cut in an undirected graph
$\mathbf{C}_{V_i}$	set of children nodes of V_i
$\mathbf{V}_n$	set of Stixels in a graph
$\mathcal{E}$	number of edges in a graph
g	number of Stixel rows in a graph
$\mathcal{N}_2$	set of neighboring Stixels in the graph
	Graph cut-related Elements
Γ	theoretical optimality bound for the α-expansion algorithm
q_L	binary variable
y_i	binary variable for active variables participating in expansion move
$\mathbf{P}_L$	set of Stixels that have class L
Υ	maximum class cardinality minus one
$\mathcal{Q}$	complexity to evaluate the objective function
	Data term-related Elements
$\mathcal{A}$	second derivative of probabiltiy distribution $Q\left(\mathcal{Z}, \boldsymbol{\Theta} \mid \mathbf{L}\right)$ with respect to the parameters $\boldsymbol{\Theta}$
η	normalization constant
p_{out}	outlier probabiltiy
V_x^{min}	minimum observed lateral velocity [m/s]
V_x^{max}	maximum observed lateral velocity [m/s]
V_z^{min}	minimum observed longitudinal velocity [m/s]
V_z^{max}	maximum observed longitudinal velocity [m/s]
X^{min}	minimum observed lateral coordinate [m]
X^{max}	maximum observed lateral coordinate [m]
Z^{min}	minimum observed longitudinal coordinate [m]
Z^{max}	maximum observed longitudinal coordinate [m]
H^T	height threshold [m]
π_{bike}	bicycle mixing coefficient
$\pi_{vehicle}$	parking vehicle mixture coefficient
π_{occ}	occluded object mixture coefficient
μ_{disp}	expected disparity deviance between neighboring object [pixels]
Δl_i^t	binary variable indicating a class change between neighboring Stixels
ξ_{out}	binary variable indicating an outlier Stixel
$\Delta v_{res,i}^t$	velocity residual from zero [m/s]
ζ	binary variable describing the old labeling confidence
	Object-related Elements
n	object index
β	index for one specific background object

$\mathbf{s}_n$	1D segment describing an object
$\mathcal{U}_n^{Ref}$	image column of leftmost visible object cuboid corner point
$\mathcal{U}_n^{Ref}$	disparity of leftmost visible object cuboid corner point
$\dot{\mathcal{X}}_n$	object velocity in X-direction [m/s]
$\dot{\mathcal{Z}}_n$	object velocity in Z-direction [m/s]
$\mathcal{X}_n$	lateral object center position [m]
$\mathcal{Y}_n$	height of object center [m]
$\mathcal{Z}_n$	longitudinal object center position [m]
$\lvert\Delta\mathcal{X}\rvert_n$	object width in X-direction [m]
$\lvert\Delta\mathcal{Z}\rvert_n$	object width in Z-direction [m]
$\mathcal{H}_n$	object height [m]
c_n	current object class of object n
u_n^l	leftmost horizontal image coordinate of object n in pixels
u_n^r	rightmost horizontal image coordinate of object n in pixels
B	number of stationary objects
$\mathbf{\Theta}$	hidden object parameter vector
K	dimension of $\mathbf{\Theta}$
λ_{X_j}	inverse sigmoid slope for lateral object center coordinate [m]
λ_{Z_j}	inverse sigmoid slope for longitudinal object center coordinate [m]
$\mathcal{V}_n^{exc}$	exclusion volume of the $n-$th object $[m^3]$
$\mathrm{Corr}\,(\Theta_r)$	driving corridor for object r
	Stixel-related Elements
N	number of Stixels in an image
$\mathcal{S}$	set of Stixel indices
i	pixel or Stixel index
j	pixel or Stixel index
l_i	class of Stixel i
$\vec{z}_i$	vector of measurements for Stixel i
$\mathfrak{C}_{\mathrm{stixel}}$	Stixel confidence value
$\bar{\mathfrak{C}}_{\mathrm{stereo}}$	mean stereo confidence for a Stixel
$\mathfrak{D}_{\mathrm{stereo}}$	percentage of valid disparity measurements for a Stixel
w_{stixel}	Stixel width in pixels
	Coordinate systems
${}^c\vec{x}_i$	3D position given in default left-handed camera coordinate system
${}^o\vec{x}_i$	3D position given in left-handed object coordinate system
${}^c\mathbf{M}_o$	transformation matrix from the object coordinate system to the camera coordinate system

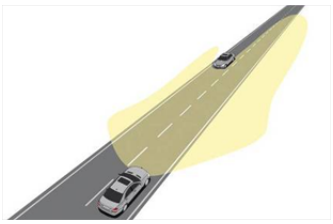
1 Introduction

1.1 Motivation

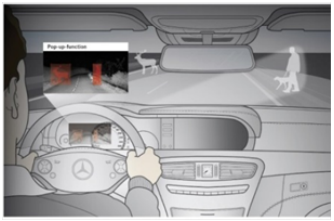
1.1.1 Why Driver Assistance?

The road to accident free driving is a key challenge for modern driver assistance systems and is the shared vision of thousands of academic and industrial researchers worldwide. Numerous driver assistance systems that are already commercially available demonstrate impressively that this way is a success story, see Figure 1.1 for an overview.
According to data collected by the Federal Office for Statistics of Germany, the introduction of the Electronic Stability Program (ESP) for example lead to a reduction of the number of accidents in which drivers lost control of their vehicles and left their lane by about 42 percent [Bundesamt für Statistik, 2009, Lie et al., 2004].
Furthermore, an analysis part of the GIDAS project (German In-Depth Accident Study) revealed that the introduction of distance control and brake assist systems (BAS) could reduce the number of head-to-tail collisions by about 36 percent [Fach and Ockel, 2009].
Further assistance systems allow for traffic sign and pedestrian recognition, autonomous parking, adaptive cruise control (ACC), lane departure warning or active lane keeping or active night view and blind spot assistance. These systems have contributed to reduce the number of road fatalities to now reach a historic low in Germany and Europe [Daimler AG, 2012], see Figure 1.2.
However, there is still a long way to go. Worldwide, two people are killed and more than 95 are seriously injured every minute in road accidents [Daimler AG, 2012]. Insofar, as even a single death is one too many, these numbers are alarming and underline not to give up its efforts to further improve safety and assistance systems in order to be able to further reduce the number of accidents worldwide.
Future driver assistance and safety systems aim to support and relieve the driver in more and more complex driving situations with fully or partly automated assistance functions. These developments realize the concept of a *smart vehicle* [Gavrila and Philomin, 1999].
Autonomous driving is currently one of the most prospering and challenging topics of driver assistance and the developed systems are definitely superb examples of engineering technology. Prominent examples are the Google self-driving car [Thrun, 2010, Erico, 2013], the Stadtpilot project [Institute of Control Engineering, 2007], Autobahnpilot [Kämpchen et al., 2012], AnnieWay [KIT, 2006], the PROMETHEUS project (Programme for a European Traffic of Highest Efficiency and Unprecedented Safety) from 1986 until 1995 [Prătorius, 1993, Williams, 1988, Dickmanns et al., 1994], the Argo Project from 1996 until 1999 [Broggi, 1999], the VisLab Intercontinental Autonomous Challenge in 2010 [Broggi et al., 2009], the European Land Robot Trial (ELROB) starting in 2006 [ELROB, 2006], the DARPA Grand Challenge in 2004 and 2005 [Thrun

(a) Traffic sign detection.

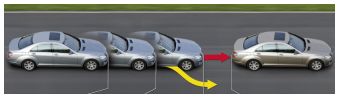


(b) Adaptive High Beam.

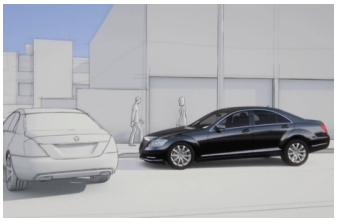


(c) NightView.

(d) Attention Assist.



(e) Pre-Crash Braking.

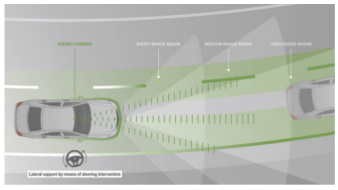


(f) Pre-Crash Braking.

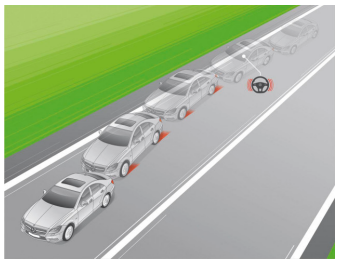
(g) Pedestrian Detection.

(h) Active Body Control.

(i) Parking.



(j) Adaptive Cruise Control.



(k) Lane Keeping.

(l) Blind Spot.

Figure 1.1: Available driver assistance systems for the new Mercedes E- and S-class 2013.

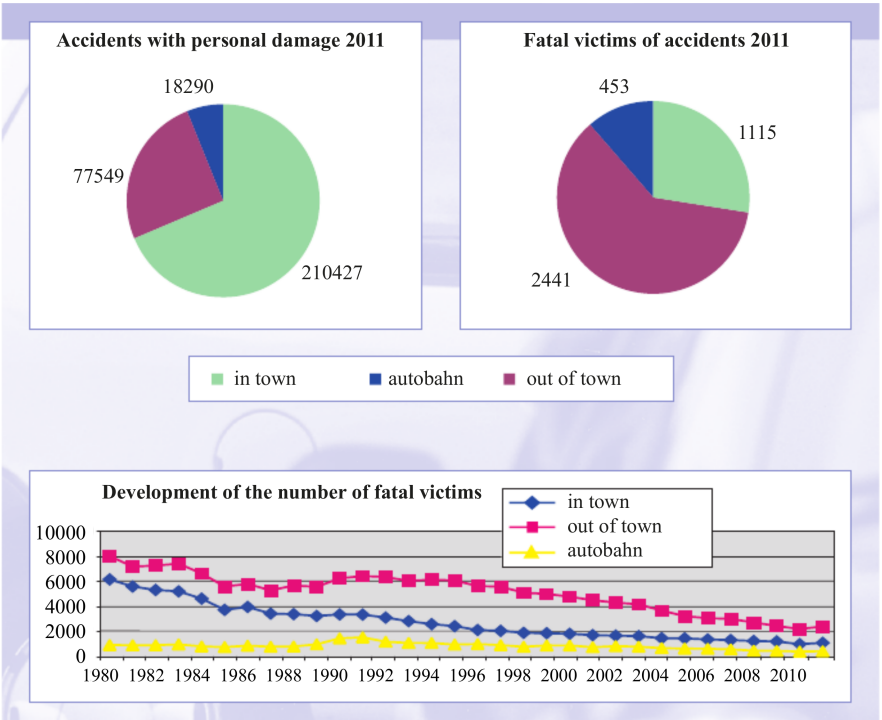


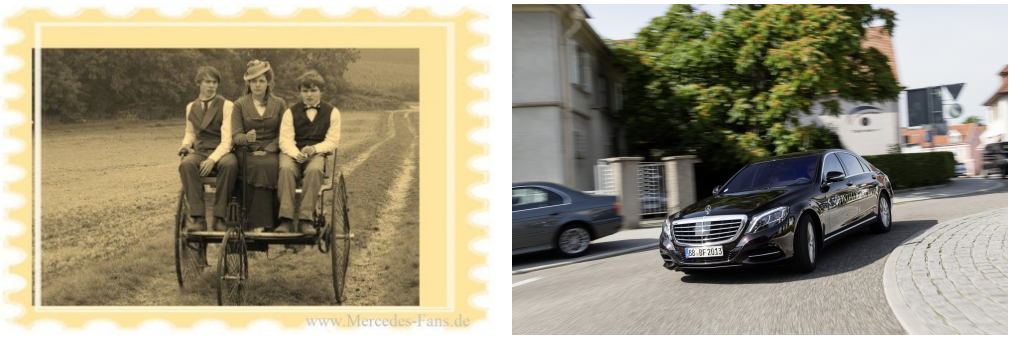
Figure 1.2: Accident statistics of Germany. Above, the absolute numbers for road accidents and road fatalities are shown. Below, the temporal development of road fatalities is given. The graph is modified taken from [ADAC, 2013].

et al., 2006] or the DARPA Urban Challenge in 2007 [Buehler et al., 2009, Montemerlo et al., 2008, Urmson et al., 2007].
For all the above mentioned projects, environment perception plays a key role. For this purpose, most of the vehicles are equipped with a whole sensor zoo like Radar, Lidar, cameras or ultrasonic sensors that image the vehicle environment. These sensors are, so to say, the "eyes" of the vehicle. All of these sensors have their individual pros and cons which will not be addressed in greater detail here. See [Faerber, 2004] for a review article on these sensors. It is important to note here that the sensors partially complement each other.
Two eyes are better than one - this old turn of phrase describes very well the basic idea of sensor fusion. Sensor fusion tries to combine the advantages of multiple sensors in a way that their individual shortcomings are reduced to a minimum. Different sensors are sensitive to different types of obstacles [Thrun et al., 2005] and partly have different characteristics. See [Heinrich, 2005, Liu et al., 2008, Mobus and Kolbe, 2004, Alessandretti et al., 2007] for a fusion of cameras and Radar, [Premebida et al., 2009,Kämpchen, 2007] for fusion of Lidar and vision and [Weiss et al., 2004] for a fusion approach of all three sensors.
Thus, in order to get a complete picture of the environment, multiple sensors are required. Only together the sensors can achieve the high level of redundancy and the low error rates that are required by the Automotive Safety Integrity Levels (ASIL), ISO 26262 [Hillenbrand, 2011].
The collected sensor data is subsequently used to explore and measure the vehicle environment and to extract the information that is required for autonomous driving including the determination of the drivable freespace, detection and tracking of moving and stationary obstacles inside and outside the planned driving corridor, self-localization, lane recognition, path-planning or detection of traffic signs and traffic lights.

1.1.2 On the Trails of Bertha Benz

In 1886, Dr. Carl Benz [1] invented the automobile in Mannheim (Reich Patent No. 37435) [Bertha Benz Memorial Club e.V., 2013] - but nobody wanted to buy it. It required the pioneering spirit and business sense of his wife, Cäcilie Bertha Benz [2], to present his horseless coach to a broad public. In 1888, Bertha took a first ride with her both sons from Mannheim to Pforzheim and back and demonstrated impressively the practicability and performance of her husband's invention. Carl was unaware of all this. Her live presentation for marketing reasons became a great success and shapes our society even today with almost a billion drivers worldwide.
In 2013, 125 years later, Daimler continued this success story and demonstrated its technology leadership by an autonomous vehicle taking the same route from Mannheim to Pforzheim [Franke et al., 2014, Ziegler et al., 2014, Dang et al., 2015]. The route from Mannheim to Pforzheim was chosen for these historical reasons. However, an expansion to further, mapped routes is at least conceivable.
This highly autonomous vehicle extends already existing autonomous functions like the Stop&Go-Pilot in traffic jams [Schopper et al., 2013] or the recently presented Highway

[1] 1844-1929
[2] 1849-1944



(a) The maiden voyage of the first automobile by Bertha Benz in 1888 [E-Mags Media GmbH, 2013].

(b) Autonomous long-distance drive in rural and urban traffic in 2013 in the tracks of Bertha Benz [Bertha Benz Memorial Club e.V., 2013].

Figure 1.3: Milestones and pioneering achievements of automotive development yesterday and today.

Pilot project [Ewing, 2013, Ingraham, 2013, English, 2012].
One of the special features of the autonomous Bertha Benz vehicle is the usage of standard sensor components that are already in series production such as Radar or stereo cameras. This contrasts with other prototypical autonomous vehicles such as the Google self-driving car [Thrun, 2010, Erico, 2013] that uses high complex and expensive laser scanners to capture the 3D environment.
The limitation to standard sensors as cameras places significantly higher demands on the used algorithms. The present work was an important component for autonomous driving in this Bertha Benz project of historic significance.

1.1.3 Why Object Segmentation?

In this work, the component of motion-based object segmentation using a stereo camera system is investigated which is a key component for autonomous driving. The object formation step is a central link between classical measurement recording and higher-level maneuver recognition and situation analysis modules. Therefore, its algorithmic output needs to be highly stable since for example a path-planning module directly depends on the output of this stage.
Typically, the preceding algorithms create a noisy image of the environment. This sensor image has to be interpreted by the object formation step which translates the cluttered observations into a higher-level object description. For this purpose, the object formation step requires a realistic sensor model and prior knowledge on the segmentation task and on the current traffic scene. In addition, a high ability to generalize to unseen traffic scenes is important. Typical traffic scenes are extremely versatile with respect to object types or inter-object constellations such as occlusions. This versatility makes it difficult to use pure classification-based approaches and in addition requires for more general concepts such as motion and 3D information.
The object formation step is important since a stable, minimal scene description in terms of objects is desirable for most subsequent algorithms. This contrasts with other

scene representations such those obtained from various tracking before detection approaches like [Franke et al., 2005, Pfeiffer and Franke, 2010, Günyel et al., 2012]. In the presence of strong noise typically it is difficult to handle their algorithmic output directly.
In the present work, a stereo camera system is used for measurement recording. A stereo camera system is a comparatively cheap 3D sensor system, compared with a Laser Scanner or a PMD (Photonic Mixing Device) camera. Besides that, a stereo camera system is characterized by a high angular resolution and its ability to capture relative object motion directly via optical flow for example. However, a stereo camera system has a limited range due to the inverse measuring principle. A stereo camera system with a realistic baseline has negligible disparity dominated by noise for 3D world points in the far field [Wojek et al., 2010]. Furthermore, usually huge amounts of data have to be processed. For example, typical dense disparity maps contain about half a million individual disparity measurements, that means that for 25 frames per second about 50 MB have to be transferred and processed per second. Processing such large amounts of data is challenging, especially when multiple algorithms have to operate simultaneously on the data and when striving for real-time capability.
In order to better handle such huge amounts of data, more efficient medium-level representations have been proposed. In the simplest case, this can be a downsampled image [Ess et al., 2009]. More precise image abstractions - since taking into account the actual image information - are superpixels as proposed in [Levinshtein et al., 2009, Achanta et al., 2010, Achanta et al., 2012, Veksler et al., 2010, Felzenszwalb and Huttenlocher, 2004] or the Stixel representation [Pfeiffer, 2012] that is used in the present work for reasons of efficiency and stability.
To sum up, object segmentation bridges the gap between noisy sensor observations and a higher-level object description that is required by subsequent algorithms. In this work, the object formation step is deliberately restricted on generic motion and 3D information, in order to be able to address as many scenarios as possible. Additional and completely orthogonal appearance information can be readily added - if required at all. This step remains for future works in this field, see [Scharwächter et al., 2013, Scharwächter et al., 2014, Cordts et al., 2014].

1.2 Thesis Contributions

In this thesis, novel real-time approaches for object segmentation of stereo image sequences from a moving platform are presented. A traffic scene is assumed to be composed of an unknown number of moving objects which can move rigidly relative to the camera coordinate system and, in addition, stationary background such as houses, crash barriers or traffic signs.
The proposed algorithms yield a complete scene description where each Stixel is assigned to exactly one of the moving object classes or to stationary background. Additionally, the introduced approaches estimate the relevant object parameters such as their pose, motion state and dimensions.
Motion-based object segmentation is highly relevant for traffic scene understanding, for

autonomous driving and it provides important information to solve further problems like SLAM and map creation [Merrell et al., 2007, Pollefeys et al., 2008], ego motion estimation [Badino, 2007], object tracking [Barth, 2010], superresolution [Zomet et al., 2001], video decomposition [Zappella et al., 2008] or compression [Khan and Shah, 2001]. However, full 3D motion information is largely not taken into account for object segmentation in literature.
Due to the measurement principle of the used stereo camera system, the object segmentation step has to be able to cope with significant noise. Insofar, regularization and redundancies are essential for improving the segmentation. In order to obtain an optimal solution, the segmentation task is formulated as a Bayesian optimization problem, which allows to incorporate regularizing prior knowledge into the segmentation.
In this work, two approaches for object segmentation are proposed, evaluated and compared. Both approaches use the Dynamic Stixel World instead of dense stereo and optical flow information, see Section 2.4 and 2.5. This step increases the stability of the object segmentation significantly and it reduces the computation time by several orders of magnitude, thus enabling real-time automotive use in the first place. Using the Stixel World for object segmentation is the *first main contribution*.
The first approach is based on dynamic programming, see Section 2.6.1 and 4. These chapters show how to take into account and how to improve segmentation results using non-local priors such as shape or object size information which is impossible or which is prohibitively expensive with more local, conventional graph optimization approaches such as graphcut or belief propagation. This insight is the *second main contribution*.
The *third main contribution* is the project work for the "Bertha Benz Memorial Drive", the autonomous drive from Mannheim to Pforzheim based on series sensors.
Further contributions are a special outlier class for stereo tear-off edges, the modeling of object occlusions and a temporal coupling of the segmentation results.

1.3 Related Work

Urban traffic scene segmentation for detecting and tracking of other moving objects has been explored by many researchers in the computer vision community over the past two decades. This section gives an overview on different related approaches. Besides camera-based segmentation, there is a vast literature devoted to the same topic using other input sensors such as Radar, Lidar or Sonar which are not dealt with here.
The field of vision-based object segmentation can be roughly categorized into *supervised* and *unsupervised* approaches. In the latter case, a distinction between different object instances such as different vehicle objects is made. Furthermore, the actual number of objects and some of their parameters is not known in advance for the unsupervised case. Based on this difference, literature can be further divided based on the specific scenario, the features used for the segmentation or based on the inference scheme.
Nevertheless, the number of directly related approaches is small. Of course, these closely related approaches will be discussed in greater detail below. During the work on this thesis, there was a low number of directly related approaches dealing with scene flow-based object segmentation for stereo image sequences, especially there were no reference implementations available for comparison. Besides that, a comparison with

already existing series software is problematic. Large open datasets like the KITTI dataset [Geiger et al., 2013] were not available yet. Comparisons are generally difficult since dealing with noise is one of the main challenges for object segmentation and this step clearly depends on an individual sensor model. Most algorithms will completely fail without a suitable noise model of the input data.
Insofar, in principle, in this section only ideas and concepts can be discussed. This section starts with a short subsection on some recent developments in the field of supervised segmentation and continues with a brief discussion of related unsupervised segmentation approaches.

1.3.1 Supervised Segmentation Approaches

Image segmentation has been tackled successfully as a multi-class labeling problem for static environments, e.g. by [Ess et al., 2009] using color and texture cues. Stereo information is used optionally here. In this work, Ess *et al.* aggregate the image content into small image squares of size 8×8 pixels which are then classified into various classes such as road, car, building, bush and sky. The output of this classification step is used as input to a second scene classification stage that distinguishes 8 different types of road layouts and detects the presence of cars or pedestrians in front of the ego vehicle. Each patch is classified independently of all other patches and for each patch the label of the classifier with the maximum response is chosen. The second scene classification stage has to compensate for the noisy output of the first stage. To sum up, Ess *et al.* present an interesting concept for scene classification and towards traffic scene understanding, especially the scene classification step tries to bridge the gap between local patch classification and object and scene detection. However, it remains unclear how this concept generalizes to further traffic scenes. What is a meaningful subdivision or categorization of traffic scenes? Secondly, an experimental comparison with other state of the art methods for object tracking of a leading vehicle or optical lane tracking would be desirable. How is the accuracy of this method influenced by the rough image quantization? Thirdly, the independence assumption of the local image patches seems questionable. The underlying scene layout rather seems to be a hidden variable that should influence the patch classification as well.
Similarly, [Brostow et al., 2008, Brostow et al., 2009, Xiao and Quan, 2009] perform a pixel-wise labeling into similar object classes as mentioned above, however without the scene classification step. These approaches additionally take into account structure from motion derived 3D world information such as height, the distance to the camera path, surface orientation, feature track density or a residual reconstruction error. All of these approaches are also limited to classify static scenes. [Brostow et al., 2008] classifies each pixel independently using randomized decision trees, whereas [Brostow et al., 2009] applies a quality-sensitive higher-order CRF [Kohli et al., 2009]. As usual, the higher-order information is taken from unsupervised mean shift segmentations. Both approaches are pioneering work with impressive results.
Recently, there is a trend to use superpixels for image segmentation, e.g. in [Ladicky et al., 2009, Micusik and Kosecka, 2009, Xiao and Quan, 2009, Zhang et al., 2010]. In [Xiao and Quan, 2009, Zhang et al., 2010], the authors propose to use superpixels for segmentation in order to obtain more discriminating features defined over larger image

regions and to reduce the computational costs. The work of [Zhang et al., 2010] is solely based on dense depth information obtained via structure from motion. It uses five simple depth-related cues, namely the surface-normal, local and global planarity, height above the ground and the distance to the camera path. The output of a randomized decision forest is segmented by means of graphcut. This way, detection of some extreme classes such as sky, road and building works quite well, but other classes such as pedestrians, bicyclists, cars or fences perform worse than in the compared approach of [Brostow et al., 2008] that additionally takes into account appearance information.
The work by [Micusik and Kosecka, 2009] uses appearance information taking into account visual words co-occurrence statistics enriched with geometric height information. The idea of the co-occurrence statistics is to learn the mutual spatial appearance of visual words. In the simplest case, this can include ordering-constraints such as sky is always above the road. This way, the performance of [Brostow et al., 2008] could be improved significantly.
In [Ladickỳ et al., 2012], stereo reconstruction and object class labeling are solved jointly. It is assumed that there is a helpful coupling between both problems given by the object height. Besides that, a joint pairwise interaction is proposed, namely an object classes boundary is more likely to occur if the disparity of two neighboring pixels differs significantly. This method was shown to outperform a stand-alone stereo reconstruction result by a margin of up to 25%. The performance gain for the segmentation step was negligible. The inference is performed in an alternating manner between the disparity space $\mathbb{D}$ for fixed labeling $\mathbf{L}$ and vice versa. This way, a single optimization step takes only $\mathcal{O}\left(|\mathbb{D}| + |\mathbf{L}|\right)$ graphcut steps instead of the naive α-expansion complexity $\mathcal{O}\left(|\mathbb{D}| \cdot |\mathbf{L}|\right)$ which would yield a nearly optimal solution. Here $|.|$ denotes the dimensionality of the respective feature space, e.g. the number of object classes or disparity values. The concept of joint optimization has been extended recently to simultaneous human segmentation, depth and pose estimation [Sheasby et al., 2012]. In this work, inference is performed via Dual Decomposition [Dantzig and Wolfe, 1960, Boyd et al., 2007].
[Floros and Leibe, 2012] proposed two extensions to [Brostow et al., 2008], namely a temporal smoothing via a sliding window higher-order P^N potential [Kohli et al., 2009] and local geometry potentials. The latter describe the geometry of the neighboring pixels via different elongation and eccentricity measures. This term is also taken into account by means of a P^N potential. On average, the temporal coupling leads to a small improvement, the benefit from the 3D potential is almost undetectable. By modeling the temporal coupling via a higher-order potential, the computational load rises significantly since this step requires the inference over several frames simultaneously. The computation time is not discussed in the paper. Furthermore, the method of [Ladickỳ et al., 2012] outperforms the approach of Floros *et al.* .
[Lempitsky et al., 2012] introduce a branch-and-bound extension to the original graphcut algorithm. This way, it becomes possible to introduce a shape prior or a color-distribution prior into the segmentation. The approach of [Cremers et al., 2008] is similar but is based on convex variational image segmentation. For object parameter domains that can be hierarchically clustered, the proposed algorithms find the global optimum. The assumption that object parameter space can be clustered hierarchically might be questionable in many cases. The approach of [Lempitsky et al., 2012] is much

better than an exhaustive search, but it is not real-time capable (in the order of seconds on a modern CPU). Furthermore, the approach is limited to one foreground object. Complexity would rise exponentially in the number of foreground objects.
Recently, dynamic programming [Bellman, 1954] has been rediscovered for image segmentation. A major advantage of dynamic programming is the fact that it does not require the underlying energy function to be submodular as in the case of graphcut. [Felzenszwalb and Zabih, 2011] point out differences and commonalities between graphcut and dynamic programming. [Felzenszwalb and Veksler, 2010] were probably the first who used dynamic programming to perform inference for tiered scenes consisting of ground, object and sky. However, the approach is limited to one object per column. It is difficult to extend this approach to multiple layers since complexity grows exponentially in the number of tiers. [Zheng et al., 2012] have proposed an approximation that additionally uses the more efficient generalized distance transform [Felzenszwalb and Huttenlocher, 2012], [Crandall et al., 2012] use an iterative optimization strategy, [Veksler, 2012] introduce an α-expansion-like approximation based on dynamic programming and [Strekalovskiy and Cremers, 2011] generalized the result of Felzenszwalb to multiple tiers by means of a relaxation of an integer convex program, but with an even greater time complexity than [Felzenszwalb and Veksler, 2010]. Furthermore this approach is also approximate due to randomized rounding. In [Veksler, 2005], Veksler introduces an efficient stereo algorithm based on dynamic programming on a tree. The algorithm keeps the most important edges of the usual 4-connected neighborhood. The most important edges are defined via gray value similarity of neighboring pixels or according to the depth of neighboring pixels p and q inside a homogeneous intensity region which can be computed efficiently via distance transform [Borgefors, 1986]. The proposed approach yields inferior results in comparison with state of the art stereo optimization methods such as graphcut or belief propagation [Scharstein and Szeliski, 2002], but it is far more efficient. The main difficulty is probably the information loss due to discarding various edges. Suppose there is an image of size $N \times N$ pixels. Choosing a four-connected grid in the image as neighborhood system yields an overall amount of about $2N^2$ edges. In comparison, the tree structure keeps about N^2 edges. Since the underlying energy function is the same for both cases, it is comprehensible that the tree decomposition performs worse.
Similarly, in [Deng and Lin, 2006] a tree decomposition based on dynamic programming for stereo reconstruction has been proposed that operates on individual line segments and [Bleyer and Gelautz, 2008] optimize a whole tree for each pixel by means of dynamic programming.
Finally, [Liu et al., 2011] ambulate a whole different way for segmentation based on an image retrieval and label transfer framework via GIST and dense SIFT correspondences. In this approach, a test image is compared with a large number of database images. The best matches in the database are found via GIST and SIFT matching and the system warps the annotations of the database images to the query image. Unfortunately, the proposed approach is not real-time capable: The SIFT feature matching turns out to be the time critical part, it takes about 31 seconds for two images of size 256x256 pixels with coarse to fine matching. Furthermore, the approach requires for a huge database with annotated images for matching.

1.3.2 Unsupervised Segmentation Approaches

For the unsupervised approaches presented in this section, the exact number of objects does not need to be known in advance. This characteristic is important for motion-based object segmentation.
There are so-called vertical extensions to the original graphcut algorithm introduced by [Feng et al., 2010]. In this work, an arbitrary number of classes can be found. Feature space is split and merged repeatedly via K-Means clustering [Steinhaus, 1956] and the usual graphcut algorithm performs the labeling into both classes that correspond to the cluster centers. The introduced method is shown to outperform similar methods with respect to efficiency and accuracy. However, the obtained results typically show oversegmentation since the data terms are too restrictive. Furthermore, since relying on an iterative method, the approach is susceptible for local minima.
[Delong et al., 2012] propose an extension to the α-expansion graphcut algorithm to choose the correct number of classes based on a MDL prior [Zhu and Yuille, 1996]. In this contribution, the interesting application scenario of geometric multi-model fitting is investigated. For this purpose, a huge number of initial hypotheses is provided. The number of classes grows exponentially with the feature space dimension. Insofar, this approach is not an option for the high-dimensional optimization problem that is investigated in the present work.
A similar approach to [Delong et al., 2012] is investigated by [Yuan and Boykov, 2010] based on a problem formulation in the continuous domain. The resulting convex optimization problem is solved via second order cone programming (SOCP) [Sturm, 1999]. However, the authors conclude that their proposed algorithm is still too slow and that a fast algorithm to solve the MDL segmentation problem is missing.
In [Wojek and Schiele, 2008], the authors present a two-layered graphical model for combining the object detections from a classifier with static classes as street, building, sky etc. The classifier detections are represented by higher-order nodes in their underlying graphical model. Besides a HOG-based object detector, various texture and location features are considered. Moving objects are tracked with an extended Kalman filter and the predicted vehicle position is taken as a temporal segmentation prior. The authors stress that information flow is bidirectional, that is segmentation profits from the object detector and vice versa. For inference, loopy belief propagation is employed. The actual computing time is not discussed in this paper. The approach is a practical extension to introduce higher-order object detections into a CRF that, in return, can yield a pixel accurate segmentation.
Moreover, in [Wojek et al., 2010], the same authors introduce a related monocular approach for segmentation and tracking in multi-object scenes. This time, inference is performed with a Markov-Chain-Monte-Carlo (MCMC) framework [Green, 1995] to simulate the posterior distribution. The inference step takes roughly 1 second per image on recent hardware. Results are rather noisy in that work, the strength of the approach basically originates from the object detectors and from the tracking.
Similarly, [Ladickỳ et al., 2010] couple object detectors and CRFs for traffic scene labeling. In principle, this is the same approach as [Wojek and Schiele, 2008] or [Wojek et al., 2010], but without the tracking and scene dynamics part since focus is more on static segmentation. Motion information is not taken into account. However, the

approach of [Ladickỳ et al., 2010] uses a different, more efficient inference method. The object detector responses are taken into account as higher-order P^N nodes [Kohli et al., 2009] in the CRF. The stand-alone performance of the object detectors is not reported. On average, segmentation is reported to profit from taking into account the classifier information.
Similar work has also been done before for example by [Gu et al., 2009, Gould et al., 2009, Winn and Shotton, 2006].
[Felzenszwalb and Huttenlocher, 2004] introduce a region segmentation algorithm based on greedy decisions. Briefly the algorithm proceeds as follows: As usual, the algorithm represents an input image as a graph where each pixel represents one vertex in this graph, the edges are constructed by connecting pairs of pixels that are neighbors in the sense of an 8-connected neighborhood. Initially, each pixel is in its own component. The algorithm then sorts all edges according to their weight where the weight of an edge is defined by the absolute intensity difference between the pixels. Broadly speaking, the approach then merges different components if they are similar enough based on their gray value difference. The merging threshold is adapted during the algorithm and depends on the component properties. The algorithm is very efficient ($\mathcal{O}(\mathcal{E} \cdot \log \mathcal{E})$, where $\mathcal{E}$ denotes the number of edges in the image), but it lacks any optimality properties. It is based on local decisions and it lacks a robust formulation [3] but it can take into account object properties such as object shapes.
In [Unger et al., 2012], a continuous approach for joint motion segmentation and optical flow calculation is presented. The optimization is based on an alternating, iterative computation of the optical flow field and of motion segmentation. The segmentation step tries to minimize the perimeter of the segments, a similarity measure for the optical flow vectors inside each segment and simultaneously tries to keep the number of labels small via a label cost term. The motion field inside each segment is described by a linear operator. Typically, about one hundred segments are found per image, the exact number is influenced by the weight term of the label costs. In general, the segments are not objects. The algorithm does not impose any particular topological constraints on the segments explicitly in the optimization, but instead each segment is split into spatially connected regions and regions with less than 10 pixels are skipped. This step is based on heuristics. The computation time is up to one hour on a modern GPU. In general, the algorithm converges to a local optimum. For that reason, a good initialization close to the optimum is required which limits the approach.
Very recently, [Schulter et al., 2013] have presented a monocular optical flow-based approach for unsupervised object discovery. The approach detects moving objects based on the magnitude of their optical flow vectors. The ego motion is taken into account by estimating an affine model of the camera movement based on the optical flow at the image border via RANSAC [Fischler and Bolles, 1981]. A bounding box for all moving objects is estimated. Optionally, the moving objects can be classified afterwards into a predefined set of classes. The whole approach is formulated as a Markov Random Field, however the object extraction step lacks any probabilistic formulation. Depth information for motion estimation is not taken into account. Altogether, results are

[3] Unfortunately, a robust formulation would make the problem NP-hard [Felzenszwalb and Huttenlocher, 2004].

quite promising.
[Lempitsky et al., 2010] have introduced a Fusion-Move optimization algorithm that can optimize continuous MRFs. [Trobin et al., 2008] is similar but it is a full continuous, variational approach. In [Lempitsky et al., 2010], always two solutions for the optical flow problem are fused using the so-called QPBO graphcut algorithm [Rother et al., 2007] that can theoretically optimize non-submodular energy functions. Insofar this is a continuous-discrete approach. For the optical flow estimation, about 200 suboptimal solutions with their individual pros and cons are combined optimally by minimizing a single energy function. After each proposal is visited once, the obtained solution is clustered into 64 clusters using K-Means to add new proposals and to avoid local minima. In a subsequent second stage of the optimization, a continuous optimization step (conjugate gradients applied in a coarse-to-fine manner) is performed that helps to diversify the proposal solutions, which may be required for relatively smooth areas. The overall approach is rather expensive. The adequacy of the overall energy function is not discussed. However, using multiple optimization strategies suggests that it is not possible to specify or to optimize a single objective function that describes the optical flow estimation problem adequately.
In [Bachmann, 2009, Bachmann, 2010], an EM-like approach for traffic scene segmentation is introduced. The approach formulates the tasks of scene reconstruction, motion estimation and image segmentation in a joint model and solves them in an iterative way. The computing time is around 2 to 5 seconds per image of size 512×384 pixels, depending on the number of object hypotheses. The number of objects is determined heuristically.
[Barth et al., 2010] describe a Conditional Random Field approach for multi-class traffic scene segmentation based on dense depth and motion information. The approach does not estimate the motion state of the moving objects, this information has to be provided externally. Ordering constraints such as sky is above the street are reported to improve segmentation results considerably.
[Sun et al., 2012] have introduced a layered segmentation and optical flow estimation approach. The goal of this approach is to estimate the number of layers in a scene, to reason about their depth ordering (visibility) and to infer the optical flow for each layer. Temporal smoothness is enforced on the basis of a constant layer visibility over several frames which increases performance significantly. The estimation problem under consideration is formulated as a CRF, inference is done with a series of non-standard graphcut moves based on the QPBO algorithm [Kolmogorov and Rother, 2007]. Furthermore, continuous optimization is applied similar to [Lempitsky et al., 2010] for optical flow refinement. The algorithm takes about 5 hours to compute one image of size 640×480 pixels which makes it impractical for real-time application. Furthermore, layers describe affine flow areas and not objects. Nevertheless, this modeling yields excellent optical flow results.
Furthermore, there is a large number of clustering-based methods for object detection and formation using sparse or dense scene flow, see e.g. [Lenz et al., 2011, Muffert et al., 2012, Guevara et al., 2012]. Typically, these methods can be applied or can be easily adapted to a large number of scenarios. However, there is no real regularization in these methods. For this reason, in the present work a different approach is developed.

1.4 Organization of the Thesis

The remainder of this thesis is organized as follows. Chapter 2 introduces the algorithmic pre-processing steps of the segmentation as far as they are required to understand this thesis. This includes a brief, more general introduction to dense stereo reconstruction 2.1 and to the used Semi-Global Matching (SGM) stereo algorithm 2.2, a brief discussion of optical flow estimation 2.3 as well as a discussion of the Stixel extraction 2.4 and of Stixel motion estimation based on Kalman filtering 2.5. This chapter ends with an introduction and a discussion of the optimization methods that are applied for inference in this thesis, namely dynamic programming 2.6.1 and multi-class graphcut optimization 2.6.2.
The proposed segmentation approaches are presented in Chapter 3 and 4. Chapter 3 presents a graphcut extension for unsupervised motion-based object segmentation and Chapter 4 introduces a dynamic programming-based approach. Section 4.6 gives a prospect on a possible extension of the dynamic programming-based approach to multiple Stixel rows.
Finally, Chapter 5 gives an outlook on future research as well as the conclusions of this thesis.

2 Technical Background

2.1 Stereo Vision

Intelligent driver assistance systems can perceive the vehicle environment and based on this input they can interpret the current traffic situation. For this purpose, they require a detailed knowledge about depth and motion in the respective scene.
The 3D information can in principle be extracted using a wide range of sensors with different advantages and disadvantages, including Sonar (Sound navigation and ranging) [Elfes, 1987, Moravec and Elfes, 1985], Lidar (Light detection and ranging) [Steinemann et al., 2012, Spies and Spies, 2006, Schütz et al., 2012], Radar (Radio detection and ranging) [Schneider, 2005, Homm et al., 2010, Schmid et al., 2010] or PMD camera (Photonic Mixing Device) [Park et al., 2011, Dal Mutto et al., 2012] based sensors. These are all so-called *active sensors* that transmit electromagnetic waves. The distinction is made based on the wavelength range of the emitted electromagnetic spectrum. See [Faerber, 2004] for a review article on these *active sensors*.
For camera systems, various techniques can be applied to reconstruct the depth information that is lost due to the projection on the image plane. Examples include depth from time-of-flight [Cualain et al., 2007, Hussmann et al., 2008], depth from (active) triangulation [Lorenz, 1985, Gehrig et al., 2009], depth from phase [Jähne, 2005], depth from focus [Cualain et al., 2007], shape from shading [Horn and Brooks, 1986, Galliani et al., 2012], depth from multiple projections (tomography) [Kalender, 2011, Buzug, 2012], structure from motion [Häming and Peters, 2010, Zhang et al., 2010] or structure from texture [Ikeuchi, 1984, Witkin, 1981, Aloimonos and Swain, 1988].
The focus of the present work is on stereo vision, a special case of the depth determination by triangulation. In this case, depth information of a scene is inferred from two or more images taken from different viewpoints. Humans and many predatory animals directly exploit this mechanism of spatial vision for near-range depth estimation [Goldstein, 2010]. A schematic view of a typical setup is shown in Figure 2.1.
In this illustration, the optical axes of both cameras are in parallel and displaced by a translational component in only one single direction, in this case along the X-axis. This displacement is referred to as *baseline* b of this stereo camera system. The advantage of this so-called ideal stereo configuration is that a 3D point is projected on the same image row in both camera images. This fact reduces the correspondence analysis of both images to a one-dimensional search problem along the so-called epipolar lines e_{p_i} which correspond to the image rows in this ideal case, see Figure 2.2. The displacement of corresponding image points between both camera images is known as *parallax* or *disparity* d. It is calculated from Figure 2.1

$$d = u_R - u_L = f_x \cdot \frac{\Delta X + b/2}{\Delta Z} - f_x \cdot \frac{\Delta X - b/2}{\Delta Z} = \frac{f_x \cdot b}{\Delta Z}. \tag{2.1}$$

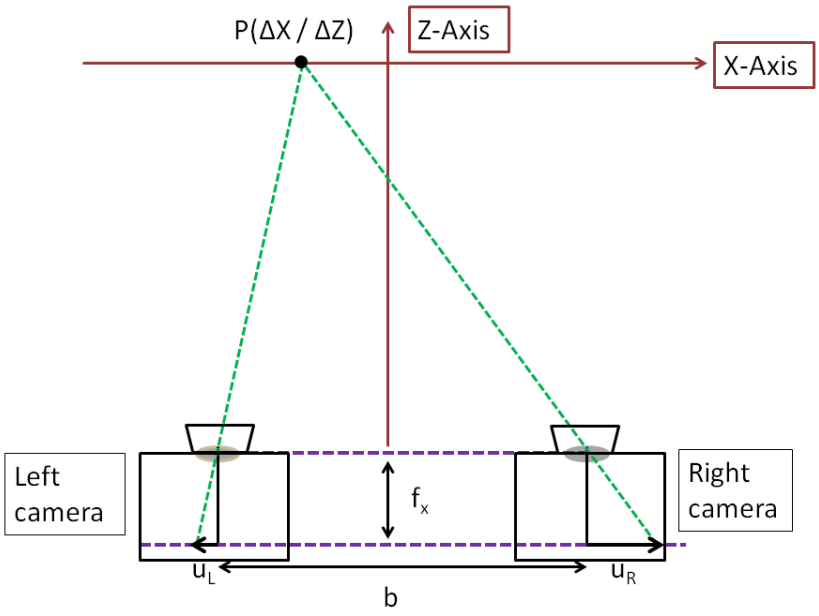


Figure 2.1: Ideal stereo configuration. Both cameras are separated in one direction by a distance referred to as baseline b. The image planes of both cameras are in parallel. A point P is projected to slightly different positions u_L and u_R in both image planes. The distance $u_R - u_L$ is designated disparity d.

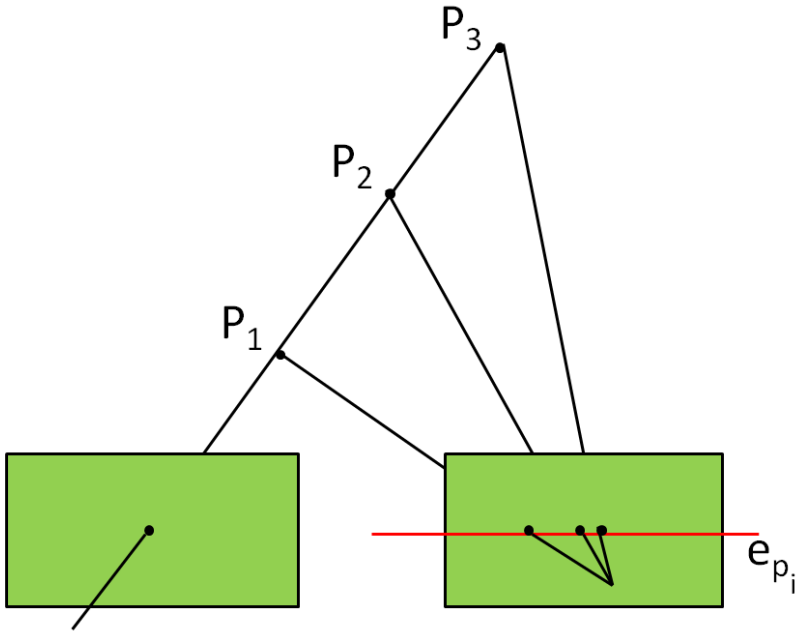


Figure 2.2: All points P_1, P_2 and P_3 along the viewing ray of the left image are projected onto the epipolar line e_{P_i} in the right camera.

In general, both cameras are displaced by a general translation and rotation. In this case, the transformation between both cameras has to be determined. This step is referred to as *calibration* of the stereo camera system. For a calibrated system with known *extrinsic* parameters, it is possible to transfer both images into this ideal stereo configuration by warping both images. This process is called *rectification*.
Finally, from Equation 2.1 it becomes clear that for a known disparity d, the actual distance $Z = \Delta Z$ is given by

$$Z = \frac{f_x \cdot b}{d}, \tag{2.2}$$

where f_x denotes the focal length of the cameras.

2.2 Semi-Global Matching

As mentioned in Section 2.1, stereo depth estimation can be understood as a one-dimensional correspondence analysis problem.
Classical stereo approaches search for these correspondences by exploiting a constant brightness assumption and search for pixels or small patches with similar intensity in both images. However, such an approach has several drawbacks: Firstly, there might be multiple pixels with the same intensity due to periodic structures or weak texture. Secondly, the observed intensity is directly subjected to noise. Thirdly, some areas in the scene cannot be observed by both cameras due to occlusions. So, in practice, the stereo estimation problem does not always have a unique solution. Further boundary conditions, e.g. by introducing a support region around the pixel of interest or by imposing a smooth solution make the stereo estimation problem solvable in the first place. For these reasons, modern stereo algorithms usually cast the stereo matching problem as a global energy optimization problem. Such global methods take into consideration not only local statistics, but also constraints defined over larger regions such as smoothness or ordering constraints. This way, they are able to estimate a depth information for almost all pixels of an image, see [Scharstein and Szeliski, 2002, Brown et al., 2003] for a review and Figure 2.3 for two examples.
A very efficient stereo optimization method has been introduced by
[Hirschmuller, 2005]. The proposed Semi-Global Matching (SGM) algorithm yields dense disparity maps and works in a dynamic programming-like fashion [Bellman, 1954] by optimizing only along a finite set of scanlines, see Figure 2.4 for an illustration.
Recently, the SGM approach has been improved by [Hirschmuller and Gehrig, 2009] to be more insensitive to decalibration and violations of the constant brightness assumption.
Besides that, [Hermann and Klette, 2012] have introduced an iterative SGM variant that is less susceptible to blurring at strong depth edges, called foreground fattening. In this case, the reconstruction of object borders is refined.
In [Hirschmuller and Scharstein, 2009], the authors propose several cost functions to robustify the SGM algorithm with respect to varying illumination conditions.
Several SGM variants have been proposed that allow for real-time operation, including FPGA [Gehrig et al., 2009], GPU [Haller and Nedevschi, 2010] and CPU [Gehrig and Rabe, 2010] solutions.

Figure 2.3: Comparison of a sparse correlation-based disparity map shown above and a dense disparity map obtained from SGM below. The sparse correlation stereo is much noisier with clear blob-like reconstruction errors in the upper part of the image, SGM performs significantly better. The color encodes the disparity (red=large, green=small).

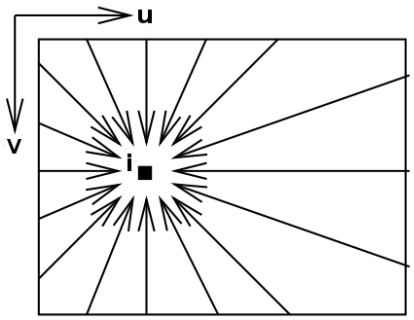


Figure 2.4: Principle of optimization of SGM. For each pixel i, the costs along a certain number of scanlines is accumulated.

Further global optimization methods for the stereo estimation problem are numerous. These include, among others, dynamic programming [Ohta and Kanade, 1985, Veksler, 2005], loopy belief propagation [Yu et al., 2007, Sun et al., 2003, Tappen and Freeman, 2003, Felzenszwalb and Huttenlocher, 2006, Meltzer et al., 2005, Yang et al., 2010], graphcut [Tappen and Freeman, 2003, Kolmogorov and Zabin, 2004, Hong and Chen, 2004], iterated conditional modes (ICM) [Besag, 1986], simulated annealing [Kirkpatrick et al., 1983, Černỳ, 1985] or mean field annealing [Peterson and Söderberg, 1989], graduated non-convexity (GNC) [Blake and Zisserman, 2012], genetic algorithms [Holland, 1975, Goldberg and Holland, 1988] convex relaxations [Kumar and Torr, 2008, Rother et al., 2007], tree decompositions [Halin, 1976, Dawid et al., 2007, Wainwright et al., 2005], non-linear diffusion [Scharstein and Szeliski, 1998], dense correlation [Mühlmann et al., 2002] or total variation [Ranftl et al., 2012].

2.3 Optical Flow Estimation

Motion information constitutes an essential source of information for understanding of traffic scenes. Optical flow denominates the projection of a three-dimensional motion field onto the image plane, induced by the movement of individual objects and possibly due to the movement of the camera. The approach to estimate the movement of the objects from optical flow observations is a classical *inverse problem* [Aster et al., 2013], which tries to infer model parameters or the "cause" from observed measurements, the "effect".
The presented segmentation approaches in this thesis take into account the full three-dimensional movement of small image patches, called Stixels, see Section 2.5. The optical flow is not sufficient to determine this full motion information. Full motion estimation has to take into account optical flow *plus* depth motion that is extracted using any of the 3D reconstruction approaches introduced in Section 2.1. This full three-dimensional motion estimation is referred to as *scene flow* [Vedula et al., 1999].
Hence, since motion information of the Stixels is partly extracted via optical flow information, it is important to understand the principles and limits of optical flow estimation in order to know failure scenarios and to be able to specify uncertainty measures.
While other sensors like Radar systems can measure the relative movement of other traffic participants directly via Doppler Shift frequency modulation [Skolnik, 1962], motion estimation using camera systems is mostly casted as a two-dimensional correspondence analysis problem. In contrast, the stereo matching problem introduced in Section 2.1 is only a one-dimensional correspondence problem along the so-called epipolar lines.
In the simplest case, the underlying brightness constancy assumption is given by

$$I\left(u, v, t\right) = I\left(u + \delta u, v + \delta v, t + \delta t\right), \tag{2.3}$$

where I denotes the apparent image intensity at pixels (u, v) and $(u + \delta u, v + \delta v)$ for two consecutive images captured at time t and $t + \delta t$ and $(\delta u, \delta v)^T$ is the unknown image displacement or optical flow vector. In practice, the brightness constancy assumption is often violated, e.g. due to illumination changes in the scene. However, the constant brightness assumption forms the basis for most optical flow algorithms. More robust alternatives will be addressed shortly below.

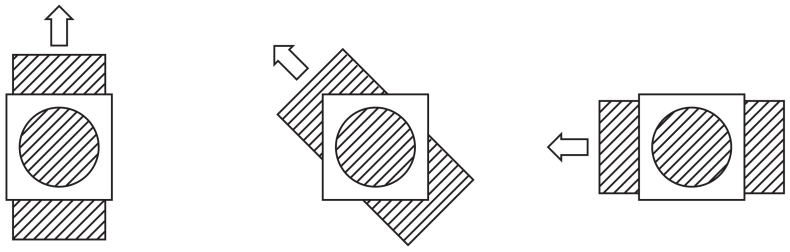


Figure 2.5: Independent of the actual motion direction, the grating structure looks identical when viewed through the aperture [Blakemore et al., 1990]. This phenomenon is widely known as *aperture problem*.

Equation 2.3 only yields one equation for two unknowns $(\delta u,\, \delta v)^T$, so the solution is not unique, see Figure 2.5. Therefore, an additional constraint is needed in order to be able to determine the optical flow. Optical flow methods can be roughly partitioned into two groups how to obtain this additional constraint: sparse, correlation-based methods and dense, variational flow reconstruction approaches.
Modern dense variational approaches solve for the optical flow field by minimizing a suitable energy functional. In the pioneering work of [Horn and Schunck, 1981], an additional smoothness constraint on the optical flow field was imposed. This way, an optical flow vector can be estimated for almost all pixels in the image (besides occluded regions). Similarly, Bayesian techniques utilize probabilistic smoothness constraints, usually in the form of discrete Markov Random Fields [Konrad and Dubois, 1991, Iu, 1995]. The pioneering work of Horn and Schunck has been improved by imposing the discontinuity-preserving TV-L1 norm [Zach et al., 2007] instead of the over-smoothing L2 norm by [Horn and Schunck, 1981]. Currently, the best berforming optical flow methods like [Vogel et al., 2015] regularize over larger patches assuming a scene composed of rigidly moving planar regions, thus they basically couple segmentation and flow estimation.
In the present work, a more efficient sparse optical flow method is applied. Such correlation-based block matching techniques try to overcome the previously described *aperture problem* by assuming that all pixels in a small block undergo the same motion [Gharavi and Mills, 1990, Liu and Zaccarin, 1993]. For this purpose, the region translation is estimated by minimizing an error function like the Zero-mean Sum of Squared Differences (ZSSD) or the Zero-mean Sum of Absolute Differences (ZSAD) of the intensities inside the small block [Giachetti, 2000]. For example, the ZSSD error function is defined as

$$\epsilon_{\text{ZSSD}} = \sum_{\mathcal{W}} \Big(\Big(I\,(u, v,\, t) - \bar{I}\,(u, v,\, t) \Big) - \Big(I\,(u + \delta u,\, v + \delta v,\, t + \delta t) - \bar{I}\,(u + \delta u,\, v + \delta v,\, t + \delta t) \Big) \Big)^2, \tag{2.4}$$

where $\bar{I}\,(u, v,\, t)$ and $\bar{I}\,(u + \delta u,\, v + \delta v,\, t + \delta t)$ denote the mean intensities in both consecutive images in the analyzed window $\mathcal{W}$. This error function is more robust with respect to global illumination changes.

Figure 2.6: Sparse KLT optical flow result based on the ZSSD matching criterion.

Besides this similarity measure, the census transformation has been used successfully as an illumination-robust cost metrics, both for sparse optical flow reconstruction [Stein, 2004] and for dense flow fields [Müller et al., 2011]. The signature-based method from [Stein, 2004] can additionally cope with, theoretically, arbitrarily large displacements in the image plane using efficient hash table indexing.
In the present thesis, the well-known KLT tracker proposed by Kanade, Lucas and Tomasi [Shi and Tomasi, 1994, Tomasi and Kanade, 1991, Lucas et al., 1981] is used, see Figure 2.6 for an example. By approximating the intensity function at $t + \delta t$ by means of a first order Taylor expansion, the brightness constancy assumption 2.3 becomes

$$\begin{aligned} I(u, v, t) &\overset{2.3}{=} I(u + \delta u, v + \delta v, t + \delta t) \\ &\approx I(u, v, t) + \frac{\partial I}{\partial u}\delta u + \frac{\partial I}{\partial v}\delta v + \frac{\partial I}{\partial t}\delta t, \end{aligned} \tag{2.5}$$

resulting in the famous *optical flow constraint*

$$\nabla I(u, v, t)^T \begin{pmatrix} \delta u \\ \delta v \end{pmatrix} + \frac{\partial I}{\partial t}\delta t = 0, \tag{2.6}$$

where $\nabla I(u, v, t) = \left(\frac{\partial I}{\partial u}, \frac{\partial I}{\partial v}\right)^T$ is a short notation for the intensity gradient.
Lucas and Kanade assumed the image motion inside a small window $\mathcal{W}$ to be constant, resulting in the modified optical flow constraint

$$\left(\sum_{\mathcal{W}} w(u, v) \nabla I(u, v, t)^T\right) \begin{pmatrix} \delta u \\ \delta v \end{pmatrix} = -\sum_{\mathcal{W}} w(u, v) \frac{\partial I}{\partial t}\delta t, \tag{2.7}$$

with $w(u, v)$ as a weight function. The solution can - formally - be obtained as

$$\begin{aligned} \begin{pmatrix} \delta u \\ \delta v \end{pmatrix} = &\begin{pmatrix} \sum w(\mathbf{x})^2 \frac{\partial I}{\partial u}\frac{\partial I}{\partial u} & \sum w(\mathbf{x})^2 \frac{\partial I}{\partial u}\frac{\partial I}{\partial v} \\ \sum w(\mathbf{x})^2 \frac{\partial I}{\partial v}\frac{\partial I}{\partial u} & \sum w(\mathbf{x})^2 \frac{\partial I}{\partial v}\frac{\partial I}{\partial v} \end{pmatrix}^{-1} \\ &\begin{pmatrix} \sum w(\mathbf{x})^2 \frac{\partial I}{\partial u}\frac{\partial I}{\partial t}\delta t \\ \sum w(\mathbf{x})^2 \frac{\partial I}{\partial v}\frac{\partial I}{\partial t}\delta t \end{pmatrix}. \end{aligned} \tag{2.8}$$

It becomes clear based on Equation 2.8 that the optical flow can only be determined if the matrix to be inverted is non-singular, i.e. the eigenvalues of this matrix have to be non-zero, see Figure 2.5 for a visualization. The matrix defined in Equation 2.8 is also referred to as *structure tensor* [Jähne, 2005]. For that reason, Tomasi and Kanade proposed to consider only those regions for optical flow estimation where both eigenvalues of the structure tensor are above a given threshold [Tomasi and Kanade, 1991]. Since the upper limit of the larger eigenvalue of the structure tensor is defined by the discretization of the gray values, only the smaller eigenvalue has to be analyzed [Rabe, 2011].
The Taylor expansion in Equation 2.5 only holds for very small displacements, so several tricks are applied to extend the scope of this approximation.
Firstly, the linearization is performed iteratively until convergence.
Secondly, the displacement is estimated at multiple scales (pyramid levels) of the input images.
Thirdly, the optical flow vectors for the KLT tracker can be pre-initialized for example based on prior knowledge on the expected ego-motion-induced displacements.
However, none of these concepts proves to be sufficient in practice. Large displacements ($\geq$ 30 pixels) as they often occur in driver assistance applications cannot be handled by the KLT flow estimation module. This leads to difficulties with crossing traffic or for close, oncoming objects. Motion segmentation as proposed in this work can help for example to pre-initialize optical flow values.
For the full three-dimensional motion estimation problem referred to as *scene flow* [Vedula et al., 1999] computation, in total four images are needed: additionally, the stereo images at two consecutive time steps are taken into account. The additional estimated depth motion can help to constrain the optical flow estimation and to solve for ambiguities. Several approaches have been proposed in literature, see [Huguet and Devernay, 2007, Wedel et al., 2008, Vedula et al., 2005, Wedel et al., 2011, Pons et al., 2007, Zhang and Kambhamettu, 2001].
The performance of scene flow computation can be further increased with temporal filtering. Kalman filters form the basis of the so-called 6D Vision principle [Franke et al., 2005, Rabe et al., 2010]. In this approach, the observed disparity changes and the optical flow translation vectors of individual pixels are fused over time by means of a Kalman filter. Any optical flow scheme and any stereo estimation method can be used as input data for this approach. At the moment the 6D Vision approach is one of the most accurate, robust and powerful motion estimation approaches for stereo image sequences, see [Rabe et al., 2010] for a comparison with conventional scene flow estimation.

2.4 Static Stixel World

Instead of considering individual pixels, in the present work, a medium-level representation called Stixel World proposed in [Pfeiffer and Franke, 2011, Badino et al., 2009, Pfeiffer et al., 2012] is used. In the following, the Stixel computation is briefly discussed since the dynamic programming-based Stixel extraction is related to the Stixel segmentation introduced in Chapter 4 that is also based on dynamic programming. Insofar a

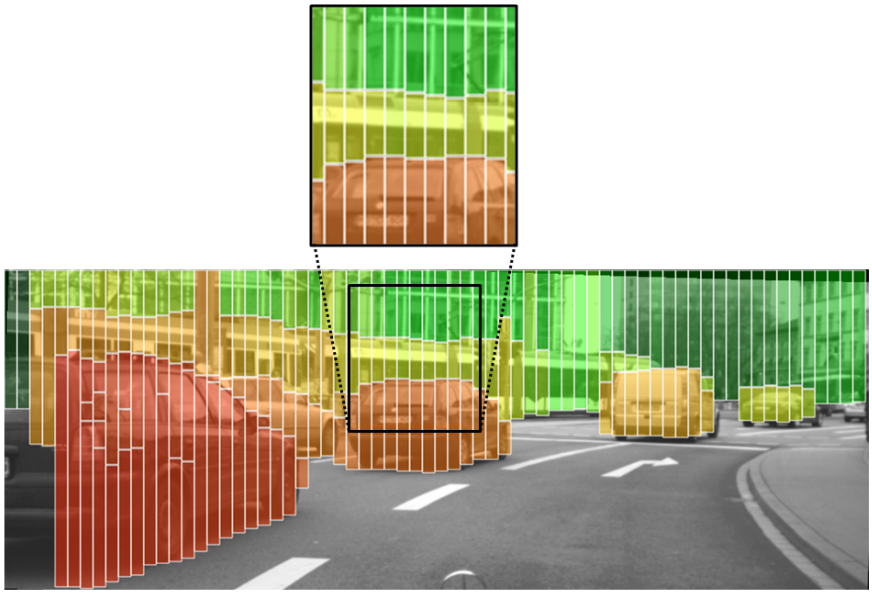


Figure 2.7: The Stixel World partitions an input image column-wise into several obstacle layers and street. The distance is color encoded ranging from red (close) to green (far away).

detailed understanding of the Stixel computation is important in the sense of related work for a comparison of both approaches. Furthermore, using the Stixel World brings great advantages but also might cause artifacts for a subsequent object segmentation. Thus, knowing the strengths and weaknesses of the Stixel World is important for this work.
The Stixel World yields the important freespace information and it provides an efficient object representation that is required by many driver assistance systems. See Figure 2.7 for an example scenario. Originally, the Stixel World was introduced in [Badino et al., 2009] as a medium level scene representation for traffic scenarios. The Stixel representation is characterized to be compact, robust to outliers and easy to access. The Stixel World partitions an input image I column-wise into several layers of one of the two classes $\mathcal{C}_{\text{Stixel}} \in \{\text{street, obstacle}\}$. It exploits the fact that typically man-made environments are dominated by either horizontal or vertical planar surfaces. Accordingly, a Stixel is defined as a thin rectangle with a fixed pixel width and vertical pose that approximates the underlying disparity image. This way, the relevant depth information of the respective traffic scene is represented with a few hundred Stixels instead of hundreds of thousands of individual stereo depth measurements. This compression of the input data volume also reduces the computational burden for subsequent driver assistance applications, in some cases by several orders of magnitude, see e.g. [Benenson et al., 2012b, Benenson et al., 2012a, Enzweiler et al., 2012, Benenson et al., 2011, Erbs et al., 2012b]. Besides that, the Stixel World is insensitive to stereo outliers which also

increases robustness of subsequent algorithms.
More formally, let $\mathcal{D}$ denote a disparity image, which is assumed to be of size $W \times H \in \mathbb{N}^2$. Temporal aspects are not taken into account in the Stixel computation. As indicated above, for each column the multi-layered Stixel World corresponds to a labeling $\mathbf{L}$ into the classes $\mathcal{C}_{\text{Stixel}} \in \{\text{street, obstacle}\}$ from the set $\mathbb{L}$ of all possible labelings. The resulting labeling is in the form of [Pfeiffer and Franke, 2011]

$$\begin{aligned}\mathbf{L} &= \{\mathbf{L}_u\}, \text{with } 0 \leq u < W \\ \mathbf{L}_u &= \{\mathbf{s}_n\}, \text{with } 1 \leq n \leq N_u \leq H \\ \mathbf{s}_n &= \{v_n^b, v_n^t, c_n, f_n(v)\}, \text{with } 0 \leq v_n^b \leq v_n^t < H,\ c_n \in \mathcal{C}_{\text{Stixel}}.\end{aligned} \tag{2.9}$$

These equations imply that a labeling $\mathbf{L}$ for the whole image I is composed of W column-wise labelings $\mathbf{L}_u$. These column labelings are assumed to be independent of each other for reasons of efficiency. Again, each column labeling consists of N_u segments $\mathbf{s}_n$. A segment represents a connected set of pixels with the same class $\mathcal{C}_{\text{Stixel}}$.
In the following, a Stixel is synonymous with an obstacle segment $\mathbf{s}_n$, the street segments are ignored. The image row coordinates v_n^b and v_n^t denote the position of the Stixel base point and of the top point, respectively. Finally, $f_n(v)$ is an arbitrary function that computes the expected model disparity of that segment at row v, where $v_n^b \leq v \leq v_n^t$. The segments $\mathbf{s}_{n-1}$ and $\mathbf{s}_n$ are adjacent such that each pixel is assigned to exactly one segment. For each labeling $\mathbf{L}_u \in \mathbf{L} \in \mathbb{L}$ of column u the following ordering applies

$$0 = v_1^b \leq v_1^t \leq ... < v_{N_u}^b \leq v_{N_u}^t = H - 1, \text{ with } v_{n-1}^t + 1 = v_n^b,\ 2 < n \leq N_u. \tag{2.10}$$

The final Stixel labeling is equivalent to the most probable labeling $\mathbf{L}^*$ defined as

$$\mathbf{L}^* = \arg\max_{\mathbf{L} \in \mathbb{L}} p\left(\mathbf{L} \mid \mathcal{D}\right). \tag{2.11}$$

The posterior probability is decomposed into

$$p\left(\mathbf{L} \mid \mathcal{D}\right) \propto p\left(\mathcal{D} \mid \mathbf{L}\right) \cdot p\left(\mathbf{L}\right) \approx \prod_{u=0}^{w-1} p\left(\mathbf{D}_u \mid \mathbf{L}_u\right) \cdot p\left(\mathbf{L}_u\right). \tag{2.12}$$

In this factorization, neighboring columns are considered as independent of each other. $\mathbf{D}_u \in \mathcal{D}$ denotes the vertical disparity measurement vector of column u. The individual disparity measurements $d_v \in \mathbf{D}_u$ are considered as independent, too. The data term, $p\left(\mathbf{D}_u \mid \mathbf{L}_u\right)$ penalizes the deviation of the disparity measurements from the expected disparity values. Objects are assumed to have a constant depth, so $f_n(v) = \mu$ is constant for each segment. The road is assumed to be flat (height is zero), so $f_n(v) = \alpha \cdot (v_{\text{hor}} - v)$, where α is the expected ground disparity gradient and v_{hor} is the row coordinate of the horizon. See Figure 2.8 for a visualization of the underlying disparity models for both classes $\mathcal{C}_{\text{Stixel}}$.
The data term $p\left(\mathbf{D}_u \mid \mathbf{L}_u\right)$ is modeled as a mixture model consisting of a uniform distribution to model outliers and a Gaussian distribution that quadratically penalizes deviations from the expected disparity model $f_n(v)$. Due to the independence assumption of the measurements,

$$p\left(\mathbf{D}_u \mid \mathbf{L}_u\right) = \prod_{n=1}^{N_u} \prod_{v=v_n^b}^{v_n^t} p\left(d_v \mid \mathbf{s}_n, v\right) \tag{2.13}$$

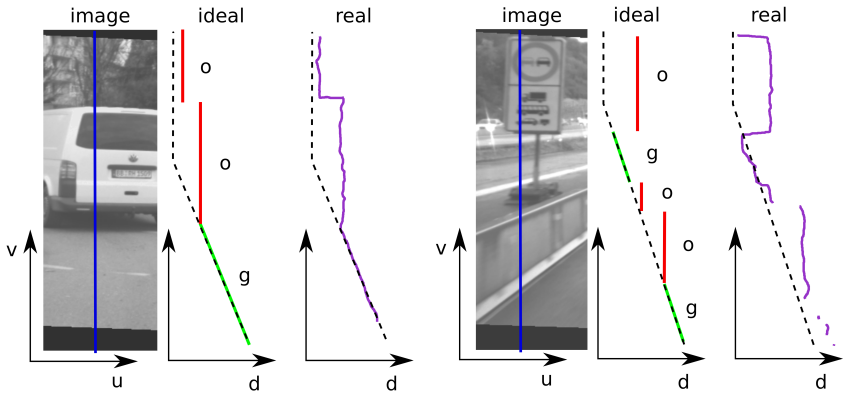


Figure 2.8: Visualization of the Stixel data term model taken from [Pfeiffer et al., 2012]. The figure illustrates the underlying disparity models for obstacles (o) and street (g).

holds. Additionally, there are invalid disparity measurements due to occlusions. Let p_{out} denote the outlier probability and $p_{\nexists}^{c_n}$ the fixed probability that an invalid stereo measurement is observed for a given class $c_n \in \mathcal{C}_{\text{Stixel}}$. With that background, the single pixel disparity likelihood is defined as

$$p\left(d_v \mid \mathbf{s}_n, v\right) = \begin{cases} p_{\exists}^{c_n}\left(d_v \mid \mathbf{s}_n, v\right) \cdot \left(1 - p_{\nexists}^{c_n}\right) & \text{, if } \exists(v) = 1, \\ p_{\nexists}^{c_n} & \text{, otherwise} \end{cases} \tag{2.14}$$

and

$$p_{\exists}^{c_n}\left(d_v \mid \mathbf{s}_n, v\right) = \frac{p_{\text{out}}}{d_{\max} - d_{\min}} + A_{\text{norm}} \cdot \exp^{-\frac{1}{2}\left(\frac{d_v - f_n(v)}{\sigma^{c_n}(f_n, v)}\right)^2}. \tag{2.15}$$

In this equation, the standard deviation $\sigma^{c_n}\left(f_n, v\right)$ incorporates a class-specific noise model for the disparity measurements. $\exists(v) = 1$ is an existential quantifier for the disparity measurement at pixel (u, v) and A_{norm} is a normalization constant given by

$$A_{\text{norm}} = \frac{1 - p_{\text{out}}}{A_{\text{range}}} \frac{1}{\sigma^{c_n}\left(f_n, v\right) \cdot \sqrt{2\pi}}, \tag{2.16}$$

and A_{range} is the Gaussian normalization constant for the finite interval of possible disparities $d_{\min} \leq d \leq d_{\max}$

$$A_{\text{range}} = 0.5 \cdot \left[\text{erf}\left(\frac{d_{\max} - f_n(v)}{\sqrt{2}\sigma^{c_n}\left(f_n, v\right)}\right) - \text{erf}\left(\frac{d_{\min} - f_n(v)}{\sqrt{2}\sigma^{c_n}\left(f_n, v\right)}\right)\right], \tag{2.17}$$

where $\text{erf}\left(.\right)$ denotes the Gauss error function. In the used setup, $d_{\min} = 0$ pixels and $d_{\max} = 128$ pixels holds.
The prior term $p\left(\mathbf{L}_u\right)$ is modeled as a first order Markov chain

$$p\left(\mathbf{L}_u\right) = p\left(\mathbf{s}_1, ..., \mathbf{s}_{N_u}\right) \approx p\left(\mathbf{s}_1\right) \cdot \prod_{n=2}^{N_u} p\left(\mathbf{s}_n \mid \mathbf{s}_{n-1}\right) \tag{2.18}$$

and incorporates semantic aspects between adjacent segments including

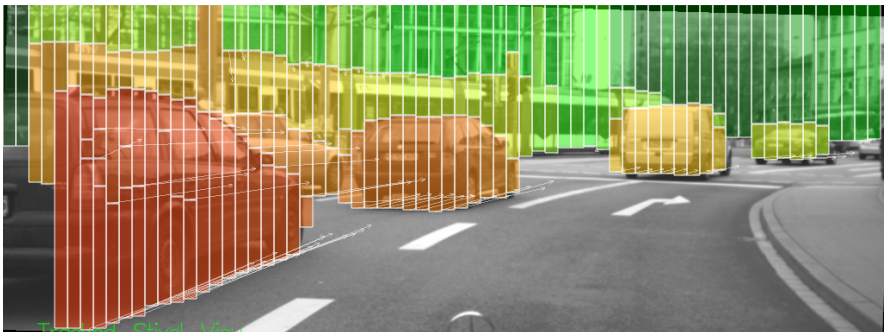


Figure 2.9: The Dynamic Stixel World continues the 6D-Vision principle for superpixel tracking. The arrows show the predicted Stixel position for the next half second.

- Bayesian Information Criterion (BIC) [Bishop, 2007, Gallup et al., 2010]: The number of objects captured along every column is small and dispensable cuts should be avoided.
- Gravity Constraint: Flying objects are unlikely. Object segments that are adjacent to street segments should stand on the ground surface.
- Ordering constraint [Gehrig and Franke, 2007]: When traversing an image from the bottom to the top along a column, the distance of 3D points tends to increase.

Recently, the Stixel phantom rate (Stixels that are erroneously detected as obstacles but they are street or sky in reality) was shown to be reducible significantly taking into account stereo confidences [Pfeiffer et al., 2013]. In this case, the outlier probability p_{out} is set individually for each pixel based on a learned mapping from stereo confidences to the outlier probability p_{out}.

2.5 Dynamic Stixel World

Motion information underlying the temporal development of the current vehicle environment plays a key role for collision prevention and for driver assistance in general.
In the present work, motion information of other objects is inferred from a tracking of the Stixels over time. This step follows the 6D-Vision principle [Franke et al., 2005] that was already mentioned at the end of Section 2.3. In the Stixel tracking, the image positions as well as the disparities of the Stixels are measured at each time step and are fused by means of a Kalman filter [Pfeiffer et al., 2012, Pfeiffer and Franke, 2010].
The required ego motion information can be measured by the inertial sensors of the experimental vehicle or can be determined by an image-based estimation, see e.g. [Badino, 2007, Milella and Siegwart, 2006, Kitt et al., 2010].
In contrast to 6D-Vision, for the Stixel tracking only a four dimensional state vector is estimated. The height-related components (height $Y = 0$ and height velocity $\dot{Y} = 0$) are omitted, since moving objects such as cars or bicycles are assumed to move on the

planar ground plane.
To sum up, the state vector $\vec{X}^{\,t} := \left(X^t, Y^t = 0, Z^t, \dot{X}^t, \dot{Y}^t = 0, \dot{Z}^t\right)^T$ is estimated. Throughout this work, a left-handed coordinate system is used. In this coordinate system, the Z-axis points along the driving direction, the Y-axis points upwards and the X-axis to the right. Within the time interval Δt, the ego vehicle coordinate system with ego velocity $\vec{v}_{ego}$ given by

$$\vec{v}_{ego} = V_{ego} \cdot \begin{pmatrix} \sin\left(\dot{\psi} t\right) \\ 0 \\ \cos\left(\dot{\psi} t\right) \end{pmatrix} \tag{2.19}$$

moves

$$\Delta\vec{x}_{ego} = \int_0^{\Delta t} \vec{v}_{ego}\,\mathrm{d}t = \frac{V_{ego}}{\dot{\psi}} \cdot \begin{pmatrix} 1 - \cos\left(\dot{\psi}\Delta t\right) \\ 0 \\ \sin\left(\dot{\psi}\Delta t\right) \end{pmatrix}, \tag{2.20}$$

assuming a constant ego velocity V_{ego} and a constant yaw rate $\dot{\psi}$ within the time interval Δt. A Stixel located a

$$\vec{x}^{\,t-1} = \left(X^{t-1}, 0, Z^{t-1}\right)^T \tag{2.21}$$

with velocity

$$\vec{V}^{t-1} = \left(\dot{X}^{t-1}, 0, \dot{Z}^{t-1}\right)^T \tag{2.22}$$

moves within this time interval Δt to

$$\vec{x}^{\,t} = \mathbf{R}_y\left(\Delta\psi\right)\left(\vec{x}^{\,t-1} + \vec{V}^{t-1}\Delta t - \Delta\vec{x}_{ego}\right), \tag{2.23}$$

where $\mathbf{R}_y$ corresponds to the 3×3 rotational matrix revolving around the y-axis

$$\mathbf{R}_y\left(\Delta\psi\right) = \begin{pmatrix} \cos\Delta\psi & 0 & \sin\Delta\psi \\ 0 & 1 & 0 \\ -\sin\Delta\psi & 0 & \cos\Delta\psi \end{pmatrix}. \tag{2.24}$$

describing the orientation of the new rotated ego coordinate system. Positive angles in this left-handed coordinate system correspond to clockwise rotations. Accordingly, the resulting system model of the extended Kalman filter is defined. The predicted Stixel state

$$\vec{X}^{\,t} = \left(\vec{x}^{\,t}, \vec{V}^{\,t}\right)^T = \left(X^t, 0, Z^t, \dot{X}^t, 0, \dot{Z}^{\,t}\right)^T \tag{2.25}$$

evolves from the current state $\vec{X}^{t-1}$ via the state transition matrix $\mathbf{A}^t$ and is influenced by the control input vector $\vec{B}^{\,t}$ that contains speed and yaw rate. The process noise $\vec{\omega}^{\,t}$ is assumed to be Gaussian white noise with covariance matrix $\mathbf{Q}^t$. This way, Equation 2.23 becomes

$$\vec{X}^{\,t} = \mathbf{A}^t \cdot \vec{X}^{t-1} + \vec{B}^{\,t} \cdot V_{ego} + \vec{\omega}^{\,t}. \tag{2.26}$$

More precisely, $\mathbf{A}^t$ is given by

$$\mathbf{A}^t = \begin{pmatrix} \mathbf{R}_y(\Delta\psi) & \Delta t \mathbf{R}_y(\Delta\psi) \\ \mathbf{0}_{3\times3}(\Delta\psi) & \mathbf{R}_y(\Delta\psi) \end{pmatrix} \tag{2.27}$$

and

$$\vec{B}^t = \frac{1}{\dot{\psi}} \begin{pmatrix} \cos\left(\dot{\psi}\Delta t\right) - 1 \\ 0 \\ -\sin\left(\dot{\psi}\Delta t\right) \\ \mathbf{0}_{3\times1} \end{pmatrix}. \tag{2.28}$$

Since the Stixel tracking algorithm works on rectified images, it is possible to use the pinhole camera model [Hartley and Zisserman, 2000]. The non-linear measurement equation is given by

$$\vec{z}^t = \begin{pmatrix} u \\ v \\ d \end{pmatrix} = \frac{1}{Z} \begin{pmatrix} X f_x \\ Y f_y \\ b f_x \end{pmatrix} + \vec{\gamma}^t = \mathbf{H}^t \cdot \vec{X}^t + \vec{\gamma}^t, \tag{2.29}$$

where f_x and f_y denote the scaled focal lengths in pixels, b is the baseline of the stereo camera system and the measurement matrix $\mathbf{H}^t$ is given by

$$\mathbf{H}^t = \frac{1}{Z} \begin{pmatrix} f_x & 0 & 0 & 0 & 0 & 0 \\ 0 & f_y & 0 & 0 & 0 & 0 \\ 0 & 0 & b f_x / Z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{2.30}$$

$\vec{\gamma}^t$ is assumed to be Gaussian white noise with covariance matrix $\mathbf{R}^t$. Since this projection is a non-linear function, the resulting observation matrix $\mathbf{H}^t$ – which maps the state space to measurement space – is the Jacobian approximation of Equation 2.29. An extended Kalman filter is used in this case.
In order to find the correspondences between Stixels of different time steps, a GPU-based KLT flow implementation similar to [Sinha et al., 2006] is used. Inside a Stixel, all optical flow measurements are averaged by taking the median of the involved flow measurements. This step helps to obtain more reliable flow measurements for the tracking. Averaging the optical flow values requires the optical flow inside the whole Stixel to be constant. Note that this assumption does not hold in general. For example, for object motion in longitudinal direction, the assumption of a constant optical flow is not valid since the lengths of optical flow vectors slightly increase from the focus of expansion to the image boundaries. This effect is neglected in the Stixel tracking. Although it would be more correct for example to estimate an affine flow profile for each Stixel, typically there a just a few flow measurements inside each Stixel. So it makes sense to just take the median flow value in order to gain robustness.
In order to preserve the original Stixel grid structure, the tracking scheme presented in [Pfeiffer and Franke, 2010] had to be adopted. See [Scharwaechter, 2012] for details.

2.6 Inference Algorithms for Stixel Segmentation

In the present work, the object segmentation of the Stixels into objects is formulated as a Bayesian energy minimization problem. Each Stixel is assigned to exactly one object class such as static background or different moving objects. The aim is to find the optimal Stixel assignment, that is an assignment with minimal costs with regard to an adequate metric. Such optimization problems arise in all areas of science, engineering, business and industry [Heath, 1998].
Due to the comparatively large number of Stixels, the resulting objective function is high-dimensional with hundreds of interrelated variables. Efficiently minimizing such functions is non-trivial and NP-hard in many cases [Boykov et al., 2001, Blake and Zisserman, 2012].
In the present work, two different optimization techniques are used. One approach is based on so-called Markov Random Fields [Moussouris, 1974, Li, 2009]. Markov Random Fields (MRFs) provide a powerful framework to describe the probability of a set of interrelated random variables. A MRF expresses the energy [1] of a labeling, that is a particular assignment of values to the random variables corresponding to the class of all Stixels, as a sum of potentials, where each summand only depends on a subset (more precisely, on a maximal clique, see Figure 2.13) of the random variables [Kohli and Kumar, 2010].
For MRFs, a vast amount of efficient optimization strategies have been proposed. See [Li, 2009] for an overview and [Szeliski et al., 2008, Tappen and Freeman, 2003] for comparative studies. In this work, a multi-class graphcut optimization scheme [Boykov et al., 2001] is chosen. This algorithm is very efficient and satisfies certain optimality properties, cf. Subsection 2.6.2. Furthermore, an approach based on dynamic programming [Bellman, 1954] is presented. Dynamic programming allows to find the true optimum of an energy function if certain conditions are met, see 2.6.1. Both concepts are introduced in the following.

2.6.1 Dynamic Programming

Inference on a linear chain

The concept of dynamic programming has been introduced by [Bellman, 1954]. Generally spoken, dynamic programming is a method for solving complex problems by breaking them down into simpler subproblems. In order to be applicable, the problem under consideration must exhibit *optimal substructure* and *overlapping subproblems* [Cormen et al., 2001].
Let $N \in \mathbb{N}$ denote the number of Stixels or pixels in an image I. Furthermore, let $\mathcal{J} = \{1, ..., J\}$ denote the set of semantic classes. For each Stixel, a random variable l_i is introduced, $i = 1...N$. In this section, the time index t is omitted as a superscript for better readability and since there is no risk of confusion. Furthermore, any further possible conditional dependencies of any of the introduced probability distributions are ignored for the same reasons.
A labeling $\mathbf{L} \in \mathbb{L}$ from the set $\mathbb{L}$ of all possible labelings is defined to be a realization

[1] The energy is defined as the negative logarithm of the probability [Gray, 1990].

of these random variables, i.e. each random variable takes the value from exactly one of the classes in $\mathcal{J}$, for example $\mathbf{L} = \{l_1 = 1, l_2 = 2, ..., l_N = 3\}$.
Suppose there is a probability density function $p(\mathbf{L})$ for such a labeling $\mathbf{L}$

$$p(\mathbf{L}) = p(l_1, ..., l_N) = p(l_1) \cdot \prod_{i=2}^{N} p(l_i \mid l_1, ..., l_{i-1}) \tag{2.31}$$

and assume that the conditional probability distribution on the right-hand side is independent of all previous Stixel labelings except the most recent. In this case, a first order Markov chain [Bishop, 2007] is obtained

$$p(\mathbf{L}) = p(l_1, ..., l_N) = p(l_1) \cdot \prod_{i=2}^{N} p(l_i \mid l_{i-1}). \tag{2.32}$$

The maximization of Equation 2.32 is equivalent to the minimization of the corresponding negative log-likelihood term, called energy E [Gray, 1990]

$$E(\mathbf{L}) := -\log p(\mathbf{L}) = E(l_1) + \sum_{i=2}^{N} E(l_i \mid l_{i-1}). \tag{2.33}$$

A priori, minimizing N variables each with J states requires to evaluate J^N realizations for $\mathbf{L}$. However, the factorization properties of Equation 2.33 suggest a far more efficient algorithm. Consider for example the minimization of l_1. Since $E(l_1) + E(l_2 \mid l_1)$ are the only terms that depend on l_1, it is possible to perform a one-dimensional minimization over l_1 to obtain a function of l_2. This term will be referred to as $\hat{E}_1(l_2)$. $\hat{E}_1(l_2)$ is evaluated along with $E(l_3 \mid l_2)$ to define $\hat{E}_2(l_3)$ and so on. This means that the minimum is computed recursively

$$\min_{l_1, l_2, ... l_N} \left[E(l_1) + \sum_{i=2}^{N} E(l_i \mid l_{i-1}) \right] = \tag{2.34}$$

$$\min_{l_N} \left[\underbrace{\min_{l_{N-1}} \left[\underbrace{... \min_{l_2} \left[\underbrace{\min_{l_1} \left[E(l_1) + E(l_2 \mid l_1) \right]}_{\hat{E}_1(l_2)} + E(l_3 \mid l_2) \right] + ...}_{\hat{E}_2(l_3)} \right]}_{\hat{E}_{N-1}(l_N)} \right]. \tag{2.35}$$

The idea of dynamic programming is to generate a sequence of functions of one variable by intelligent insertion of brackets

$$\begin{aligned} \hat{E}_1(l_2) &= \min_{l_1} E(l_2 \mid l_1) + E(l_1) \\ \hat{E}_2(l_3) &= \min_{l_2} \left[\hat{E}_1(l_2) + E(l_3 \mid l_2) \right] \\ &\vdots \end{aligned} \tag{2.36}$$

which can be rewritten in recursive form as

$$\hat{E}_i\left(l_{i+1}\right) = \min_{l_i}\left[\hat{E}_{i-1}\left(l_i\right) + E\left(l_{i+1} \mid l_i\right)\right]. \tag{2.37}$$

$\hat{E}_i\left(l_{i+1}\right)$ can also be interpreted as the *belief* of Stixel i for Stixel $i+1$ taking class l_{i+1} or the *message* passed forward from Stixel i to Stixel $i+1$, see [Bishop, 2007]. The optimal solution is obtained from

$$\arg\min_{\mathbf{L}} E\left(\mathbf{L}\right) = \arg\min_{l_N} \hat{E}_{N-1}\left(l_N\right) \tag{2.38}$$

via backtracking. For this purpose, the optimal class choice l_i^* in Equation 2.37 for $\hat{E}_i\left(l_{i+1}\right)$ is stored and is returned by $l_i^* = \arg\min_{l_i} \hat{E}_i\left(l_{i+1}\right)$.
At the end of the minimization, that is after the computation of
$\hat{E}_1\left(l_2\right), \hat{E}_2\left(l_3\right), ..., \hat{E}_{N-1}\left(l_N\right)$

$$l_N^* = \arg\min_{l_N} \hat{E}_{N-1}\left(l_N\right) \tag{2.39}$$

and

$$\begin{aligned} l_{N-1}^* &= \arg\min_{l_{N-1}} \hat{E}_{N-1}\left(l_N^*\right) \\ l_{N-2}^* &= \arg\min_{l_{N-2}} \hat{E}_{N-2}\left(l_{N-1}^*\right) \\ &\vdots \\ l_1^* &= \arg\min_{l_1} \hat{E}_1\left(l_2^*\right) \end{aligned} \tag{2.40}$$

can be obtained. This algorithm is known as Viterbi algorithm [Viterbi, 1967].
So whenever a probability distribution can be decomposed as done in Equation 2.32, dynamic programming yields the optimal solution irrespective of the properties of the underlying energy function. Of course, dynamic programming can be used for other factorizations as well, even for the general case in Equation 2.31. However, complexity of the optimization rises exponentially in the dimension ν of $\hat{E}\left(l_i \mid l_{i-1}, ..., l_{i-\nu}\right)$.
Summing up, dynamic programming yields the true global minimizer for the general energy function

$$\begin{aligned} E\left(l_1, l_2, l_3\right) &= E\left(l_1\right) + E\left(l_2 \mid l_1\right) + E\left(l_3 \mid l_2, l_1\right) \\ &\approx E\left(l_1\right) + E\left(l_2 \mid l_1\right) + E\left(l_3 \mid l_2\right) \end{aligned} \tag{2.41}$$

whenever the equality holds for the general inequality

$$\min_{l_1,l_2,l_3}\left[E\left(l_1\right) + E\left(l_2 \mid l_1\right) + E\left(l_3 \mid l_2\right)\right] \leq$$

$$\min_{l_3}\left\{\underbrace{\min_{l_2}\left[\underbrace{\min_{l_1}\left(E\left(l_1\right) + E\left(l_2 \mid l_1\right)\right)}_{\hat{E}_1(l_2)} + E\left(l_3 \mid l_2\right)\right]}_{\hat{E}_2(l_3)}\right\}. \tag{2.42}$$

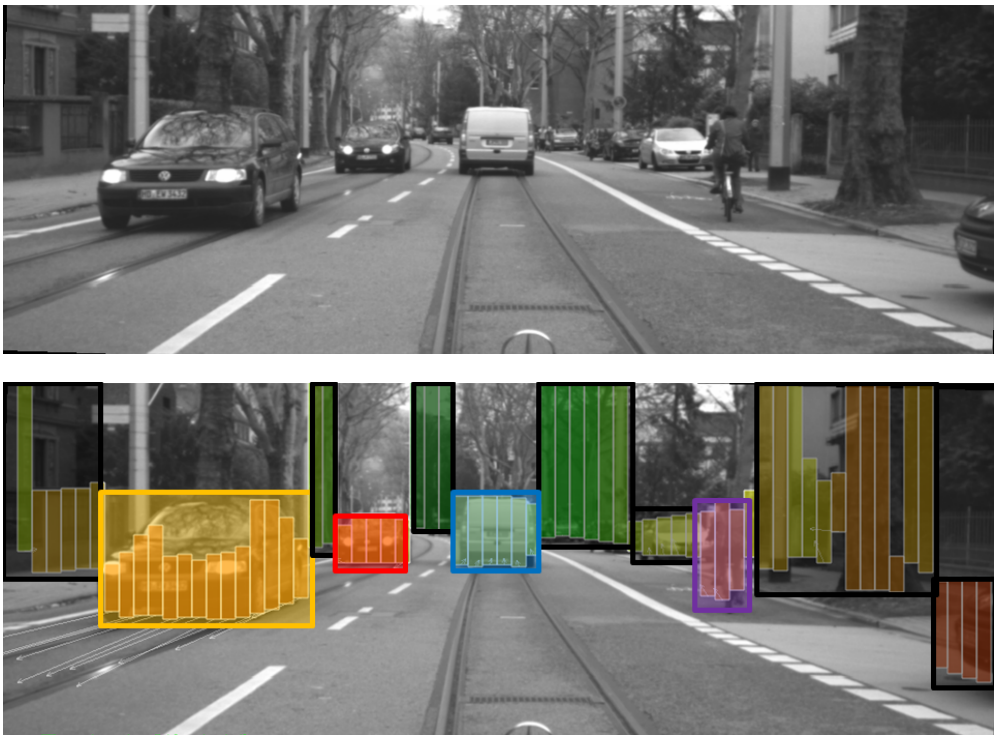


Figure 2.10: Visualization of the segment concept. All dynamic Stixels are grouped to segments indicated by boxes which correspond to the objects in the scene.

This means that $E(l_3 \mid l_2)$ must indeed be independent of l_1 as stated. In this case, the problem has optimal substructure. This conclusion can be generalized to situations with more variables and higher dimension ν. If there is no such conditional independence, the optimization is NP-hard.
An efficient generalization of this concept is to consider segments [Sarawagi and Cohen, 2004] of Stixels. In this case, a labeling $\mathbf{L}$ is defined to be a composition of $\mathcal{F}$ intervals, segments or objects $\mathbf{L} = \{\mathbf{s}_n\}, n = 1...\mathcal{F}$ with a left start position u_n^l, a right end position u_n^r and a class c_n,

$$\mathbf{s}_n := \left(u_n^l, u_n^r, c_n\right). \tag{2.43}$$

The concept of a segment is a collective term for all Stixels or pixels inside this region, see Figure 2.10. This definition is in analogy with Equation 2.9.
Using this segment concept, the Markov-chain factorization 2.32 can be rewritten as

$$p(\mathbf{L}) = p(\mathbf{s}_1, ..., \mathbf{s}_\mathcal{F}) = p(\mathbf{s}_1) \cdot \prod_{n=2}^{\mathcal{F}} p(\mathbf{s}_n \mid \mathbf{s}_{n-1}), \tag{2.44}$$

analogous to Equation 2.18. In case of considering segments, a similar recursive relation to 2.37 can be deduced, whereby it is explicitly optimized over the lengths of the

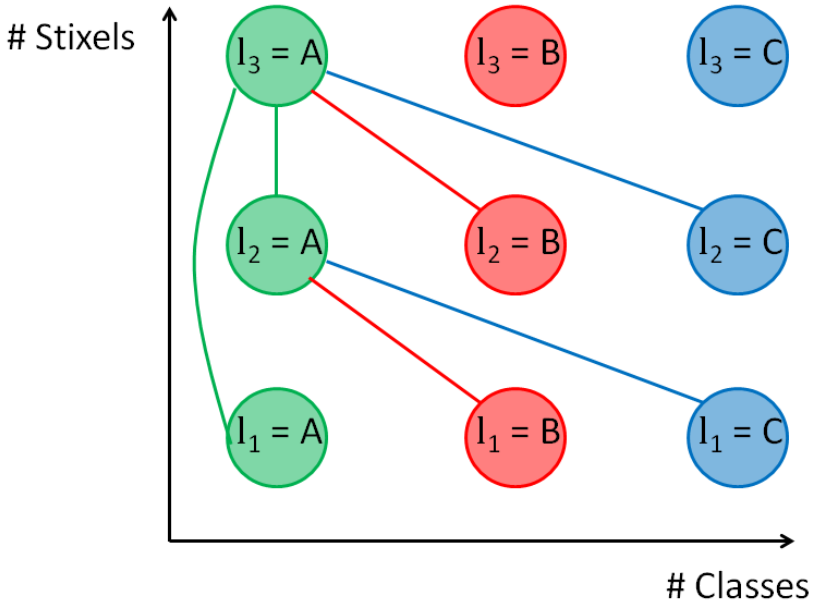


Figure 2.11: Visualization of the segment-based dynamic programming. A toy example for 3 classes and 3 Stixels is shown, the Markov chain order is $\nu = 2$. The figure shows all the transitions that end up with Stixel number 3 taking class A which are relevant for $\hat{E}(l_3 = A)$.

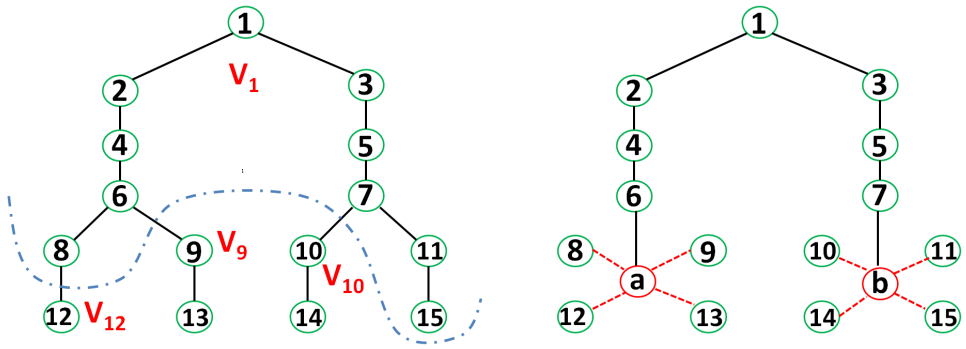
segments up to length $\nu + 1$

$$\hat{E}(l_i) = \min_{j,l_j} \left[\hat{E}(l_j) + E^i_{j+1}(l_i \mid l_j) \right], \text{ with}$$
$$\{j \in \mathbb{N} \mid (0 \leq j < i) \wedge (i - \nu \leq j)\}, \{l_j \in \mathcal{J}\}, \tag{2.45}$$

where $\hat{E}(l_0)$ is initialized with zero. In this equation, $E^i_{j+1}(l_i \mid l_j)$ quantifies the probability for a segment including all Stixels from Stixel $j+1$ to Stixel i taking the same class l_i under the condition that the Stixels from the previous segment have decided for class l_j,

$$E^i_{j+1}(l_i \mid l_j) := E(l_{j+1} = l_i, l_{j+2} = l_i, \ldots \mid l_j). \tag{2.46}$$

If it is not possible to constrain the value domain of ν based on prior knowledge, this parameter has to be set to the total number of Stixels N. See Figure 2.11 for a visualization of this concept.
For the case that the cost table $E^i_{j+1}(l_i \mid l_j)$ can be precomputed and can be obtained in $\mathcal{O}(1)$, the complexity of the minimization is $\mathcal{O}(\nu \cdot N \cdot J^2)$. In the present work, $\nu = N$ was unbounded, resulting in a complexity of $\mathcal{O}(N^2 \cdot J^2)$.



(a) On the minimum cut shown in blue for a tree-structured graphical model rooted at node number 1. The resulting tree partitioning into the sets of nodes V_1, V_9, V_{10} and V_{12} is shown in red.

(b) On the proposed pruning step that aggregates the nodes 8,9,12,13 to supernode a and the nodes 10,11,14,15 to supernode b to reduce the tree depth.

Figure 2.12: Visualization of the dynamic programming step on a tree.

Note that under certain circumstances it is possible to reduce complexity of the minimization to $\mathcal{O}\left(N^2 \cdot J\right)$ with a different optimization technique based on the generalized distance transform [Felzenszwalb and Huttenlocher, 2012]. However, the class of possible transition probabilities $E^i_{j+1}\left(l_i \mid l_j\right)$ has to be restricted in this case. For that reason, the more expensive but more general Dynamic Programming-based minimization was chosen in the present work. Besides that, the profit that can be generated for $\nu = 6$ in this work is small.
The segment model combines simultaneously a great modeling freedom with an efficient optimization. The key advantage of this model in comparison with a traditional Conditional Random Field [Lafferty et al., 2001] or with a Hidden Markov Model [Rabiner, 1989] is the ability to take into account higher-order object properties such as object sizes or shape information that require knowledge on all involved Stixels simultaneously. This idea will again be taken up in Chapter 4 where object segmentation will be formulated using this dynamic programming-based segment model.

Inference on a tree

The concept of dynamic programming on linear array structures can be generalized to tree structures. It is well known that inference for trees and other graphs with low treewidth can be performed exactly with dynamic programming, however complexity rises exponentially in the treewidth [Bertele and Brioschi, 1972, Garey and Johnson, 1979]. Consider the (sub)tree shown in Figure 2.12(a). Node 1 is the root vertex of the tree. Each node $i = 1...N$, except the root, has a parent $\Pi(i)$, e.g. $\Pi(6) = 4$. The tree analogue of Equation 2.37 is given by [Veksler, 2005]

$$\hat{E}_i\left(l_{\Pi(i)}\right) = \min_{l_i}\left[E\left(l_{\Pi(i)} \mid l_i\right) + \sum_{j \in \mathbf{C}_i} \hat{E}_j\left(l_i\right)\right], \tag{2.47}$$

where $\mathbf{C}_i$ is the set of children of node i. For example, for node 4 this means

$$\hat{E}_6\left(l_4\right) = \min_{l_6}\left[E\left(l_6 \mid l_4\right) + \hat{E}_8\left(l_6\right) + \hat{E}_9\left(l_6\right)\right]. \tag{2.48}$$

The generalization of the higher-order model given by Equation 2.45 is slightly more complex [Erbs et al., 2014]. The optimal cut $C_i^* \in \mathcal{C}_i$ from the set of all possible cuts $\mathcal{C}_i$ for a subtree rooted at node i, T_i, is a partition $C_i^* = \{\mathbf{V}_j\}$ of T_i, i.e.

$$\mathbf{V}_{i_1} \bigcup \ldots \bigcup \mathbf{V}_{i_\mathcal{F}} = T_i \tag{2.49}$$

$$\mathbf{V}_j \bigcap \mathbf{V}_k = \emptyset \;\; \forall j \neq k, \tag{2.50}$$

where each $\mathbf{V}_j$ is a connected set of nodes all taking class l_j such that

$$\hat{E}\left(l_i\right) = \min_{\mathcal{C}_i, l_j}\left[E\left(\mathbf{V}_i, l_i \mid \{\mathbf{V}_j, l_j\}\right) + \sum_{j \in \mathbf{C}_{\mathbf{V}_i}} \hat{E}\left(l_j\right)\right]. \tag{2.51}$$

is minimized. Analogously, $\mathbf{C}_{\mathbf{V}_i}$ denotes the set of children nodes of the set $\mathbf{V}_i$.
See Figure 2.12(a) for a visualization. A possible optimal cut for the subtree T_1 rooted at node 1 is shown in blue where $\mathbf{V}_1 = \{1, 2, 3, 4, 5, 6, 7, 8, 11, 15\}$, $\mathbf{V}_{12} = \{12\}$, $\mathbf{V}_9 = \{9, 13\}$ and $\mathbf{V}_{10} = \{10, 14\}$. The children nodes $C_{\mathbf{V}_1} = \{9, 10, 12\}$ contribute the energies $\hat{E}\left(l_{12}\right)$, $\hat{E}\left(l_9\right)$ and $\hat{E}\left(l_{10}\right)$ to Equation 2.51 and $E\left(\mathbf{V}_1, l_1 \mid \mathbf{V}_9, l_9, \mathbf{V}_{10}, l_{10}, \mathbf{V}_{13}, l_{13}\right)$ describes the probability that all Stixels in $\mathbf{V}_1$ take the same class l_1, given the information that the set of Stixels specified by $\mathbf{V}_{12}$ took class l_{12}, the Stixels in $\mathbf{V}_9$ took class l_9 and the Stixels in $\mathbf{V}_{10}$ took class l_{10}.
In the end, the optimal assignment of the root node 1 is obtained via

$$\hat{E}^* = \min_{l_1 \in \mathcal{J}} \hat{E}\left(l_1\right) \tag{2.52}$$

and the optimal solution can be again extracted by backtracking.
However, typically the resulting overall tree is quite complex, see Figure 4.22(a). Under these conditions, efficient, real-time capable inference by means of dynamic programming is impossible. Note that when taking into account higher-order segments, complexity rises exponentially in the number of branches.
Nevertheless, in many instances, single objects can be respresented by simple subtrees, see Figure 4.21, so-called *spiders* which are subtrees with at most one branch vertex. A branch vertex is a vertex of degree greater than two [Gargano et al., 2002]. Accordingly, in the present work the maximum number of branch vertices was artificially limited to one by *pruning* [Pearl, 1984].
To see this, consider Figure 2.12(b). In order to infer the best cut for root node 1, C_1^*, the tree nodes $\{8, 9, 12, 13\}$ and $\{10, 11, 14, 15\}$ are replaced by single supernodes a and b. The best solutions so far for all classes for these subtrees are stored in these supernodes and these optimal partial solutions aren't changed in the following. Starting from these optimal partial solutions the optimal solution for the root is searched. So in principle, this method tries to attach further nodes to a partial optimal solution. The possible cuts inside the nodes $\{8, 9, 12, 13\}$ and $\{10, 11, 14, 15\}$ do not need to be tested in the optimization since they are absorbed in the respective supernodes.

This *spider-approximation* makes the optimization real-time capable and it allows to perform the optimization in a single pass with one global objective function, without any pre-segmentation steps. Nevertheless, for objects that can be represented by spiders, the proposed algorithm finds the global optimum. In practice, the object description via spiders subtrees is completely sufficient for most scenarios, see Figure 4.21 or 4.22 for examples. Currently, problems just arise for very close objects where the Stixel World tends to oversegment, especially for transparent panes. See the leading purple car shown in Figure 4.22(a) for an example. In this case, the corresponding object is represented by a more complex subtree and inference would become more expensive, since the number of branch vertices increases. Here the pruning step only yields a local optimum since relying on optimal partial solutions from the supernodes. It will be part of future work to avoid the Stixel oversegmentation for close objects.
Note that in the worst case the approach does the same as classical dynamic programming, see Equation 2.47 which also aggregates all the previous information into the most recent node. However, when considering typical tree structures as shown in Figure 4.22, it is apparent that there are many linear structures. It is possible to do better for these linear structures using the proposed algorithm. Why not use this information? See Section 4.6 for further discussion.

2.6.2 Graph Cut

The Stixel class assignment into stationary background or various moving objects can be cast as a Bayesian optimization problem to find the solution corresponding to the maximum a posteriori (MAP) probability. This way, a posterior probability $p(\mathbf{L}|\mathcal{Z})$ is specified that rates the probability to observe a certain Stixel class assignment $\mathbf{L} = \{l_1, l_2, ..., l_N\}$ from the set $\mathbb{L}$ of all possible labelings given the set of observations $\mathcal{Z}$. The observations $\mathcal{Z}$ are themselves regarded as instantiations of a random variable $\mathbb{Z}$ that represents the full space of all possible observations that can arise [Blake et al., 2011]. In this section, again the time index t is omitted as a superscript for better readability. There is no risk of confusion since this is the only time step under consideration in this section.
Applying Bayes' theorem

$$p(\mathbf{L} \mid \mathcal{Z}) = \frac{p(\mathcal{Z} \mid \mathbf{L}) \cdot p(\mathbf{L})}{p(\mathcal{Z})} \propto \underbrace{p(\mathcal{Z} \mid \mathbf{L})}_{\text{likelihood}} \cdot \underbrace{p(\mathbf{L})}_{\text{prior}} \tag{2.53}$$

allows to alter the MAP estimation problem to a regularized (due to the prior) maximum likelihood (due to the data likelihood) estimation problem. The so-called *evidence term* $p(\mathcal{Z})$ is dropped since it does not change the MAP solution which solely depends on $\mathbf{L}$. Of course, this is a restriction since all potential knowledge on the confidence in the MAP solution is lost. However, computing

$$p(\mathcal{Z}) = \sum_{\mathbf{L}} p(\mathcal{Z} \mid \mathbf{L}) \cdot p(\mathbf{L}) \tag{2.54}$$

in general is intractable. Typically, for $J = 6$ classes and $N = 300$ Stixels, the summation in Equation 2.54 involves $J^N = 6^{300} \approx 2.78 \cdot 10^{233}$ summands, an unimaginably big

number. For example, the number of atoms in the universe is much lower, it is about 10^{77} [Beutelspacher et al., 2006]. Note that some inference algorithms such as belief propagation [Yedidia et al., 2003] or under certain circumstances graphcut [Kohli and Torr, 2006] can yield such confidence measures.
Instead of maximizing Equation 2.53, equivalently one minimizes

$$E = -\log p\left(\mathbf{L} \mid \mathcal{Z}\right) \propto \underbrace{E\left(\mathcal{Z} \mid \mathbf{L}\right)}_{:=-\log p(\mathcal{Z}|\mathbf{L})} + \underbrace{E\left(\mathbf{L}\right)}_{:=-\log p(\mathbf{L})}, \tag{2.55}$$

since the logarithm is a monotonous functions, that is the maximum of $p\left(\mathbf{L} \mid \mathcal{Z}\right)$ is still the maximum of $\log p\left(\mathbf{L} \mid \mathcal{Z}\right)$ and instead of maximizing $\log p\left(\mathbf{L} \mid \mathcal{Z}\right)$ one minimizes $-\log p\left(\mathbf{L} \mid \mathcal{Z}\right)$. This step is preferable since maximizing Equation 2.53 is susceptible to arithmetic underflow.
The prior term allows to incorporate prior knowledge via a prior distribution over the quantity that has to be inferred. For example, typically there are strong correlations between Stixels that are nearby in an image, that is they often belong to the same physical object. A Markov Random field describes the dependencies between Stixels via an undirected graphical model of a set of random variables $(l_1, l_2, ..., l_N)$, called vertices. The vertices in this graph encode for example the class of the Stixels. Dependencies between Stixels are represented via edges in the graph, see Figure 2.13. In a MRF, the random variables obey a local Markov property, that is the Stixel class l_i is conditionally independent of all other Stixel classes given the class decision of its neighbors $\{l_{ne(i)}\}$

$$l_i \perp\!\!\!\perp l_{j \setminus ne(i)} \mid l_{ne(i)}, \quad \forall j = 1...N, j \neq i. \tag{2.56}$$

The Hammersley-Clifford theorem [Clifford, 1990] states that a probability density which meets the Markov property 2.56 factorizes over the maximal cliques $\mathcal{C}$ of the graph $\mathcal{G}$,

$$p\left(\mathbf{L} \mid \mathcal{Z}\right) \propto \prod_{\mathcal{C}} p\left(\mathbf{L}_{\mathcal{C}} \mid \mathcal{Z}\right). \tag{2.57}$$

A clique of the graph $\mathcal{G}$ is a subset of vertices, such that for every two vertices in the clique, there exists an edge connecting the two. A maximal clique is a clique that cannot be extended by including one more adjacent vertex without loosing the property of being a clique, see Figure 2.13.
Minimizing Equation 2.55 for $\mathbf{L}$ is NP-hard (non-deterministic polynomial-time hard) [Garey and Johnson, 1979] in general [Blake et al., 2011, Boykov et al., 2001, Blake and Zisserman, 2012], that is there is no algorithm that generally minimizes Equation 2.55 in polynomial time . However, there are certain families of functions for which Equation 2.55 can be minimized in polynomial time or even in real-time [Darbon, 2008,Schlesinger and Flach, 2006]. Pseudo-boolean submodular functions are one such family.
Submodular functions are the analogous boolean counterpart to continuous convex functions. A pseudo-boolean function $E : \{0,1\}^N \rightarrow \mathbb{R}$ is submodular if, and only if, for all label assignments $\mathbf{L_a}, \mathbf{L_b} \in \{0,1\}^N$, the function satisfies the condition [Blake et al., 2011]

$$E\left(\mathbf{L_a}\right) + E\left(\mathbf{L_b}\right) \geq E\left(\mathbf{L_a} \vee \mathbf{L_b}\right) + E\left(\mathbf{L_a} \wedge \mathbf{L_b}\right), \tag{2.58}$$

where $\vee$ and $\wedge$ are componentwise OR and AND, respectively. All pseudo-boolean functions of arity 1 ($N = 1$) are submodular. For pseudo-boolean functions of arity 2,

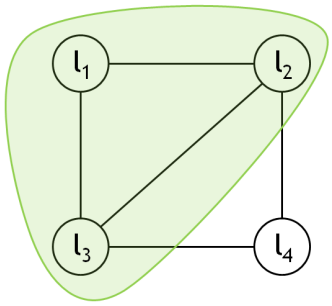


Figure 2.13: An undirected graph consisting of four Stixels. The black lines indicate that there is a link between these nodes. A maximal clique consisting of l_1, l_2 and l_3 is marked green. However, l_2, l_3 and l_4 is a maximal clique, too.

Equation 2.58 is equivalent to

$$E(1,0) + E(0,1) \geq E(1,1) + E(0,0). \tag{2.59}$$

Note that the sum of two or more submodular functions is another submodular function. This way, the factorization property 2.57 ensures that the overall energy is submodular as long as the different summands like 2.59 are submodular.
The best known algorithm for minimizing general submodular functions has a worst case complexity of $\mathcal{O}(N^5\mathcal{Q} + N^6)$ [Orlin, 2009], where $\mathcal{Q}$ is the time taken to evaluate the function. This scaling behavior makes the algorithm impractical for the Stixel assignment task, where N is in the order of 300.
However, the subclass of submodular functions of arity 2 (quadratic pseudo-boolean functions) can be optimized far more efficiently, since this task was shown to be equivalent to finding the so-called minimum cut of the corresponding graph [Ivănescu, 1965]. The minimum cut of a weighted or unweighted graph [Cormen et al., 2001] is a partitioning of its vertices into two disjoint subsets whose cutset has the smallest number of elements (unweighted case) or smallest sum of weights possible (weighted case). The theorem of Ford and Fulkerson [Ford and Fulkerson, 1962] states that the task of computing the minimum cut is equivalent to computing the maximum flow in a flow network that can be pushed from a source terminal to a sink terminal, see Figure 2.14 for an illustration. There are various implementations of the Ford–Fulkerson method, see [Goldberg and Tarjan, 1988] for an overview. The graphcut implementation used in present work was proposed by [Boykov and Kolmogorov, 2004]. This method runs in $\mathcal{O}(N^2 \cdot \mathcal{E} \cdot |C^*|)$, where $\mathcal{E}$ denotes the number of edges and $|C^*|$ is the cost of the minimum cut. In practice the algorithm outperforms other standard maxflow implementations such as Dinic's algorithm [Dinic, 1970] which is $\mathcal{O}(N^2 \cdot \mathcal{E})$ in spite of the higher theoretical complexity [Boykov and Kolmogorov, 2004].
Besides the efficient special case of submodular quadratic pseudo boolean functions, approximations for more general functions are required. One important case which is also required in this thesis is general multi-label optimization with more than two classes.

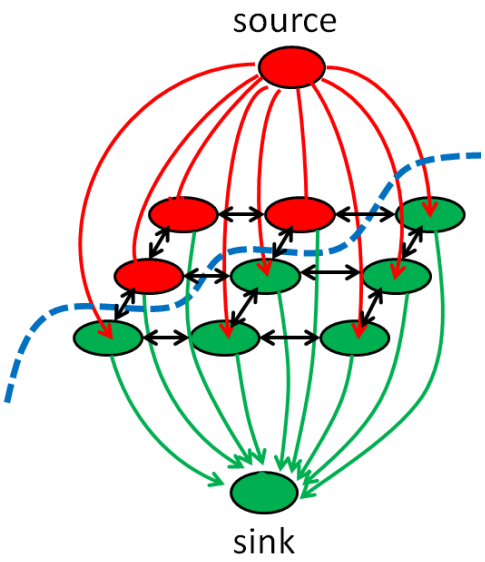


Figure 2.14: Visualization of the graphcut concept. All nine Stixels of one image are either assigned to the source terminal shown in red or to the sink terminal shown in green. The black arrows indicate the connectivity of neighboring nodes. The resulting minimum cut shown in blue maximizes Equation 2.57 for submodular functions of arity 2.

The move-making α-expansion and α-β-swap algorithms [Boykov et al., 2001] are such approximations for the more general multi-label case. These move-making algorithms allow several nodes in the graph to change their current labeling $\mathbf{L}$ simultaneously to a new labeling $\mathbf{L}^{\text{new}}$ that is within one move from the current labeling. Both algorithms choose the move with the largest decrease in energy, a single move can be found efficiently by means of the basic graphcut algorithm [Kolmogorov and Zabin, 2004]. The move-making algorithms start with an initial labeling $\mathbf{L}$ and sequentially step forward to a new labeling $\mathbf{L}^{\text{new}}$ within one move from the current labeling which leads to the largest decrease in energy. This step is repeated several times until the energy stops decreasing or until a maximum number of steps is reached.
If only a single pixel label change is considered for an admissible move, the well-known ICM algorithm [Besag, 1986, Kittler and Föglein, 1984] can be seen as a move-making algorithm as well.
The α-β-swap algorithm is applicable for all pairwise energies whenever the following condition is satisfied [Blake et al., 2011]:

$$E(\alpha,\alpha) + E(\beta,\beta) \leq E(\alpha,\beta) + E(\beta,\alpha) \quad \forall \alpha,\beta \in \mathcal{J}. \tag{2.60}$$

Equation 2.60 is satisfied if E is a metric, quasi-metric, or semi-metric. Note that only the binary term matters, since each unary term is submodular as stated above and the sum of submodular functions is again submodular. In Equation 2.60, $\mathcal{J}$ again denotes the set of possible labelings.

Similarly, the expansion algorithm can be used whenever

$$E(\alpha,\alpha)+E(\beta,\gamma)\leq E(\beta,\alpha)+E(\alpha,\gamma)\quad\forall\alpha,\beta,\gamma\in\mathcal{J} \tag{2.61}$$

is satisfied which holds if E is a metric or quasi-metric [Blake et al., 2011].

1 Start with an arbitrary initial labeling $\mathbf{L}$;
2 Set success := 0;
3 **for** *each label* $\alpha\subset\mathcal{J}$ **do**
  3.1 Find $\mathbf{L}^*=\arg\min E(\mathbf{L}^{\text{new}})$ among $\mathbf{L}^{\text{new}}$ within one α-expansion of $\mathbf{L}$;
  3.2 **if** $E(\mathbf{L}^*)<E(\mathbf{L})$ **then**
    set $\mathbf{L}:=\mathbf{L}^*$ and success := 1
  end
end
4 If success = 1 goto **2**;
5 Return $\mathbf{L}$;

Algorithm 1 : The α-expansion algorithm [Boykov et al., 2001].

1 Start with an arbitrary initial labeling $\mathbf{L}$;
2 Set success := 0;
3 **for** *each pair of labels* $\{\alpha,\beta\}\subset\mathcal{J}$ **do**
  3.1 Find $\mathbf{L}^*=\arg\min E(\mathbf{L}^{\text{new}})$ among $\mathbf{L}^{\text{new}}$ within one α-β-swap of $\mathbf{L}$;
  3.2 **if** $E(\mathbf{L}^*)<E(\mathbf{L})$ **then**
    set $\mathbf{L}:=\mathbf{L}^*$ and success := 1
  end
end
4 If success = 1 goto **2**;
5 Return $\mathbf{L}$;

Algorithm 2 : The α-β-swap algorithm [Boykov et al., 2001].

A detailed outline of both the α-expansion algorithm and the α-β-swap algorithm is given in Algorithm 1 and 2. To sum up, the α-expansion algorithm is faster than the α-β-swap algorithm but the latter is more general. One cycle in the swap algorithm takes J^2 iterations, whereas a cycle in the expansion algorithm only takes J iterations. Interestingly, the solution of the expansion algorithm meets certain optimality conditions. The local minimum obtained from the α-expansion algorithm is within a known factor of the global minimum. This factor can be as small as 2 and depends on $E(l_i,l_j)$. More specifically, let $\mathbf{L}^*_\alpha$ be a local minimum obtained from the α-expansion algorithm and let $\mathbf{L}^*$ be the global optimum. In this case,

$$E(\mathbf{L}^*_\alpha)\leq 2\Gamma E(\mathbf{L}^*) \tag{2.62}$$

with

$$\Gamma=\max_{i,j\in\mathcal{E}}\left(\frac{\max_{\alpha\neq\beta\in\mathcal{J}}E(l_i=\alpha,l_j=\beta)}{\min_{\alpha\neq\beta\in\mathcal{J}}E(l_i=\alpha,l_j=\beta)}\right) \tag{2.63}$$

was shown to hold [Boykov et al., 2001], that is the optimality of the α-expansion algorithm is bounded by the ratio of the largest nonzero value of $E(l_i = \alpha, l_j = \beta)$ to the smallest nonzero value of $E(l_i = \alpha, l_j = \beta)$. Note that Γ is well defined since $E(l_i = \alpha, l_j = \beta) \neq 0$ for $\alpha \neq \beta$ because E is a metric. This bound is rather of theoretical significance, usually the solution of the α-expansion algorithm is much closer to the global optimum. In practice, the significance of this bound is limited since even very poor solutions might fall inside this bound. This bound is not tight enough.
For the α-β-swap algorithm a comparable bound does not exist.
Recently, a strong interest in higher-order Random Fields has emerged, to a large extent due to the development of efficient inference methods [Kohli et al., 2009, Delong et al., 2012]. The main advantage of the higher-order terms is the possibility to model long range interactions that cannot be expressed via pairwise terms. Furthermore, pairwise CRFs tend to oversmooth and quite often do not match the actual object contour [Kohli et al., 2009].
In [Ishikawa, 2009,Rother et al., 2009], it was shown that any general higher-order MRF with binary labels can be reduced to a first-order one with unary and pairwise clique potentials. However, in general the number of required additional auxiliary variables increases exponentially as the order of the given energy increases. Therefore, inference in dense Random Fields is often intractable. Accordingly, researchers have focused on fast special cases that need significantly less auxiliary variables, see [Kohli and Kumar, 2010] for examples.
In this work, the Bayesian Information Criterion (BIC) [Schwarz, 1978] is taken into account, a penalty that is added whenever a certain class α is present in the segmentation $\mathbf{L}$ to avoid overfitting and dispensable classes. The BIC term can be rewritten as

$$E_{BIC} = \sum_{L \in \mathcal{J}} h_L \delta_L(\mathbf{L}), \tag{2.64}$$

where h_L is a constant for class L and $\delta_L(\mathbf{L})$ is an indicator function defined as

$$\delta_L(\mathbf{L}) := \begin{cases} 1, & \text{if } \exists i : l_i \in L \\ 0, & \text{otherwise.} \end{cases} \tag{2.65}$$

Usually, a single α-expansion step is defined as follows [Delong et al., 2012]. A node i is said to be *active* if it changes its current labeling l_i^{current} that is currently not α to α. In this case, the binary variable $y_i := 1$. Otherwise, if it keeps its current labeling, $y_i := 0$. This way, the energy of an α-expansion step is augmented by the binary variables $\{y_i\}$, $i = 1...N$ resulting in

$$E(\mathbf{L}^{\text{new}}) = E^{\alpha}(\mathbf{L}^{\text{new}}) + \sum_{L \in \mathbf{L}^{\text{current}}} \left(h_L - h_L \cdot \prod_{i \in \mathbf{P}_L} y_i \right) + \sum_{L \notin \mathbf{L}^{\text{current}}} \left(h_L - h_L \cdot \prod_i \bar{y}_i \right). \tag{2.66}$$

In this equation, $E^{\alpha}(\mathbf{L}^{\text{new}})$ denotes the energy of the ordinary α-expansion step as given in Algorithm 1 and $\mathbf{P}_L$ is the set of nodes that currently have the label L

$$\mathbf{P}_L = \left\{ i : l_i^{\text{current}} \in L \right\}. \tag{2.67}$$

This means the second term on the right-hand side is zero if the labeling L is saved and is not present any longer in the new labeling $\mathbf{L}^{\text{new}}$. In this case, all the nodes that currently have this labeling L are active, $y_i = 1 \ \forall i \in \mathbf{P}_L$ and the second term becomes zero. Before, the second term was $h_L > 0$ which corresponds to a penalty. The third term accounts for the case that the labeling $L = \alpha$ was not used at all in $\mathbf{L}^{\text{current}}$, that is $\mathbf{P}_L = \emptyset$, but is now introduced in the new labeling $\mathbf{L}^{\text{new}}$. In this case, $\bar{y}_i := 1 - y_i = 0$ holds for at least one node and the penalty h_L is added to the current energy.
The higher-order term $h_L \cdot \prod_{i \in \mathbf{P}_L} y_i$ can be transformed into a binary energy by introducing only one additional auxiliary variable $q_L \in \{0, 1\}$ [Freedman and Drineas, 2005]

$$- h_L \cdot \prod_{i \in \mathbf{P}_L} y_i = \min_{q_L \in \{0,1\}} h_L \left[(|\mathbf{P}_L| - 1)\, q_L - \sum_{i \in \mathbf{P}_L} y_i q_L \right], \tag{2.68}$$

which can be optimized by means of graphcut, since the binary terms $y_i q_L$ have negative coefficients, so the second derivatives are non-positive and thus submodular [Boros and Hammer, 2002]. The third term can be transformed analogously, see [Delong et al., 2012].
Even in the presence of label costs, a similar bound to Equation 2.62 can be proven. In [Delong et al., 2012]

$$E\left(\mathbf{L}_\alpha^*\right) \leq (2\Gamma + \Upsilon)\, E\left(\mathbf{L}^*\right) + \sum_{L \subset \mathcal{J}} h_L \tag{2.69}$$

is proven, where Γ was already introduced in Equation 2.63 and

$$\Upsilon = \max_{L \subset \mathcal{J}, h_L > 0} |\mathbf{P}_L| - 1. \tag{2.70}$$

$|\mathbf{P}_L|$ denotes the cardinality of class L, so Υ represents the largest class subset of $\mathcal{J}$. This equation holds for a metric binary term and positive unary terms [Delong et al., 2012]. This means that for modest label costs the bound is similar as in case for the ordinary α-expansion algorithm 2.62. However, as the number of nodes in the largest subset increases, the bound worsens. Finally, the bound is poor if the label costs are arbitrarily large.

3 Graphcut-based Object Segmentation

The detection of moving objects like vehicles, pedestrians or bicycles from a mobile platform is one of the most challenging and most important tasks for driver assistance and safety systems.
For this purpose, this chapter presents a multi-class traffic scene segmentation approach based on the Dynamic Stixel World, an efficient superpixel object representation that is briefly introduced in Section 2.5.
The aim of the segmentation is a temporally consistent decomposition of an image sequence $\mathcal{I}$ into various, non-overlapping objects and a state estimation of their relevant object parameters.
In order to reduce complexity and to exploit redundancies in temporal image sequences, a two-stage optimization strategy is proposed. This optimization scheme alternates in an expectation maximization-like (EM) [Dempster et al., 1977] fashion between estimation steps of the unknown object parameters and assignment steps of the Stixels to the object classes.
Parts of this work have already been published in [Erbs et al., 2012b,Erbs et al., 2012a].
This chapter is structured as follows: Section 3.1 introduces and motivates the underlying segmentation problem. Section 3.2 derives the presented approach from probability theory and Section 3.3 further specifies the most important modeling-related aspects. In Section 3.4 the inference step and the estimation step of the hidden object parameters are described. Section 3.5 presents experimental results. Finally Section 3.6 concludes this chapter.

3.1 Introduction

This chapter presents a stereo image-based approach for object segmentation of real-world traffic scenes in the field of cognitive automobiles and driver assistance.
As stated above, the segmentation step has the objective to decompose an image sequence $\mathcal{I}$ consistently over time into various moving objects and stationary background and to estimate their underlying object properties.
The segmentation is one example for a so-called *missing data problem* [Dempster et al., 1977]. The segmentation step involves determining which object in the scene has most probably generated the Stixels in the image. If it was known which Stixel came from which object, it would be easy to estimate the relevant object properties $\boldsymbol{\Theta}$ by a maximum likelihood estimation for example.
Vice versa, if the object properties $\boldsymbol{\Theta}$ were known, it would be possible to determine the object that has most likely generated a Stixel. This assignment step is commonly known as a *labeling problem* [Li, 2009]. So the estimation of $\boldsymbol{\Theta}^t$ and the labeling $\mathbf{L}^t$ strongly depend on each other. The difficulty is that none of them are known [Forsyth and Ponce, 2002].

The labeling concept was already introduced in subsection 2.6.1. A labeling problem was defined as an assignment of a labeling $\mathbf{L}^t$ from the finite set of classes $\mathcal{J}^t$ to all sites $i \in \mathcal{S}^t$, where the elements in $\mathcal{S}^t$ index the Stixels in an image $I^t \in \mathcal{I}$.
The set

$$\mathbf{L}^t = \{l_1^t, l_2^t, ..., l_N^t\} \tag{3.1}$$

is called a labeling of the sites $\mathcal{S}^t$ at time t in terms of the classes $\mathcal{J}^t$.
A labeling $\mathbf{L}^t$ is one specific realization of the random variable $\mathbb{L}^t$ representing the underlying set of all possible labelings. It can also be understood as a mapping from $\mathcal{S}^t$ to $\mathcal{J}^{N,t}$,

$$\mathbf{L}^t : \mathcal{S}^t \rightarrow \mathcal{J}^{N,t}. \tag{3.2}$$

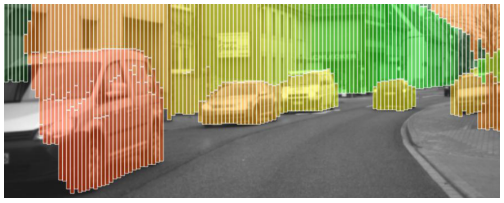
Furthermore, a posterior probability $p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathbf{L}^0, ..., \mathbf{L}^{t-1}\right)$ is introduced in the sense of a functional mapping from the valid label configuration space $\mathbb{L}^t$ to the real number interval $[0, 1]$. In this posterior probability, $\mathcal{Z}^t$ denotes a set of observations for all N Stixels. As mentioned already in Subsection 2.6.2, the observations $\mathcal{Z}^t$ are themselves considered to be instantiations of a random variable $\mathbb{Z}^t$ representing the full space of observations that can arise [Blake et al., 2011].
The aim of the segmentation is to find the most probable labeling defined as

$$\mathbf{L}^* = \arg\max_{\mathbf{L}^t \in \mathbb{L}^t} p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathbf{L}^0, ..., \mathbf{L}^{t-1}\right). \tag{3.3}$$

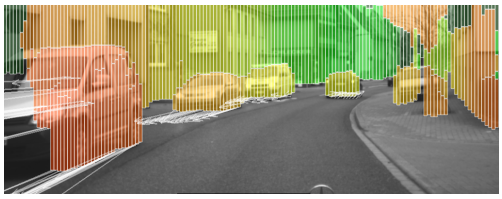
Note that the object parameter set $\mathbf{\Theta}^t = \{\Theta_1^t, ..., \Theta_M^t\}$ is considered as a hidden parameter in this posterior probability.
In general, for most cases this optimization problem is known to be NP-complete [Lempitsky et al., 2010], even if the parameter set $\mathbf{\Theta}^t$ was known. Taking into account $\mathbf{\Theta}^t$, the label space $\left|\mathbb{L}^t\right| = \left|\mathcal{J}^t\right|^N \otimes \left|\mathbf{\Theta}^t\right|$ becomes continuous and therefore uncountable. Besides that, the true number of objects, i.e. the dimension of $\mathbf{\Theta}^t$ is unknown in general.
In the following, occasionally the time index t will be omitted when only the current time step is considered and there is no risk of confusion.
In order to find the maximum a posteriori (MAP) solution defined in Equation 3.3, in this work a time-iterative two-stage optimization scheme is applied. The proposed approach tries to identify so-called *unknown moving objects* using the α-expansion graphcut scheme introduced in Subsection 2.6.2. For these unknown moving objects, their complete object state vector Θ_n^t is not yet known.
The proposed algorithm explicitly exploits the fact that typically images from a stereo image sequence $\mathcal{I}$ are strongly correlated. Hence, the optimization does not need to be performed on a single image, but can be split up to consecutive images. This way, the relevant object parameters are estimated over time and a detected unknown moving object becomes a *known moving object* in the next frame.
In contrast to this approach, single frame image segmentation might yield temporally inconsistent labeling decisions that are difficult to interpret.
Furthermore, whereas in classical EM approaches [Dempster et al., 1977] the number of object classes is usually assumed to be known in advance, the proposed approach includes this information as part of the MAP estimation problem.
The main steps of the presented approach are summarized in Figure 3.1.

(a) Left original gray value image.

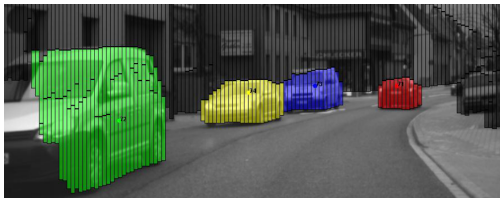
(b) SGM [Gehrig et al., 2009] stereo image. The distance is color encoded ranging from red (close) to green (far away).



(c) Multi-Layered Stixel World [Pfeiffer, 2012].



(d) Dynamic Stixel World [Scharwaechter, 2012]. The arrows show the predicted position of the Stixels for the next half second.



(e) Object segmentation result. The color encodes the different object classes, stationary background is shown in black.

Figure 3.1: Processing chain of the segmentation.

Firstly, a dense depth image is computed. In the experiments, the *Semi-Global Matching* (SGM) algorithm [Hirschmuller, 2005, Gehrig et al., 2009] is used, as shown in Figure 3.1(b), see also Subsection 2.2 for a brief introduction. SGM is a very efficient and powerful dense stereo algorithm that can be computed in real-time on a special FPGA platform without burdening the CPU.
Secondly, the multi-layered *Stixel World* [Pfeiffer and Franke, 2011] shown in Figure 3.1(c) is computed. This way, the relevant depth information in the scene is represented with a few hundred Stixels instead of hundreds of thousands of individual stereo depth measurements. Consequently, using the Stixel World instead of dense depth and motion information enables to reduce the computational burden for subsequent driver assistance applications, in some cases by several orders of magnitude, see e.g. [Benenson et al., 2012b, Benenson et al., 2012a, Erbs et al., 2012b, Enzweiler et al., 2012, Benenson et al., 2011]. The Stixel World yields the important freespace information and it provides an efficient object representation that is required by many driver assistance systems. Besides that, it is insensitive to stereo measurement outliers which boosts the robustness of subsequent algorithms.
Thirdly, the Stixels are tracked over time to estimate their motion state highly accurately by applying the 6D-Vision principle introduced in [Franke et al., 2005, Pfeiffer and Franke, 2010, Scharwaechter, 2012], see Section 2.5. This step fuses optical flow information and stereo information by means of temporal Kalman filtering as shown in Figure 3.1(d). This so-called *Dynamic Stixel World* serves as input to the proposed approach.
The objective of the segmentation is to find a complete but minimal scene description in terms of objects rather than Stixels that is required by many subsequent algorithms. The Dynamic Stixel World does not contain any information about the relations of the Stixels to each other. This independence assumption could result in wrong or inconsistent scene interpretations. Furthermore, a minimal scene description aggregates as much information as possible and it is characterized by its stability, precision and efficiency. For these reason, this chapter introduces a segmentation step that partitions the Dynamic Stixel World into several moving objects and stationary background, exemplarily shown in Figure 3.1(e).

3.2 Optimization Problem

In this section, the segmentation task is formulated as a Bayesian optimization problem. First of all, the input data of the segmentation is introduced more formally.
The given stereo camera system records an image sequence $\mathcal{I}$ for which a dense depth reconstruction is computed via the SGM algorithm, see Figure 3.1(b).
The dense stereo measurements are subsequently segmented into the multi-layered Stixel World as proposed in [Pfeiffer, 2012]. This (static) Stixel World partitions an input image $I^t \in \mathcal{I}$ column-wise into several layers of one of the two classes $\mathcal{C}_{\text{Stixel}} \in \{\text{street, obstacle}\}$, cf. 3.1(c). In the following, the street area is left unchanged and the focus is on obstacle Stixels.
Subsequently, the Stixels are tracked over time in order to estimate their motion state. In summary, each Stixel with index i is defined by solely five observations. That is its 3D world position $(X_i^t,\, Y_i^t,\, Z_i^t)$, where Y_i^t denotes the height of the Stixel relative to the camera coordinate system (the top point), and its velocity $\left(\dot{X}_i^t,\, \dot{Z}_i^t\right)$. Moving objects such as cars or bicycles are assumed to move on the ground plane, so it is sufficient to estimate a 2D motion vector. Throughout this work, a left-handed coordinate system is assumed if not stated otherwise where the Z-axis is defined by the current moving direction of the ego vehicle (straight ahead) and the positive X-axis points to the right. These five observations form a feature vector for each Stixel,

$$\vec{z}_i^t = \left(\dot{X}_i^t,\, \dot{Z}_i^t,\, X_i^t,\, Y_i^t,\, Z_i^t\right)^{\mathrm{T}}. \tag{3.4}$$

These feature vectors are again combined in a measurement array

$$\mathcal{Z}^t = \left(\vec{z}_1^t,\, \ldots,\, \vec{z}_N^t\right). \tag{3.5}$$

Now, let

$$\mathbf{L}^t = \left(l_1^t,\, \ldots, l_N^t\right)^{\mathrm{T}} \tag{3.6}$$

denote a labeling for a given input image I^t containing N dynamic Stixels. The number $\mathcal{J}^t = M + 1$ of object classes varies dynamically as the number of moving objects M does in real traffic scenes. A labeling $\mathbf{L}^t$ assigns each Stixel to exactly one moving object class or to static background,

$$l_i^t \in \{O_1,\, O_2,\, \ldots O_M,\, \mathrm{Bg}\}. \tag{3.7}$$

In order to proceed, first of all,

$$p\left(\mathbf{L}^t \mid \mathcal{Z}^t,\, \mathbf{L}^0, \ldots, \mathbf{L}^{t-1}\right) \approx p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathbf{L}^{t-1}\right) \tag{3.8}$$

is assumed since the objective is to favor temporally consistent labeling decisions which can be accomplished in the most efficient case by a first-order correlation. This step helps to simplify the following terms enormously. The previous segmentation $\mathbf{L}^{t-1}$ is assumed to be given and is fixed. The inference step can be easily extended to multiple previous time steps. However, in the present work this step was omitted since this would be computationally more demanding and it would be more expensive to collect

these statistics.
Applying Bayes' theorem yields

$$p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathbf{L}^{t-1}\right) \propto p\left(\mathcal{Z}^t, \mathbf{L}^{t-1} \mid \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^t\right) = p\left(\mathcal{Z}^t \mid \mathbf{L}^{t-1}, \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^{t-1} \mid \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^t\right). \tag{3.9}$$

Additionally, the current labeling $\mathbf{L}^t$ is further assumed to be sufficient to account for the observations $\mathcal{Z}^t$, i.e. the measurements at time t are independent of the elder labeling $\mathbf{L}^{t-1}$ given the most recent labeling $\mathbf{L}^t$. This is a common assumption in many Bayesian state estimation processes [Thrun et al., 2005]. It allows to simplify

$$p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathbf{L}^{t-1}\right) \propto \underbrace{p\left(\mathcal{Z}^t \mid \mathbf{L}^t\right)}_{\text{Data Term}} \cdot \underbrace{p\left(\mathbf{L}^{t-1} \mid \mathbf{L}^t\right)}_{\text{Temporal Expectation}} \cdot \underbrace{p\left(\mathbf{L}^t\right)}_{\text{Prior Term}}. \tag{3.10}$$

According to the Hammersley-Clifford theorem [Clifford, 1990], a probability function that satisfies the Markov properties 2.56 factorizes into the product of potential functions $\psi_{\mathcal{C}}\left(\mathcal{Z}_{\mathcal{C}}^t \mid \mathbf{L}^t\right)$ over the maximal cliques $\mathcal{C}$ of the graph, for example

$$p\left(\mathcal{Z}^t \mid \mathbf{L}^t\right) = \frac{1}{\Omega} \prod_{\mathcal{C}} \psi_{\mathcal{C}}\left(\mathcal{Z}_{\mathcal{C}}^t \mid \mathbf{L}^t\right), \tag{3.11}$$

where the partition function Ω is a normalization constant

$$\Omega = \sum_{\mathcal{Z}_{\mathcal{C}}^t} \prod_{\mathcal{C}} \psi_{\mathcal{C}}\left(\mathcal{Z}_c^t \mid \mathbf{L}^t\right). \tag{3.12}$$

See also Equation 2.57. The potential functions are arbitrary positive functions $\psi_{\mathcal{C}}\left(\mathcal{Z}_c^t \mid \mathbf{L}^t\right) \geq 0$.
However, in Equation 3.11 the true maximum clique size is unknown. As introduced in Subsection 2.6.2, inference in dense random fields is often intractable.
Nevertheless, the desired property of smooth solutions can be enforced with modest clique sizes, in the simplest case via pairwise cliques, for which very efficient inference algorithms already exist. Consequently, in the present work the probability of a labeling $\mathbf{L}^t$ is modeled as a Conditional Random Field [Lafferty et al., 2001] with a maximum clique size of two. With this assumption Equation 3.10 becomes

$$\begin{aligned} p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathbf{L}^{t-1}\right) \propto & \prod_i p\left(\vec{z}_i^t \mid \mathbf{L}^t\right) \cdot \\ & \prod_{(i,j)\in\mathcal{N}_2} p\left(\vec{z}_i^t, \vec{z}_j^t \mid \mathbf{L}^t\right) \cdot \\ & \prod_i p\left(l_i^{t-1} \mid \mathbf{L}^t\right) \cdot \\ & \prod_i p\left(l_i^t\right) \cdot \\ & \prod_{(i,j)\in\mathcal{N}_2} p\left(l_i^t, l_j^t\right). \end{aligned} \tag{3.13}$$

In this context, $\mathcal{N}_2$ denotes the set of all neighboring Stixels. The unary terms can be added in the factorization 3.13 since the potential functions are arbitrary, non-negative functions over the maximal cliques of the graph and they can be multiplied by any non-negative function of a subset of the cliques without loosing this property.
Furthermore, there is no reason not to normalize the potential functions $\psi_{\mathcal{C}}$ since the normalization constant is absorbed by the global normalization constant Z. So the potential functions can be transformed into real probability densities, hereinafter referred to as p. Next, the observations $\vec{z}_i^t$ and $\vec{z}_j^t$ are assumed to be dependent primarily on the current labeling l_i^t and l_j^t at that site. Formally, it is required that

$$p\left(\vec{z}_i^t, \vec{z}_j^t \mid \mathbf{L}^t\right) \approx p\left(\vec{z}_i^t, \vec{z}_j^t \mid l_i^t, l_j^t\right). \tag{3.14}$$

holds. Similarly,

$$p\left(l_i^{t-1} \mid \mathbf{L}^t\right) \approx p\left(l_i^{t-1} \mid l_i^t\right) \tag{3.15}$$

is stipulated. This means that for temporal regularization, the class choice of Stixel i at time $t-1$ is primarily correlated with the class choice of its immediate successor Stixel in the next frame t. The correspondence is based on optical flow. With these simplifications, Equation 3.13 becomes

$$\begin{aligned}
p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathbf{L}^{t-1}\right) \propto & \prod_i p\left(\vec{z}_i^t \mid l_i^t\right) \cdot \\
& \prod_{i,j \in \mathcal{N}_2} p\left(\vec{z}_i^t, \vec{z}_j^t \mid l_i^t, l_j^t\right) \cdot \\
& \prod_i p\left(l_i^{t-1} \mid l_i^t\right) \cdot \\
& \prod_i p\left(l_i^t\right) \cdot \\
& \prod_{i,j \in \mathcal{N}_2} p\left(l_i^t, l_j^t\right) \\
\propto & \prod_i p\left(\vec{z}_i^t \mid l_i^t\right) \cdot \\
& \prod_i p\left(l_i^{t-1} \mid l_i^t\right) \cdot \\
& \prod_i p\left(l_i^t\right) \cdot \\
& \prod_{i,j \in \mathcal{N}_2} p\left(l_i^t, l_j^t \mid \vec{z}_i^t, \vec{z}_j^t\right).
\end{aligned} \tag{3.16}$$

Hence, under these assumptions, the most probable labeling defined in Equation 3.3 minimizes the following negative log-likelihood energy $E := -\log p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathbf{L}^{t-1}\right)$ [Gray,

1990]

$$E = \underbrace{-\sum_{i=1}^{N} \log p\left(l_i^t\right)}_{:=\log Q(\mathbf{L}^t)} \underbrace{-\sum_{i=1}^{N} \log p\left(l_i^{t-1} \mid l_i^t\right)}_{:=\log Q(\mathbf{L}^{t-1}|\mathbf{L}^t)} - \underbrace{\sum_{i=1}^{N} \log p\left(\vec{z}_i^{\,t} \mid l_i^t\right)}_{:=\log Q(\mathcal{Z}^t|\mathbf{L}^t)} - \sum_{(i,j)\in\mathcal{N}_2} \log p\left(l_i^t, l_j^t \mid \vec{z}_i^{\,t}, \vec{z}_j^{\,t}\right). \tag{3.17}$$

As the next step, the hidden object parameter set $\mathbf{\Theta}^t$ is introduced by integrating the data term over all possible realizations of these random variables. This parameter set describes the state of the M moving objects in the scene, that is the position of the n-th object described by its geometric center $\mathcal{X}_n^t$ and $\mathcal{Z}_n^t$, the object velocity $\dot{\mathcal{X}}_n^t$ and $\dot{\mathcal{Z}}_n^t$, and its object dimensions, namely the object width $|\Delta\mathcal{X}|_n^t$, height $\mathcal{H}_n^t$ and length $|\Delta\mathcal{Z}|_n^t$:

$$\begin{aligned} &\mathbf{\Theta}^t = \{\Theta_1^t, \ldots, \Theta_M^t\} \text{ and} \\ &\Theta_n^t = \{\dot{\mathcal{X}}_n^t, \dot{\mathcal{Z}}_n^t, \mathcal{X}_n^t, \mathcal{Z}_n^t, |\Delta\mathcal{X}|_n^t, \mathcal{H}_n^t, |\Delta\mathcal{Z}|_n^t\}. \end{aligned} \tag{3.18}$$

In contrast to the local pairwise factorization in Equation 3.13, these hidden object parameters establish long-range correlations between Stixels. This way, the global energy function 3.17 becomes

$$E = -\log Q\left(\mathbf{L}^t\right) - \log Q\left(\mathbf{L}^{t-1} \mid \mathbf{L}^t\right) - \log \int_{\mathbf{\Theta}^t} Q\left(\mathcal{Z}^t, \mathbf{\Theta}^t \mid \mathbf{L}^t\right) \mathrm{d}\mathbf{\Theta}^t - \sum_{(i,j)\in\mathcal{N}_2} \log p\left(l_i^t, l_j^t \mid \vec{z}_i^{\,t}, \vec{z}_j^{\,t}\right). \tag{3.19}$$

A Taylor expansion of the integrand using the Laplace method [Sivia, 1996] gives

$$\begin{aligned} \log Q\left(\mathcal{Z}^t, \mathbf{\Theta}^t \mid \mathbf{L}^t\right) &= \log Q\left(\mathcal{Z}^t \mid \mathbf{L}^t, \mathbf{\Theta}^t\right) Q\left(\mathbf{\Theta}^t \mid \mathbf{L}^t\right) \\ &\approx \log Q\left(\mathcal{Z}^t \mid \mathbf{L}^t, \mathbf{\Theta}_{map}^t\right) Q\left(\mathbf{\Theta}_{map}^t \mid \mathbf{L}^t\right) \\ &\quad - \frac{1}{2}\left(\mathbf{\Theta}^t - \mathbf{\Theta}_{map}^t\right)^T \mathcal{A}\left(\mathbf{\Theta}^t - \mathbf{\Theta}_{map}^t\right) + \ldots, \end{aligned} \tag{3.20}$$

where $\mathbf{\Theta}_{map}^t$ denotes the values of $\mathbf{\Theta}^t$ at the mode of the integrand and $\mathcal{A}$ is the negative Hessian matrix of second derivatives

$$\mathcal{A} = -\nabla_{\mathbf{\Theta}}\nabla_{\mathbf{\Theta}} \log Q\left(\mathcal{Z}^t \mid \mathbf{L}^t, \mathbf{\Theta}_{map}^t\right) Q\left(\mathbf{\Theta}_{map}^t \mid \mathbf{L}^t\right). \tag{3.21}$$

A proof of the validity of this approximation can be found in the Appendix 6.1. In this case, the integrand is approximated as

$$Q\left(\mathcal{Z}^t \mid \mathbf{L}^t, \mathbf{\Theta}^t\right) Q\left(\mathbf{\Theta}^t \mid \mathbf{L}^t\right) \approx Q\left(\mathcal{Z}^t \mid \mathbf{L}^t, \mathbf{\Theta}_{map}^t\right) Q\left(\mathbf{\Theta}_{map}^t \mid \mathbf{L}^t\right) \cdot \exp\left(-\frac{1}{2}\left(\mathbf{\Theta}^t - \mathbf{\Theta}_{map}^t\right)^T \mathcal{A}\left(\mathbf{\Theta}^t - \mathbf{\Theta}_{map}^t\right)\right). \tag{3.22}$$

This results in an ordinary Gaussian integral that can be solved analytically

$$\int_{\mathbf{\Theta}^t} \exp\left(-\frac{1}{2}\left(\mathbf{\Theta}^t - \mathbf{\Theta}^t_{map}\right)^T \mathcal{A}\left(\mathbf{\Theta}^t - \mathbf{\Theta}^t_{map}\right)\right) \mathrm{d}\mathbf{\Theta}^t = \frac{(2\pi)^{K/2}}{|\mathcal{A}|^{1/2}}. \tag{3.23}$$

In Equation 3.23, K denotes the dimension of $\mathbf{\Theta}^t$. So Equation 3.19 can be approximated as

$$\begin{aligned} E \approx & -\log Q\left(\mathbf{L}^t\right) - \log Q\left(\mathbf{L}^{t-1} \mid \mathbf{L}^t\right) \\ & - \log Q\left(\mathcal{Z}^t, \mid \mathbf{\Theta}^t_{map}, \mathbf{L}^t\right) - \log Q\left(\mathbf{\Theta}^t_{map} \mid \mathbf{L}^t\right) \\ & - \frac{K}{2}\log(2\pi) + \frac{1}{2}\log(|\mathcal{A}|) - \sum_{(i,j)\in\mathcal{N}_2} \log p\left(l^t_i,\, l^t_j \mid \vec{z}^t_i, \vec{z}^t_j\right). \end{aligned} \tag{3.24}$$

Furthermore, the negative Hessian matrix $\mathcal{A}$ can be approximated

$$\begin{aligned} \mathcal{A} &= -\nabla_{\mathbf{\Theta}}\nabla_{\mathbf{\Theta}} \log Q\left(\mathcal{Z}^t \mid \mathbf{L}^t,\, \mathbf{\Theta}^t_{map}\right) Q\left(\mathbf{\Theta}^t_{map} \mid \mathbf{L}^t\right) \\ &= -\nabla_{\mathbf{\Theta}}\nabla_{\mathbf{\Theta}} \log Q\left(\mathcal{Z}^t \mid \mathbf{L}^t,\, \mathbf{\Theta}^t_{map}\right) - \nabla_{\mathbf{\Theta}}\nabla_{\mathbf{\Theta}} \log Q\left(\mathbf{\Theta}^t_{map} \mid \mathbf{L}^t\right) \\ &\approx -\nabla_{\mathbf{\Theta}}\nabla_{\mathbf{\Theta}} \log Q\left(\mathcal{Z}^t \mid \mathbf{L}^t,\, \mathbf{\Theta}^t_{map}\right). \end{aligned} \tag{3.25}$$

This approximation is valid since the prior $Q\left(\mathbf{\Theta}^t \mid \mathbf{L}^t\right)$ is assumed to be broader than the data likelihood $Q\left(\mathcal{Z}^t \mid \mathbf{L}^t,\, \mathbf{\Theta}^t_{map}\right)$. For a multivariate Gaussian distribution, for example, $-\nabla_{\mathbf{\Theta}}\nabla_{\mathbf{\Theta}} \log \mathcal{N}\left(\mathbf{\Theta}^t, \vec{\mu}, \mathbf{\Sigma}\right) = \mathbf{\Sigma}^{-1}$ corresponds to the precision matrix of the distribution. For most probability distributions considered in this work, only vague prior knowledge on the object parameters exists. As a general rule, the precision with which object parameters as those defined in Equation 3.18 can be observed is significantly higher than the precision arising from prior knowledge. Traffic scenes in general are extremely versatile and often the prior parameter distributions can only be limited by plausibility values.
Assuming that the measurements $\mathcal{Z}^t$ originate from statistically independent degradations, i.e. the measurements are conditionally independent, given the object parameters

Θ_n [Mester, 2012], allows to further simplify the determinant

$$\begin{aligned}
\log\left(|\mathcal{A}|\right) &\overset{3.25}{\approx} \log\left(|-\nabla\nabla\log Q\left(\mathcal{Z}^t \mid \mathbf{L}^t,\, \mathbf{\Theta}^t_{map}\right)|\right) \\
&= \log(|-\nabla\nabla\log\prod_i^N Q\left(\vec{z}_i^{\,t} \mid \mathbf{L}^t,\, \mathbf{\Theta}^t_{map}\right)|) \\
&= \log(|\sum_i^N \underbrace{\left(-\nabla\nabla\log Q\left(\vec{z}_i^{\,t} \mid \mathbf{L}^t,\, \mathbf{\Theta}^t_{map}\right)\right)}_{=:A_i}|) \\
&= \log(|\sum_i^N A_i|) \\
&= \log(|N\bar{A}_i|) \text{ with } \bar{A}_i = \frac{1}{N}\sum_i^N A_i \\
&= \log(N^K|\bar{A}_i|) \\
&= K\log(N) + \log(|\bar{A}_i|) \\
&\approx K\log(N),
\end{aligned} \tag{3.26}$$

since $\log(|\bar{A}_i|)$ is $\mathcal{O}(1)$ compared with $K\log(N)$ which is $\mathcal{O}(\log N)$. This expression is known as the Bayesian Information Criterion [Bishop, 2007] (BIC) which penalizes the complexity of models, that is the number of parameters K.
The simplification to drop the term $\log(|\bar{A}_i|)$ can be precarious. Difficulties can arise if the parameters of $\mathbf{\Theta}^t_{map}$ are not well-determined, for example caused by too little, insufficient measurements or strong correlations between different measurements. In this case, the precision matrix $\bar{A}_i$ can be close to singular and it is unsafe to drop it ($\log 0 = -\infty$). The effective number of parameters is smaller in this case.
Nevertheless, the made simplifications allow to reduce the complex expressions above to an especially simple form that can be evaluated efficiently. Furthermore, the BIC is insensitive to variations of the precise form of the underlying energy terms. This is particularly helpful since the precise form of the underlying energies is frequently unknown. Knowing that the precise effective number of parameters might be questionable and depends on the current observations, one attempt is to learn this parameter on the basis of ground truth material.
In summary, the energy function

$$\begin{aligned}
E = &\underbrace{-\log Q\left(\mathbf{L}^t\right) - \log Q\left(\mathbf{\Theta}^t_{map} \mid \mathbf{L}^t\right)}_{\text{prior terms}} \underbrace{-\log Q\left(\mathbf{L}^{t-1} \mid \mathbf{L}^t\right)}_{\text{temporal consistency}} \underbrace{-\log Q\left(\mathcal{Z}^t \mid \mathbf{\Theta}^t_{map},\, \mathbf{L}^t\right)}_{\text{data term}} \\
&\underbrace{-\frac{K}{2}\left(\log 2\pi - \log N\right)}_{\text{BIC}} \underbrace{-\sum_{(i,j)\in\mathcal{N}_2} \log p\left(l_i^t,\, l_j^t \mid \vec{z}_i^{\,t}, \vec{z}_j^{\,t}\right)}_{\text{smoothness term}}
\end{aligned} \tag{3.27}$$

is minimized.
The following section discusses in detail the statistical modeling of the different terms.

3.3 Definition of the Energy Terms

In this section, a possible implementation of the energy defined in Equation 3.27 is discussed. First, the modelings of the unknown moving object class and of the stationary background class are discussed. These probability density functions were learned from ground truth material from about 38000 manually labeled images. The unknown moving object class serves as an initialization for known moving objects. The unknown moving object class is further quantized into four driving behaviors: forward-moving, oncoming, left-moving and right-moving. This definition allows to separate objects that are located close together but have a different moving direction. Furthermore, the quantization step offers advantages with respect to dimension reduction and for a better object initialization. This way, the high dimensional state space $\boldsymbol{\Theta}$ is reduced to a small subset of real driving maneuvers. Furthermore, this step also helps to be able to specify a local probability of occurrence of motion states in a reduced state space (for instance, typically, oncoming objects are on the left side of the ego vehicle) that would be expensive to evaluate statistically otherwise.
Subsequently the modeling of the known moving object class is introduced. For this class, since no ground truth data was available, parametric probability models were chosen. The relevant parameters were set via cross-validation.

3.3.1 Unknown Moving Objects and Stationary Background

In contrast to the known moving objects specified in Subsection 3.3.2, for the unknown moving objects, the parameter vector $\boldsymbol{\Theta}$ is not known yet, see Equation 3.27. Hence, the modeling of the unknown moving class must be kept more general than the known moving object classes. The modeling of the unknown moving objects and of stationary background is discussed in the following.
The unary data term likelihood from Equation 3.27 is decomposed into

$$Q\left(\mathcal{Z}^t \mid \cancel{\boldsymbol{\Theta}^t_{map}}, \mathbf{L}^t\right) \overset{3.17}{:=} \prod_{i=1}^{N} p\left(\vec{z}_i^{\,t} \mid l_i^t\right), \tag{3.28}$$

where

$$\begin{aligned} p\left(\vec{z}_i^{\,t} \mid l_i^t\right) &\overset{3.4}{=} p\left(\dot{X}_i^t,\, \dot{Z}_i^t,\, X_i^t,\, Y_i^t,\, Z_i^t \mid l_i^t\right) \\ &= p\left(Y_i^t \mid l_i^t, \underset{\sim}{\dot{X}_i^t}, \underset{\sim}{\dot{Z}_i^t}, \underset{\sim}{X_i^t}, \underset{\sim}{Z_i^t}\right) \cdot p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \underset{\sim}{X_i^t},\, Z_i^t\right) \cdot p\left(X_i^t,\, Z_i^t \mid l_i^t\right) \\ &\approx \underbrace{p\left(Y_i^t \mid l_i^t\right)}_{\text{height term}} \cdot \underbrace{p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid Z_i^t,\, l_i^t\right)}_{\text{motion term}} \cdot \underbrace{p\left(X_i^t,\, Z_i^t \mid l_i^t\right)}_{\text{position term}}. \end{aligned} \tag{3.29}$$

In Equation 3.29, those dependencies were marked with a wavy line that are considered to be negligible. Such independence assumptions reduce the dimension of the underlying probability functions, making the learning step tractable even if there is only limited ground truth material available. See the following subsections for a discussion of these approximations.

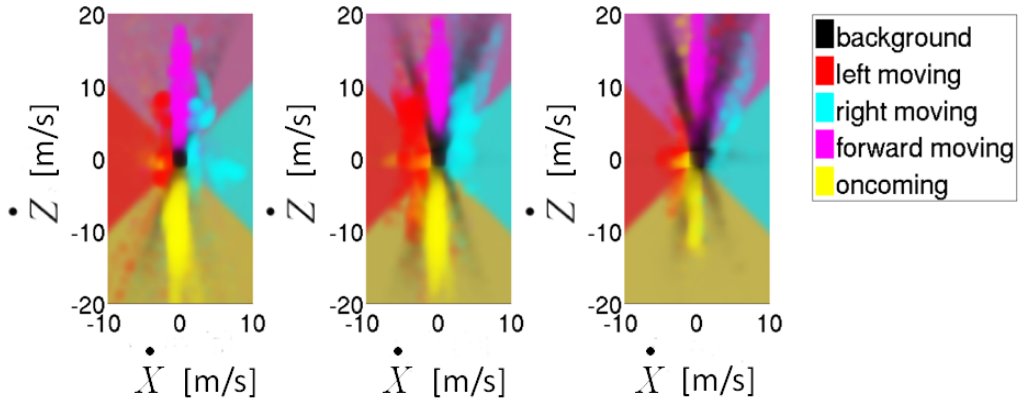


Figure 3.2: The probability $p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid l_i^t,\, Z_i^t\right)$ is color encoded for different distance ranges Z_i^t. On the left side, $Z_i^t \in \{0-20\}$ m, in the middle, $Z_i^t \in \{20-40\}$ m, and on the right side $Z_i^t \in \{40-70\}$ m is shown.

Motion probability

Clearly, the velocity measurements are the most important feature to separate moving objects from stationary background. The velocity distributions for moving objects and for static background in typical urban traffic scenes are set up from the ground truth training dataset containing manually labeled Stixels as training examples.
For the motion term, the dependency on Z_i^t is important because for a stereo camera sensor, the distance uncertainty grows quadratically with increasing distance. This theoretical scaling behavior has also been confirmed for Stixels by means of a groundtruth laser scanner, see [Pfeiffer et al., 2010]. The farther away a Stixel is, the larger is its motion uncertainty. Typically, a narrow field of view is considered in front of the ego vehicle, i.e. usually $Z_i^t \gg X_i^t$ holds. In this case, the weak dependency of $p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid l_i^t,\, X_i^t,\, Z_i^t\right)$ on X_i^t can be neglected.
The distance dimension Z is quantized to keep the learning step feasible, see Figure 3.2 for an illustration. As shown in this graph, the background motion distribution is spread more for larger distances. Hence, it becomes more difficult to separate slow moving objects from stationary background. Usually, the background distribution is modeled to be Gaussian and its variance is estimated by error propagation from estimated scene flow or optical flow confidences, cf. [Wedel et al., 2009, Lenz et al., 2011]. However, this assumption does not hold for the present setup as can be seen in Figure 3.2. Especially for larger distances, the stationary background distribution is more extended to positive Z-velocities, background is often "corunning".

Positional probability

Figure 3.3 visualizes the probability of occurrence of the unknown moving object classes and of stationary background at different world positions $\{X_i^t,\, Z_i^t\}$.
Note that in this figure, the most probable Stixel class l_i^t at different world positions is

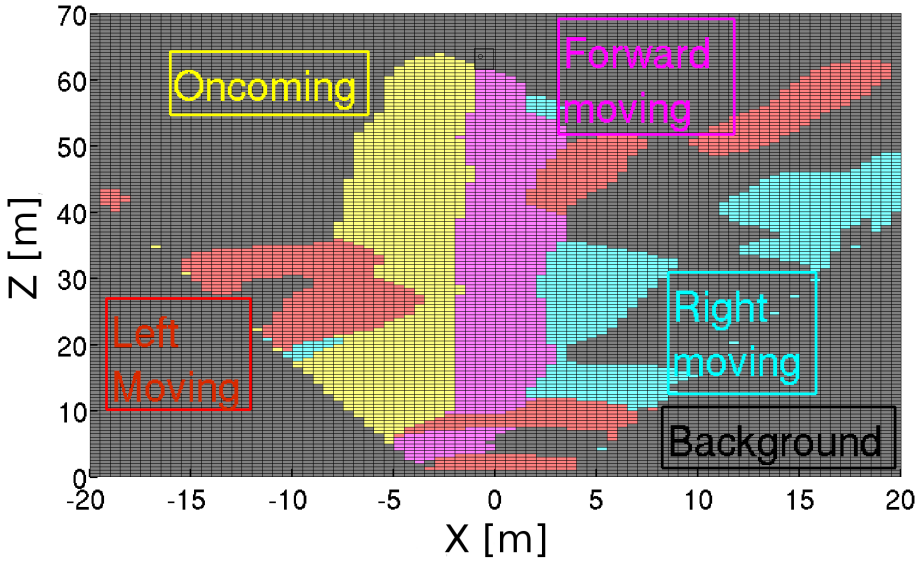


Figure 3.3: The most probable class l_i for different Stixel world positions $\{X_i, Z_i\}$ is color encoded. The colors are explained in the figure.

color-coded, i.e. $\arg\max_{l_i^t} p\left(l_i^t \mid X_i^t, Z_i^t\right)$, instead of the likelihood $p\left(X_i^t, Z_i^t \mid l_i^t\right)$. This helps to keep the visualization uncluttered. The ego vehicle is placed at the origin of the underlying (X, Z) coordinate system.
The shown positional distributions reflect various traffic related aspects: Typically, oncoming cars are located on the left-hand side of the ego vehicle at least being confronted with right-hand traffic. Stixels in front are often forward-moving due to a leading vehicle. Furthermore, Stixels close to the image border are often stationary background. This local occurrence statistics is a powerful cue for many traffic scenes.

Height probability

In Equation 3.29, the measured height Y_i^t of a Stixel is considered to be independent - given its class - of the velocity measurements $\dot{X}_i^t$, $\dot{Z}_i^t$ and independent of the position X_i^t, Z_i^t. Consider Figure 3.4 which shows the underlying height statistics. The height term favors very high Stixels to be stationary background. This is just natural, because those Stixels often model buildings or other tall infrastructure. Stixels modeling moving cars, bicycles and pedestrians have rather moderate heights typically between one and two meters. Additionally, there are some higher moving Stixels for example due to trucks or streetcars, roughly in the range between two and four meters. These two peaks can be observed in Figure 3.4 as well.

Temporal consistency term

For the temporal expectation term defined in Equation 3.27, for each Stixel at time step t a predecessor Stixel is determined by means of optical flow correspondences. In the ideal case of an error-free segmentation, the object class almost never changes. In

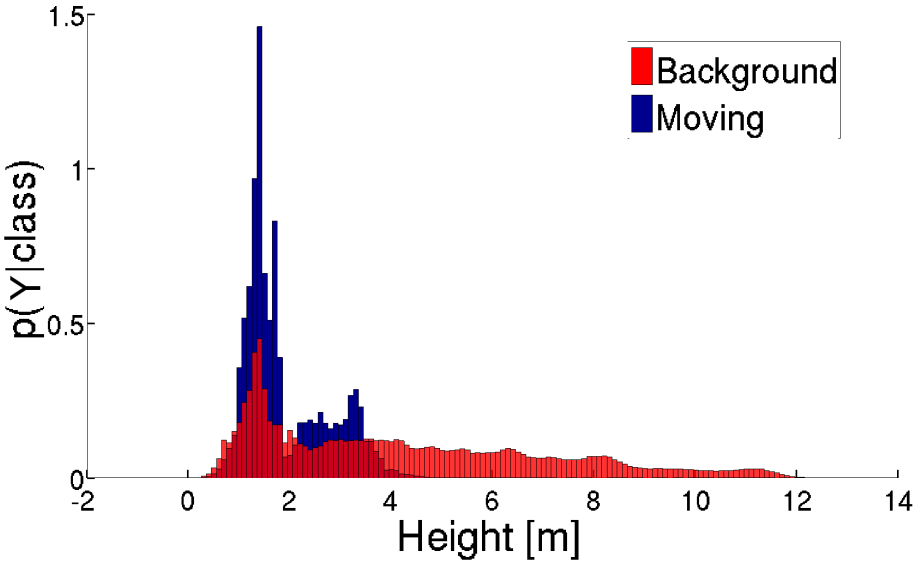


Figure 3.4: $p(Y_i \mid l_i)$ for moving objects and the static background class. The overlapping area is marked dark red.

so far, the probability for a class change in $p\left(l_i^{t-1} \mid l_i^t\right)$ should be very low. However, in practice, the old class decision could have been wrong. A very low class transition probability might freeze a wrong solution forever. This means that the actual transition probability matrix must depend on the error rate of the segmentation.
To solve this conflict, the temporal object class consistency $p\left(l_i^{t-1} \mid l_i^t\right)$ is evaluated on the basis of a training data set. Given the ground truth label l_i^t for each Stixel, the resulting class label from the previous time step l_i^{t-1} is analyzed. In most cases, the segmentation was correct. This consideration defines the transition matrix $p\left(l_i^{t-1} \mid l_i^t\right)$. The statistical findings are summarized in Table 3.1. The temporal expectation term favors a temporally consistent label decision. Nevertheless, it might also cause unwanted low-pass effects. See Subsection 4.3.3 for an extension that additionally takes into

GT / predecessor class	BG	LEFT	RIGHT	FW	ON
BG	95.57	8.39	16.73	7.91	9.16
LEFT	0.06	73.72	1.69	0.87	1.65
RIGHT	0.07	2.16	70.23	1.47	0.04
FW	3.60	1.74	10.07	89.56	0.15
ON	0.70	13.98	1.28	0.19	89.00

Table 3.1: The old class decision, l_i^{t-1} is considered to be a prior for the current segmentation l_i^t. This figure shows the statistical transition probabilities $p\left(l_i^{t-1} \mid l_i^t\right)$ in percent. BG = background, LEFT = left-moving object, RIGHT = right-moving object FW = forward-moving object and ON = oncoming object.

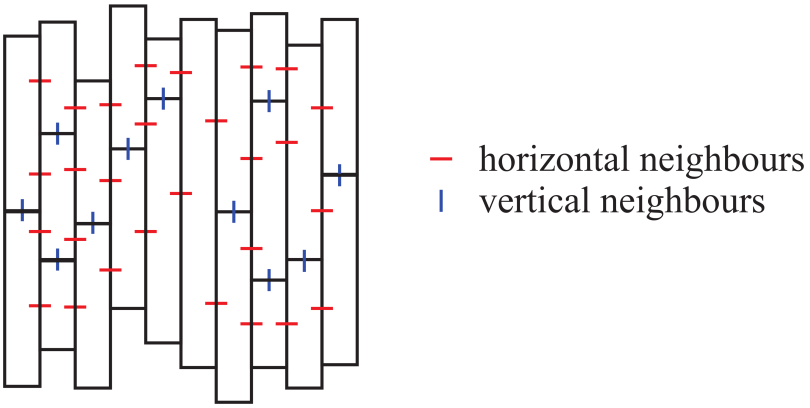


Figure 3.5: Stixel connectivity. In principle, each Stixel can have an almost arbitrary number of vertically and horizontally neighboring Stixels.

account the current observations in order to improve the temporal coupling.

Prior term

According to the training data, the prior term $p\left(l_i^t\right)$ favors the static background class. In typical urban traffic scenes, roughly 87% of all Stixels are stationary. So in general it is advisable to favor the static background class in the absence of strong data evidence which demands the opposite. The background prior and the remaining prior class probabilities are summarized in the first line of Table 3.2.

Smoothness term

The smoothness term $p\left(l_i^t, l_j^t \mid \vec{z}_i^t, \vec{z}_j^t, \mathbf{L}^{t-1}\right)$ defined in Equation 3.27 is modeled as a distance-sensitive Potts model [Ashkin and Teller, 1943, Potts, 1952], this way favoring neighboring Stixels to belong to the same class. Each Stixel is modeled to be a node in the CRF. The maximum clique size is restricted to two, and thus only nearest neighbor Stixel interactions are considered.
Since there is no regular underlying grid, each Stixel can in principle have an almost arbitrary number of neighboring Stixels, see Figure 3.5. The spatial correlation between neighboring Stixels has been investigated on the basis of a training data set, see Figure 3.6. Evidently, the correlation strongly depends on their relative depth difference. The closer two Stixels are, the more likely they belong to the same object. Since the focus is on working with stereo data, disparity deviations are considered directly rather than depth differences. In this context, the underlying disparity uncertainty σ_d is modeled to be constant. This validity of this approximation has been investigated in [Rabe, 2011, Pfeiffer et al., 2010]. Additionally, it has to be taken into account that objects (e.g. engine hoods) often do not perfectly fulfill the constant depth assumption, which is inherently assumed by the Stixel model. Objects in close proximity are therefore allowed to cover a wider disparity range than the actual disparity measuring accuracy.

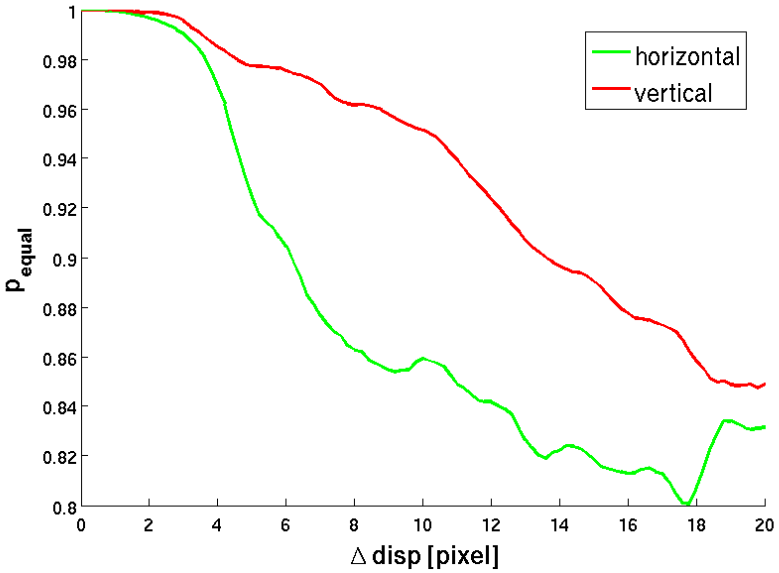


Figure 3.6: The spatial correlation between vertical (red) and horizontal (green) neighboring Stixels is plotted as a function of their mutual distances as elaborated in the text.

Consequentially, it is necessary to consider both the disparity uncertainty σ_d and a so-called *world model violation* σ_{world} in a joint metric

$$\Delta_{\text{disp}} = \frac{\|d_i - d_j\|}{\sigma_{\text{disp}}} \tag{3.30}$$

between two neighboring Stixels i and j. Hereby, σ_{disp} is defined as

$$\sigma_{\text{disp}} = \max\left(\sigma_d,\, \sigma_{\text{world}}\right). \tag{3.31}$$

Note that σ_{world} is given in pixels, it can be obtained from the expected world model violation in meters via error propagation, see 3.46. Accordingly, the smoothness term $p\left(l_i^t,\, l_j^t \mid \vec{z}_i^t, \vec{z}_j^t\right)$ is modeled as

$$p\left(l_i^t,\, l_j^t \mid \vec{z}_i^t, \vec{z}_j^t\right) = \begin{cases} -\log\left(\mathrm{p}_{\text{equal}}\left(\Delta_{\text{disp}}\right)\right), & \text{if } l_i^t = l_j^t \\ -\log\left(1 - \mathrm{p}_{\text{equal}}\left(\Delta_{\text{disp}}\right)\right), & \text{else.} \end{cases} \tag{3.32}$$

The parameters σ_d and σ_{world} have to be found via cross-validation. Note that Equation 3.32 is not properly normalized to one. However, this fact does not play a role for the resulting MAP solution.
The evaluation shown in Figure 3.6 results of the class correlation between neighboring Stixels in the training data set. In this Figure, vertical and horizontal neighboring Stixels are separated. It is shown that, typically, vertical adjacent Stixels are more

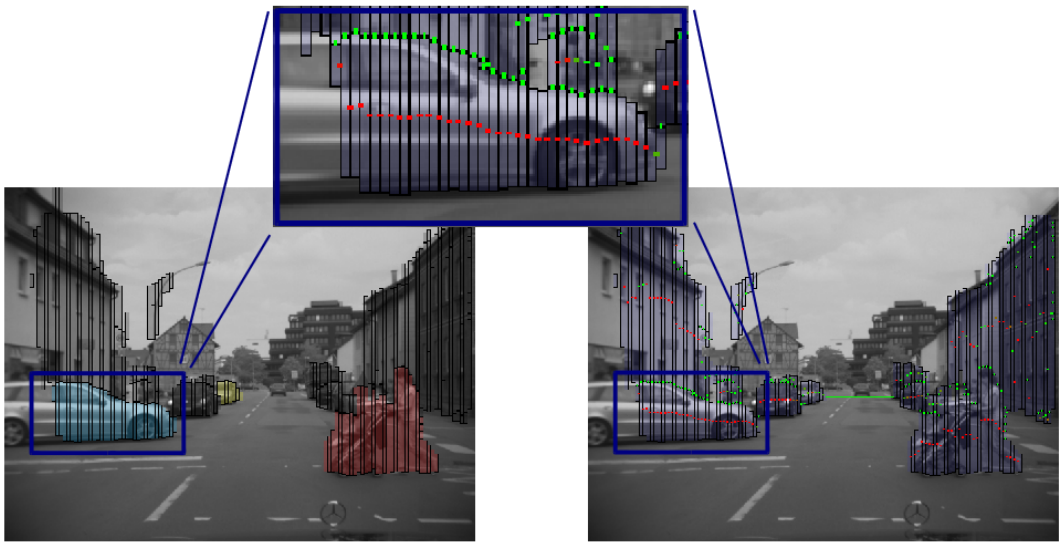


Figure 3.7: Spatial couplings between Stixels. The coupling strength is color encoded where red symbolizes strong coupling and green corresponds to a weak coupling.

correlated than horizontal neighbors which justifies the anisotropic modeling.
Figure 3.7 color-codes the resulting coupling strength from the class correlations shown in Figure 3.6 between adjacent Stixels. Red corresponds to a strong coupling and green depicts a weak coupling. Obviously, the proposed coupling approximates the shape of the objects very well.

3.3.2 Known Moving Objects

For known moving objects, the relevant probability distributions can be specified much more precisely than for the unknown moving object class or for stationary background. For these objects, prior knowledge exists: the parameter vector $\mathbf{\Theta}_{map}^{t-1}$ from the last frame specifies the relevant class parameters.

Parameter Prior

As given in Equation 3.27, the parameter prior $Q\left(\mathbf{\Theta}_{map}^{t} \mid \mathbf{L}^{t}\right)$ takes into account prior knowledge on the estimated object parameters. These parameters were already defined in Equation 3.18. In the following, the "*map*" subscript is omitted to keep the notation uncluttered. From definition 3.18

$$\begin{aligned} Q\left(\mathbf{\Theta}^{t} \mid \mathbf{L}^{t}\right) &:= Q\left(\Theta_1^t, \ldots, \Theta_M^t \mid \mathbf{L}^{t}\right) \\ &= Q\left(\Theta_1^t \mid \mathbf{L}^{t}\right) \cdot \prod_{n=2}^{M} Q\left(\Theta_n^t \mid \Theta_1^t, \ldots, \Theta_{n-1}^t, \mathbf{L}^{t}\right). \end{aligned} \tag{3.33}$$

Next, the shorthand notation

$$\Theta_{1:n}^t := \left(\Theta_1^t, ..., \Theta_n^t\right) \tag{3.34}$$

is introduced. The single object parameter prior $Q\left(\Theta_n^t \mid \Theta_{1:n-1}^t, \mathbf{L}^t\right)$ is assumed to factorize

$$\begin{aligned}
Q\left(\Theta_n^t \mid \Theta_{1:n-1}^t, \mathbf{L}^t\right) &\overset{3.18}{=} Q\left(\mathcal{X}_n^t,\, \mathcal{Z}_n^t,\, \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t,\, |\Delta\mathcal{X}|_n^t\,,\, \mathcal{H}_n^t,\, |\Delta\mathcal{Z}|_n^t \mid \Theta_{1:n-1}^t, \mathbf{L}^t\right) \\
&= Q\left(\mathcal{H}_n^t \mid \underset{\sim}{\mathcal{X}_n^t},\, \underset{\sim}{\mathcal{Z}_n^t},\, \underset{\sim}{\dot{\mathcal{X}}_n^t},\, \underset{\sim}{\dot{\mathcal{Z}}_n^t},\, \underset{\sim}{|\Delta\mathcal{X}|_n^t}\,,\, \underset{\sim}{|\Delta\mathcal{Z}|_n^t}\,, \underset{\sim}{\Theta_{1:n-1}^t}, \mathbf{L}^t\right) \cdot \\
&\quad\; Q\left(|\Delta\mathcal{X}|_n^t\,,\, |\Delta\mathcal{Z}|_n^t \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, \underset{\sim}{\mathcal{X}_n^t},\, \underset{\sim}{\mathcal{Z}_n^t}, \underset{\sim}{\Theta_{1:n-1}^t}, \mathbf{L}^t\right) \cdot \\
&\quad\; Q\left(\dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t \mid \underset{\sim}{\mathcal{X}_n^t},\, \underset{\sim}{\mathcal{Z}_n^t}, \underset{\sim}{\Theta_{1:n-1}^t},\, \mathbf{L}^t\right) \cdot \\
&\quad\; Q\left(\mathcal{X}_n^t,\, \mathcal{Z}_n^t \mid \Theta_{1:n-1}^t, \mathbf{L}^t\right) \\
&\approx \underbrace{Q\left(\mathcal{H}_n^t \mid \mathbf{L}^t\right)}_{\text{Height prior}} \cdot \\
&\quad\; \underbrace{Q\left(|\Delta\mathcal{X}|_n^t \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, \mathbf{L}^t\right)}_{\text{Width prior}} \cdot \\
&\quad\; \underbrace{Q\left(|\Delta\mathcal{Z}|_n^t \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, \mathbf{L}^t\right)}_{\text{Length prior}} \cdot \\
&\quad\; \underbrace{Q\left(\dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t \mid \mathbf{L}^t\right)}_{\text{Velocity prior}} \cdot \\
&\quad\; \underbrace{Q\left(\mathcal{X}_n^t,\, \mathcal{Z}_n^t \mid \mathcal{X}_1^t,\, \mathcal{Z}_1^t, ..., \mathcal{X}_{n-1}^t,\, \mathcal{Z}_{n-1}^t, \mathbf{L}^t\right)}_{\text{Exclusion prior}}.
\end{aligned} \tag{3.35}$$

Again, in Equation 3.35 those dependencies were marked with a wavy line that are considered to be negligible. The parameter prior is assumed to factorize into the height prior $Q\left(\mathcal{H}_n^t \mid \mathbf{L}^t\right)$, the object dimension priors $Q\left(|\Delta\mathcal{X}|_n^t \mid \dot{\mathcal{X}}_n^t, \dot{\mathcal{Z}}_n^t, \mathbf{L}^t\right)$ and $Q\left(|\Delta\mathcal{Z}|_n^t \mid \dot{\mathcal{X}}_n^t, \dot{\mathcal{Z}}_n^t, \mathbf{L}^t\right)$, the velocity prior $Q\left(\dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t \mid \mathbf{L}^t\right)$ and the object exclusion prior
$Q\left(\mathcal{X}_n^t,\, \mathcal{Z}_n^t \mid \mathcal{X}_1^t, \mathcal{Z}_1^t, ..., \mathcal{X}_{n-1}^t, \mathcal{Z}_{n-1}^t, \mathbf{L}^t\right)$.
Note that the parameter prior $Q\left(\mathbf{\Theta}^t \mid \mathbf{L}^t\right)$ still depends on the current labeling $\mathbf{L}^t$. However, it is not possible to take into account this kind of higher-order information in the graphcut optimization step, see Section 3.4. Hence, this dependency is dropped for the graphcut labeling step:

$$Q\left(\mathbf{\Theta}_{map}^t \mid \mathbf{L}^t\right) \approx Q\left(\mathbf{\Theta}_{map}^t\right). \tag{3.36}$$

This way, the graphcut labeling step becomes independent of the object parameters since $Q\left(\mathbf{\Theta}_{map}^t\right)$ becomes a constant for a fixed parameter vector $\mathbf{\Theta}_{map}^t$ that can be omitted in the graphcut step. Nevertheless, the object parameters still define the data

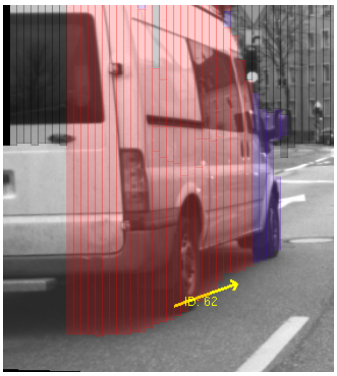


Figure 3.8: On the proposed object exclusion prior. A known, tracked object with ID 62 is shown in red. Due to noise, the front of the vehicle cannot be associated with the object. A new unknown moving object shown in purple is established. The exclusion prior prevents this since usually objects keep a certain distance.

terms as discussed in Section 3.3.2. See Chapter 4 for an optimization approach that is capable of taking into account this kind of higher-order object prior terms.
For simplicity, the height prior is assumed to be given by the learned height distribution 3.4.
The dimension priors were fixed in the experiments, i.e. a constant width and length were assumed. Alternatively, a narrow Gaussian distribution could be assumed here based on statistical knowledge. See Section 4.3.3 for an alternative.
The velocity prior is simply a uniform distribution within plausibility values,

$$Q\left(\dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t \mid \mathbf{L}^t\right) = \frac{1}{|V_x^{max} - V_x^{min}| \cdot |V_z^{max} - V_z^{min}|}. \tag{3.37}$$

Finally, for the exclusion prior it is assumed that objects keep a certain distance. Consider Figure 3.8. Due to noise, the front of the vehicle cannot be associated with the *known object* shown in red. In this case, a new unknown moving object shown in purple is established. However, both objects are very close to each other which contradicts the experience that typically objects keep a certain distance. Motivated by this result, the exclusion prior

$$Q\left(\mathcal{X}_n^t,\, \mathcal{Z}_n^t \mid \mathcal{X}_1^t,\, \mathcal{Z}_1^t, ..., \mathcal{X}_{n-1}^t,\, \mathcal{Z}_{n-1}^t, \mathbf{L}^t\right) = \begin{cases} \eta \cdot p_{in}, & \text{if } (\mathcal{X}_n^t,\, \mathcal{Z}_n^t) \in \bigcup\limits_{(k=1...n-1)} \mathcal{V}_k^{exc} \\ \eta \cdot p_{out}, & \text{otherwise} \end{cases} \tag{3.38}$$

with $p_{in} \gg p_{out}$ is defined. η is a normalization constant that ensures that the distribution is correctly normalized on the finite domain $[X_{min}, X_{max}] \otimes [Z_{min}, Z_{max}]$. $\mathcal{V}_k^{exc}$ is an exclusion volume around the position $(\mathcal{X}_k^t,\, \mathcal{Z}_k^t)$ of each moving object. In practice, an exclusion ellipsoid with principal axes in the range of a few meters is set, depending on the actual 3D uncertainty.

Data Term

The data term $Q\left(\mathcal{Z}^t \mid \mathbf{\Theta}_{map},\, \mathbf{L}^t\right)$ defined in Equation 3.27 is assumed to factorize similar to Equation 3.29

$$\begin{aligned} Q\left(\mathcal{Z}^t \mid \mathbf{\Theta}_{map},\, \mathbf{L}^t\right) &\overset{3.17}{=} \prod_{i=1}^{N} Q\left(\vec{z}_i^{\,t} \mid \mathbf{\Theta}_{map},\, \mathbf{L}^t\right) \\ &\overset{3.4}{=} \prod_{i=1}^{N} Q\left(\dot{X}_i^t,\, \dot{Z}_i^t,\, X_i^t,\, Y_i^t,\, Z_i^t \mid \mathbf{\Theta}_{map},\, \mathbf{L}^t\right), \end{aligned} \tag{3.39}$$

which is again assumed to factorize into

$$\begin{aligned} Q\left(\dot{X}_i^t,\, \dot{Z}_i^t,\, X_i^t,\, Y_i^t,\, Z_i^t \mid \mathbf{\Theta}_{map},\, \mathbf{L}^t\right) = {} & Q\left(Y_i^t \mid \underset{\sim}{\dot{X}_i^t,\, \dot{Z}_i^t,\, X_i^t,\, Z_i^t}, \mathbf{\Theta}_{map},\, \mathbf{L}^t\right) \cdot \\ & Q\left(X_i^t,\, Z_i^t \mid \underset{\sim}{\dot{X}_i^t,\, \dot{Z}_i^t},\, \mathbf{\Theta}_{map},\, \mathbf{L}^t\right) \cdot \\ & Q\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \mathbf{\Theta}_{map},\, \mathbf{L}^t\right) \\ \approx {} & Q\left(Y_i^t \mid \mathbf{\Theta}_{map},\, \mathbf{L}^t\right) \cdot \\ & Q\left(X_i^t,\, Z_i^t \mid \mathbf{\Theta}_{map},\, \mathbf{L}^t\right) \cdot \\ & Q\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \mathbf{\Theta}_{map},\, \mathbf{L}^t\right) \\ \approx {} & \underbrace{Q\left(Y_i^t \mid \mathcal{H}_n^t,\, l_i^t\right)}_{\text{height term}} \cdot \\ & \underbrace{Q\left(X_i^t,\, Z_i^t \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t,\, \mathcal{X}_n^t,\, \mathcal{Z}_n^t,\, |\Delta\mathcal{X}|_n^t\,,\, |\Delta\mathcal{Z}|_n^t\,,\, l_i^t\right)}_{\text{position term}} \cdot \\ & \underbrace{Q\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \dot{\mathcal{X}}_n,\, \dot{\mathcal{Z}}_n, l_i^t\right)}_{\text{motion term}}. \end{aligned} \tag{3.40}$$

The height term $Q\left(Y_i^t \mid \mathcal{H}_n^t,\, l_i^t\right)$ is assumed to be independent of all other observations and of all other object parameters besides the object reference height.
The positional term $Q\left(X_i^t,\, Z_i^t \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t,\, \mathcal{X}_n^t,\, \mathcal{Z}_n^t,\, |\Delta\mathcal{X}|_n^t\,,\, |\Delta\mathcal{Z}|_n^t\,,\, l_i^t\right)$ looks complicated, but it is just a bounding box prior for the observations $\vec{x}_i^{\,t} = \left(X_i^t, 0, Z_i^t\right)^T$. Finally, the motion term $Q\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \dot{\mathcal{X}}_n,\, \dot{\mathcal{Z}}_n, l_i^t\right)$ is assumed to be independent of all other observations and of all other object parameters besides the object velocity. In the following, these terms are discussed in more detail.

Motion probabilities For known moving objects, the reference velocity given by $\vec{V}_n^t := \left(\dot{\mathcal{X}}_n, 0, \dot{\mathcal{Z}}_n\right)^T$ defined in Equation 3.18 is assumed to be known from the previous time step. This reference velocity is estimated via gradient descent in Equation 3.27. The observed velocity vectors $\vec{V}_i^t := \left(\dot{X}_i^t, 0, \dot{Z}_i^t\right)^T$ are assumed to be close to the reference velocities $\dot{\mathcal{X}}_n$ and $\dot{\mathcal{Z}}_n$ apart from noise. The precise velocity distribution is modeled

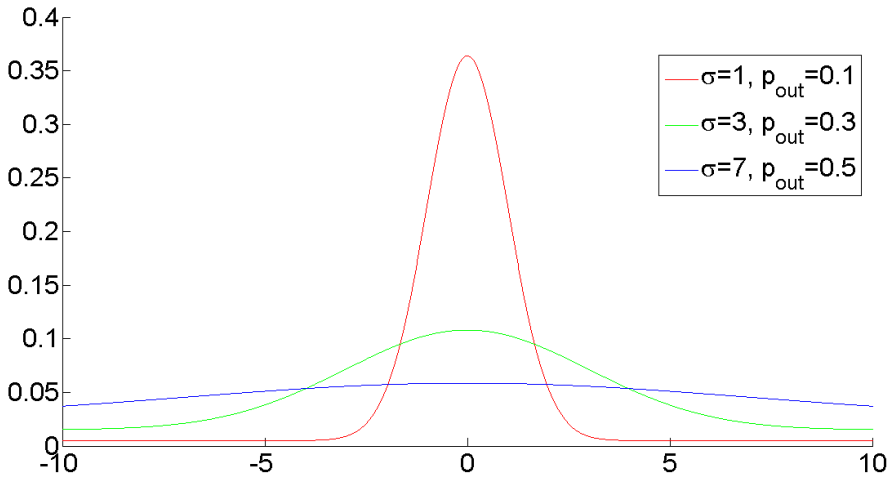


Figure 3.9: Gaussian distributions with different outlier probabilities and different standard deviations as indicated in the Figure. The distributions were normalized to the interval $[-10, 10]$.

as a weighted sum of a Gaussian distribution with a socket for outliers

$$p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \dot{\mathfrak{X}}_n,\, \dot{\mathfrak{Z}}_n\right) = (1 - p_{out}) \cdot \eta \cdot \mathcal{N}\left(\vec{V}_i^t, \vec{V}_n^t, \mathbf{\Sigma}_{i,n}^t\right) + \frac{p_{out}}{|V_x^{max} - V_x^{min}| \cdot |V_z^{max} - V_z^{min}|}, \tag{3.41}$$

where $\mathbf{\Sigma}_{i,n}^t$ denotes the covariance matrix of the velocity difference that is given by the Stixel covariance matrix, p_{out} models the frequency with which outliers occur and η is a normalization constant to ensure that the Gaussian distribution is normalized on the finite interval $[V_x^{min}, V_x^{max}] \otimes [V_z^{min}, V_z^{max}]$ (Lebesgue measure). $\otimes$ denotes the usual dyadic product.
Some example realizations are shown in Figure 3.9. The outlier distribution is assumed to be uniform over the possible range of velocity measurements. Theoretically, this range is infinite but in practice it can be limited as given above.

Positional probabilities With the object pose and dimension estimation from the previous time step, a bounding box prior for the position measurements of the n-th known moving object can be defined. In general, the old position of the n-th known moving object has to be predicted based on its estimated velocity to obtain the expected current center of the bounding box. In each time step, again, the most probable object center position can be determined via gradient descent in Equation 3.27. For the positional probability of Stixel i belonging to object n their mutual distance is decisive. A Stixel that is located far away of the object unlikely belongs to that object. On the other hand, for close Stixels the likelihood for this object class increases. This probability is

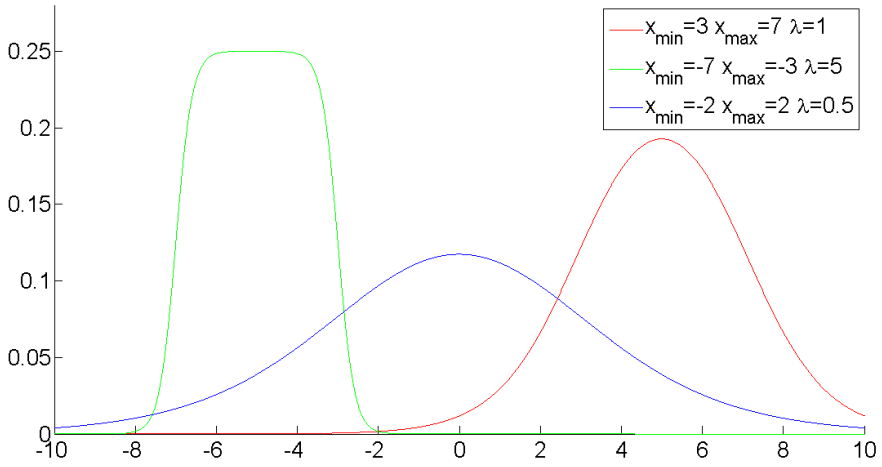


Figure 3.10: Pi-shaped probability function normalized on the interval $[-10, 10]$.

modeled as a pi-shaped probability function

$$p\left(X_i^t,\, Z_i^t \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t,\, \mathcal{X}_n^t,\, \mathcal{Z}_n^t,\, |\Delta\mathcal{X}|_n^t\,,\, |\Delta\mathcal{Z}|_n^t\,,\, l_i^t\right) = (1-p_{out}) \cdot \Pi\left({}^oX_i^t,\, |\Delta\mathcal{X}|_n^t/2, \lambda_{\mathcal{X}_n^t}\right) \cdot \Pi\left({}^oZ_i^t,\, |\Delta\mathcal{Z}|_n^t/2, \lambda_{\mathcal{Z}_n^t}\right) + \frac{p_{out}}{(X^{max}-X^{min})\cdot(Z^{max}-Z^{min})}, \tag{3.42}$$

where

$$\begin{aligned}\Pi\left({}^oX_i^t,\, |\Delta\mathcal{X}|_n^t\,, \lambda_{\mathcal{X}_n^t}\right) &= \frac{\eta}{1+\exp\left(-\lambda_{\mathcal{X}_n^t}\cdot\left({}^oX_i^t + |\Delta\mathcal{X}|_n^t\right)\right)} - \\ &\quad \frac{\eta}{1+\exp\left(-\lambda_{\mathcal{X}_n^t}\cdot\left({}^oX_i^t - |\Delta\mathcal{X}|_n^t\right)\right)}, \\ \Pi\left({}^oZ_i^t,\, |\Delta\mathcal{Z}|_n^t\,, \lambda_{\mathcal{Z}_n^t}\right) &= \frac{\eta}{1+\exp\left(-\lambda_{\mathcal{Z}_n^t}\cdot\left({}^oZ_i^t + |\Delta\mathcal{Z}|_n^t\right)\right)} - \\ &\quad \frac{\eta}{1+\exp\left(-\lambda_{\mathcal{Z}_n^t}\cdot\left({}^oZ_i^t - |\Delta\mathcal{Z}|_n^t\right)\right)}.\end{aligned} \tag{3.43}$$

Again, η is a normalization constant to ensure that the probability functions are correctly normalized on their respective domain $[X^{min}, X^{max}]$ and $[Z^{min}, Z^{max}]$. ${}^oX_i^t$ and ${}^oZ_i^t$ are the Stixel positions in the object coordinate system, that is

$$\begin{pmatrix} {}^oX_i^t \\ {}^oZ_i^t \end{pmatrix} = \begin{pmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{pmatrix} \cdot \begin{pmatrix} X_i^t - \mathcal{X}_n^t \\ Z_i^t - \mathcal{Z}_n^t \end{pmatrix}, \tag{3.44}$$

where

$$\psi := \operatorname{atan2}\left(\dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t\right). \tag{3.45}$$

These equations hold in a left-handed coordinate system where positive rotations are clockwise. The object orientation direction defined by its yaw angle ψ is assumed to be given by its motion direction $\dot{\mathcal{X}}_n^t$ and $\dot{\mathcal{Z}}_n^t$. $\lambda_{\mathcal{X}_n^t}$ and $\lambda_{\mathcal{Z}_n^t}$ model the (inverse) positional uncertainty at that position. They can be computed via error propagation from the projection equations 4.41

$$\begin{aligned}
\left|\lambda_{\mathcal{X}_n^t}\right| &= \left[\left(\frac{\partial \mathcal{X}_n^t}{\partial u}\cdot\sigma_u\right)^2 + \left(\frac{\partial \mathcal{X}_n^t}{\partial d}\cdot\sigma_d\right)^2\right]^{-1/2} \\
&= \left[\frac{\left(\mathcal{Z}_n^t\right)^2}{b^2\cdot f_x^2}\cdot\left(b^2\cdot\sigma_u^2 + \left(\mathcal{X}_n^t\right)^2\cdot\sigma_d^2\right)\right]^{-1/2}, \\
\left|\lambda_{\mathcal{Z}_n^t}\right| &= \left|\left(\frac{\partial \mathcal{Z}_n^t}{\partial d}\cdot\sigma_d\right)\right|^{-1} \\
&= \left[\frac{\left(\mathcal{Z}_n^t\right)^2}{b\cdot f_x}\cdot\sigma_d\right]^{-1}.
\end{aligned} \tag{3.46}$$

In this equation, σ_u and σ_d denote the respective image and disparity uncertainties that are assumed to be constant, see [Pfeiffer et al., 2010, Rabe, 2011].

Height probabilities Similarly, the object reference height $\mathcal{H}_n^t$ is given by the mean Stixel height of all Stixels associated with the n-th known moving object. All Stixels are supposed to have a similar height Y_i^t to this object reference height $\mathcal{H}_n^t$. This relation is again expressed by a Gaussian distribution

$$p\left(Y_i^t \mid \mathcal{H}_n^t, l_i^t\right) = (1-p_{out})\cdot\eta\cdot\mathcal{N}\left(Y_i^t, \mathcal{H}_n^t, \sigma_H\right) + \frac{p_{out}}{|H^{max}-H^{min}|}, \tag{3.47}$$

with the same parameter names as introduced above.
Next, the optimization of Equation 3.27 is discussed.

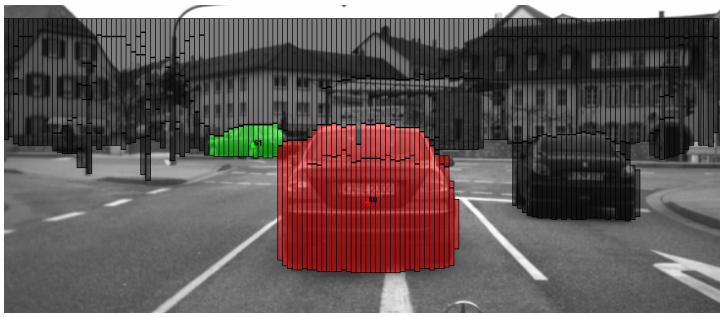
3.4 Inference

It is computationally infeasible to optimize Equation 3.27 directly because $\mathbf{\Theta}^t$ and $\mathbf{L}^t$ depend on each other. For that reason, an alternating two-stage optimization technique is applied to find the MAP solution $\mathbf{L}^*$. Note that this is just a local optimum in general. A good initialization is a prerequisite to achieve good results. For a fixed parameter vector $\mathbf{\Theta}_{map}^t$, the optimal labeling that minimizes equation 3.27 can be found using the multi-class α-expansion graphcut scheme [Boykov et al., 2001, Delong et al., 2012]. The estimation of the most probable object parameters $\mathbf{\Theta}_{map}^t$ for a fixed labeling $\mathbf{L}^*$ is discussed below. Object segmentation is a dynamic interplay between these two steps.
As discussed in Section 3.3.2, for a fixed parameter vector $\mathbf{\Theta}_{map}^t$ - taking into account the approximation 3.36 - the parameter prior term $Q\left(\mathbf{\Theta}_{map}^t \mid \mathbf{L}^t\right)$ becomes independent of the current labeling $\mathbf{L}^t$ and can be omitted in the graphcut step. See Chapter 4 for a more global optimization strategy that can take into account this kind of higher-order

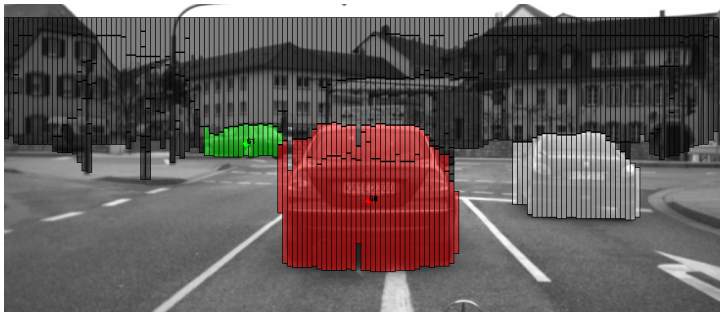
object prior terms.
In principle, the labeling step can be solved with any other optimization method such as loopy belief propagation [Yu et al., 2007, Sun et al., 2003, Tappen and Freeman, 2003, Felzenszwalb and Huttenlocher, 2006, Meltzer et al., 2005, Yang et al., 2010] or linear programming relaxations [Rother et al., 2007], too. However, the α-expansion solver is very efficient and it meets certain optimality properties as introduced in Subsection 2.6.2. For these reasons it was used in the experiments. The necessary condition 2.61 for the α-expansion algorithm to be applicable clearly holds for the Potts model shown in Figure 3.6.
The optimization of the higher-order BIC term using the α-expansion graphcut has been described already in Subsection 2.6.2.
The complexity of the α-expansion algorithm is linear in the number of classes $\mathcal{O}(J)$, see Subsection 2.6.2. A single binary graphcut step takes $\mathcal{O}(N^2 \cdot \mathcal{E} \cdot |C^*|)$, where $\mathcal{E}$ denotes the number of edges and $|C^*|$ is the cost of the minimum cut, see Subsection 2.6.2. So the highest saving potential is offered by reducing the number of nodes N in the graph. The Stixel World exactly addresses this issue. Instead of using about five hundred thousand individual pixels, an input image is described by about three hundred Stixels. The segmentation algorithm presented in this chapter has been implemented in C++ and takes about 1 ms on a single CPU core for five classes.
For the parameter estimation step, two approaches were followed in this work: the first approach tries to estimate the hidden object parameters in an iterative manner over time, the second approach uses a Radar sensor to be the source for this parameter vector. See the following Sections 3.4.1 and 3.4.2 for details.

3.4.1 Vision-based iterative parameter optimization

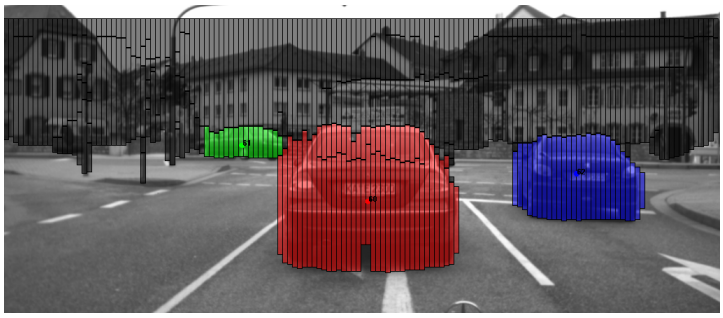
The vision-based optimization approach alternates in an Expectation-Maximization-like manner between segmentation cycles using the α-expansion graphcut scheme [Boykov et al., 2001, Delong et al., 2012] to find the most probable segmentation for fixed object parameters $\mathbf{\Theta}^{t-1}$ and it re-estimates the object parameters $\mathbf{\Theta}^t$ for a fixed segmentation $\mathbf{L}^t$.
In contrast to classical EM which requires user input for initialization and maximizes the data likelihood $p(\mathcal{Z}^t \mid \mathbf{\Theta}^t)$ directly, the proposed approach uses the probabilistic formulation presented in Subsection 3.2 without any initialization to reason about the actual number of objects. Furthermore, instead of iterating until convergence for a single image as done in [Rother et al., 2004], the optimization is performed over several images. This way, the approach exploits the strong correlations between neighboring images of the same image sequence, it is very stable and it is considerably faster. See Algorithm 3 for a detailed description and Figure 3.11 for visualization.



(a) First image. Two moving objects are already detected, the car on the right side is starting to drive.



(b) Second image. The moving object on the right side is detected as an unknown moving object, shown in white. Subsequently, its object parameter vector will be estimated as described in the main text.



(c) Third image. The unknown moving object is a known object now with its parameter vector Θ_3.

Figure 3.11: Visualization of the optimization algorithm on the basis of three successive images.

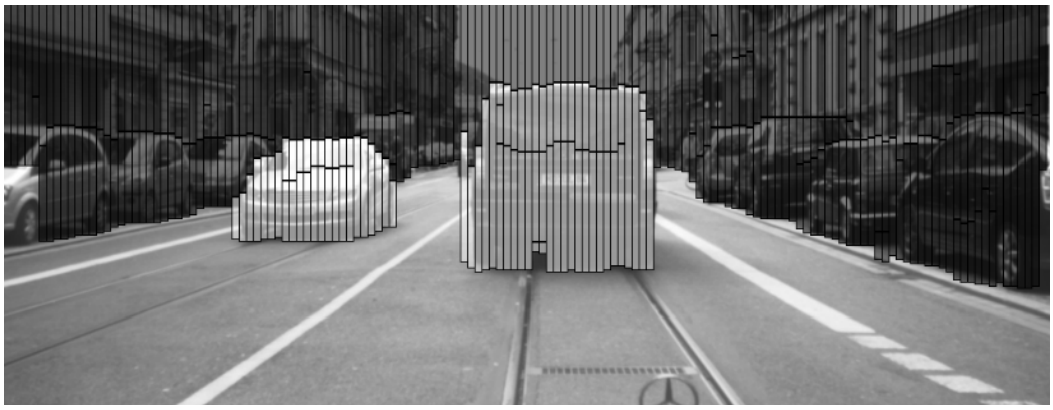


Figure 3.12: Two moving objects are detected simultaneously. Both are considered as an unknown moving object shown in white. Model selection is necessary to determine the actual number of moving objects in one frame.

Data : Dynamic Stixel World
Result : Stixel Object class segmentation $\mathbf{L}^t$ and object parameter estimation $\mathbf{\Theta}^t$
Initialize $\mathbf{\Theta}^0 = \{\}$, $E(\mathbf{L}^0) = +\text{Inf}$, $t = 1$;
1 Compute MAP solution $\mathbf{L}^t$ using α- expansion graphcut for fixed $\mathbf{\Theta}^{t-1}$;
2 Extract *unknown moving objects*;
3 Re-estimate object parameters $\mathbf{\Theta}^t$ by sampling and gradient-descent in Equation 3.27;

Algorithm 3 : Alternating, iterative labeling and parameter estimation strategy.

In order to initialize the parameters of the moving objects, a distinction is made between *known objects*, which have already been observed before and that have an already existing parameter vector $\mathbf{\Theta}^{t-1}$, and *unknown moving objects*. Unknown moving objects (UMOs) have not been observed so far and will change their status to a known object in the next frame $t + 1$ of the segmentation.
The remaining difficulty is to initially find the set of parameters $\mathbf{\Theta}^t$ that minimize Equation 3.27.
Consider Figure 3.12. In one frame, two moving objects are detected simultaneously. However, the algorithm just recognizes one large unknown moving object shown in white. In order to determine the actual number of objects, Equation 3.27 needs to be minimized with respect to $\mathbf{\Theta}^t$. A strategy of one-dimensional sampling and multidimensional gradient descent is applied to find these parameters. The approach proceeds as follows:
Initially, a zero object hypothesis is tested. In this case, the unknown moving Stixels were generated by noise, they are not statistically significant. The costs for this case can be computed by the costs of the background hypothesis for these Stixels, see Subsection 3.3.1.
Next, a one object hypothesis is tested. In this case, one moving object has generated the unknown moving Stixels. Various values for the object parameters Θ_1^t are sampled.

As stated above, the dimension variables $|\Delta\mathcal{X}|_1^t$ and $|\Delta\mathcal{Z}|_1^t$ were fixed in the experiments. A close-by solution would perhaps densely discretize the values for the object center $\mathcal{X}_1^t$ and $\mathcal{Z}_1^t$, the values for the object velocities $\dot{\mathcal{X}}_1^t$ and $\dot{\mathcal{Z}}_1^t$ and the values for the object height $\mathcal{H}_1^t$ and would test all combinations of them. In this work, however, it is assumed that one Stixel of the unknown moving Stixels defines all object parameters *simultaneously*. This way, far less hypotheses have to be tested. Each Stixel i that was labeled as an unknown moving object,

$$\mathbf{P}_{UMO} = \left\{ i \in 1...N : l_i^t \in \text{UMO} \right\}, \tag{3.48}$$

defines the *full* object hypothesis Θ_1^t, that is $\mathcal{H}_1^t = Y_i^t$, $\mathcal{X}_1^t = X_i^t$, $\mathcal{Z}_1^t = Z_i^t$, $\dot{\mathcal{X}}_1^t = \dot{X}_i^t$ and $\dot{\mathcal{Z}}_1^t = \dot{Z}_i^t$. This means that the object height/center/velocity can only be one of the heights/positions/velocities of the Stixels that were classified as an unknown moving object. From these typically 10 hypotheses, Θ_1^t is initially set to the Stixel hypothesis which minimizes Equation 3.27

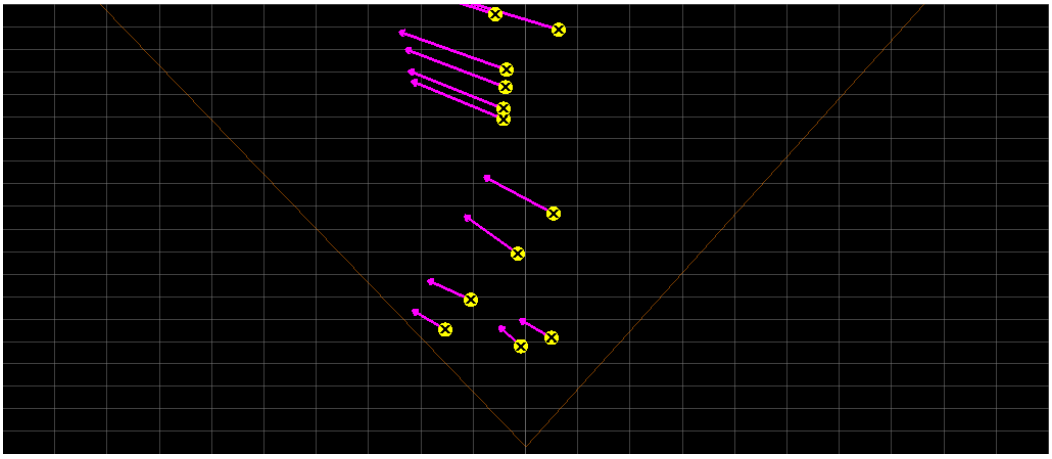
$$\Theta_{1,map}^t := \arg\min_{\Theta_n} E\left(\Theta_n\right), n \in \mathbf{P}_{UMO}, \tag{3.49}$$

where $E\left(\Theta_n\right)$ is given by Equation 3.27. $\Theta_{1,map}^t$ is refined via gradient descent in Equation 3.27 and fixed.
Next, the two objects hypothesis is tested. In this case, two moving objects have generated the unknown moving Stixels. This case is also shown in the Figure 3.12. For a fixed value of $\Theta_{1,map}^t$, the best value for $\Theta_{2,map}^t$ minimizing Equation 3.27 is searched. The possible hypotheses Θ_n are the same as before. Afterwards, both object parameters vectors $\Theta_{1,map}^t$ and $\Theta_{2,map}^t$ are refined via gradient descent simultaneously.
Next, the three object case is tested and so on.
This procedure is stopped until Equation 3.27 stops decreasing. Equation 3.27 takes into account model complexity. Both the prior $-\log Q\left(\boldsymbol{\Theta}_{map}^t \mid \mathbf{L}^t\right)$ and the BIC increase for more complex models and there is an optimal trade-off between model complexity and data agreement.
The approach greedily selects the first object that most decreases Equation 3.27 which typically corresponds to the dominating or largest object in the scene. In practice, robust terms in Equation 3.27 are required to find the true object parameters.
Since the algorithm optimizes the M objects case knowing the most probable $M-1$ objects case it can take into account object interactions such as the exclusion prior 3.38.

3.4.2 Radar-based parameter optimization

Alternatively, the initial object states $\boldsymbol{\Theta}_{map}^t$ can be obtained by an additional Radar sensor.
A Radar sensor is very well suited for detecting parallel traffic, because it can directly measure such movement via Doppler Shift. However, the lateral resolution is limited in comparison to a camera system and hence the accuracy with which crossing traffic can be observed. Thus it is beneficial to combine both sensors.
The Radar sensor used here (Continental ARS300 long range RADAR [Continental Automotive Industrial Sensors, 2011]) provides a large amount of object hypotheses, cf.

(a) Radar object hypotheses. The Radar object hypotheses marked by flipped, yellow T symbols can define the full parameter vector $\boldsymbol{\Theta}_{map}$.



(b) Radar object hypotheses shown in birds-eye view relative to the ego-vehicle. The grid size is 5 m, the magenta arrows show the predicted position within the next half second. The velocity arrows seem to be tilted due to the aspect ratio of this figure.

Figure 3.13: Radar object hypotheses to initialize $\boldsymbol{\Theta}_{map}$ in Equation 3.27.

Figure 3.14: Picture of the used stereo camera system mounted behind the windshield of the experimental vehicle. A detailed specification of the cameras is given in the main text.

Figure 3.13. The energy proposed in equation 3.27 takes into account the Bayesian Information Criterion (BIC) which penalizes model complexity which refers to the number K of parameters in the model. This number is proportional to the number of objects M in the scene. Taking this term into account, it is possible to find the true number of objects in the scene.
This approach performs sensor fusion at an early stage, in contrast for example to [Munz and Dietmayer, 2011]. However, sensor fusion in general is not in the scope of the present work. The next Section 3.5 presents experimental results.

3.5 Results

In the experiments, a stereo camera system from Bosch mounted behind the windshield of the experimental vehicle is used, see Figure 3.14. The height of the camera system is about 1.17 m with a base line of 22 cm as well as an image resolution of 1024×440 pixels. The camera system records gray value images at 25 Hz with 12 bits per pixel. The field of view is about 42° and the focal lengths $f_x, f_y \approx 1250$ pixels. The dense stereo depth maps are computed in real-time at 25 Hz on a dedicated FPGA platform using the Semi-Global Matching algorithm as described in [Hirschmuller, 2005, Gehrig et al., 2009]. The optical flow correspondences for the Stixel tracking are obtained from the well-known Kanade-Lucas-Tomasi (KLT) [Shi and Tomasi, 1994] tracker. In order

Features	BG	LEFT	RIGHT	FW	ON	**Average**	**Global**
prior (GT)	87.56	0.73	0.73	7.49	3.49	-	-
All	99.54	85.07	95.65	79.77	83.18	88.64	98.01
w/o motion	92.26	0.00	0.00	48.04	56.11	39.28	85.40
w/o height	99.69	83.68	94.91	77.40	80.63	87.26	97.95
w/o prior	88.69	93.01	97.27	89.01	88.86	91.37	88.99
w/o position	99.83	92.70	95.80	75.73	66.42	86.10	97.98
w/o binary	98.14	80.25	89.72	75.55	80.91	84.91	96.31
w/o temporal	99.70	84.30	95.17	77.96	83.08	88.04	98.06

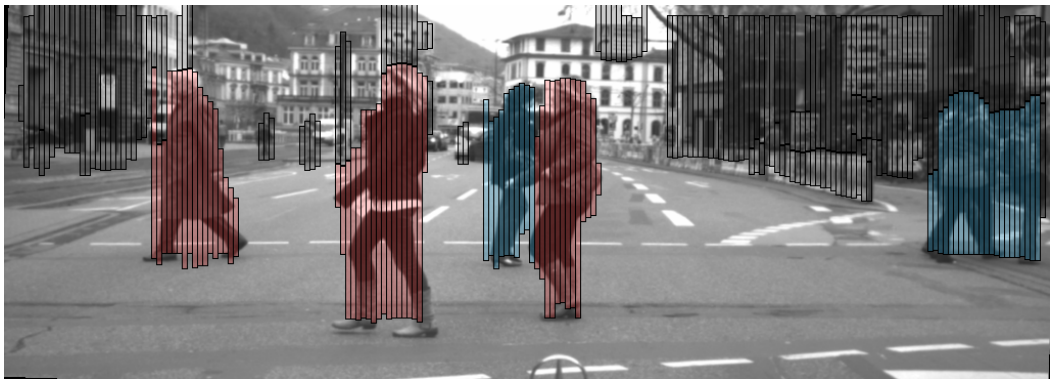
Table 3.2: Stixel-wise percentage accuracy for the evaluation sequences. BG = background, LEFT = left-moving object, RIGHT = right-moving object, FW = forward-moving object and ON = oncoming object. "Global" denotes the percentage of Stixels that were correctly classified, "Average" is the average of the per-class accuracies.

to determine the required ego motion estimation, speed and yaw rate are extracted from the inertial sensors of the experimental vehicle.
In the following Subsection 3.5.1, the detection rates of the unknown moving objects and of stationary background are evaluated. This concept allows to initialize the known moving objects as described in Section 3.3 and 3.4. The subsequent full segmentation cycle taking into account the known moving objects is evaluated in Subsection 3.5.2.

3.5.1 Unknown Moving Objects and Background

To test the performance of the unknown moving object initialization, the segmentation results of these classes were compared with a manually labeled ground truth data set, containing about 8000 images recorded from the experimental vehicle. All experiments have been performed with a single parameter set, and thus without any manual parameter tuning.
The performance of the system is summarized in Table 3.2. The overall labeling accuracy is about 98%. Besides that, distinct features are omitted in order to test their influence on the final segmentation result.
As it turns out, the best overall performance is achieved taking into account all the proposed features from Section 3.3 and leaving out the temporal term. In this case, the average labeling accuracy is 98.06%. However, when taking account the temporal coherence constraint, the results are quite similar with 98.01%. The difference amounts to about 100 Stixels. The positive influence of the temporal constraint is canceled by unwanted low-pass effects. Partially, the Stixel Kalman Filters need multiple frames to converge to their final velocity. All Stixels are initialized with zero velocity. In this case, the temporal coupling extends a possible wrong labeling decision. Besides that, the Stixel tracking is quite stable, thus the benefit of a temporal smoothing is small. Anyway, a temporally consistent labeling decision is desirable for many applications.
The motion cue turns out to be the most discriminative feature as expected. By ignoring this term, the global performance decreases to 85.40%, as shown in Table 3.2. In this case, some maneuvering classes, such as right-moving, are not classified cor-

(a) Example result from the training data set.



(b) Example scene from the evaluation sequence.

Figure 3.15: The used training (left) and evaluation (right) sequences.

rectly any longer at all. Still, as one can see from the results, the other features also play an important role for the global segmentation result. The prior term, for example, proves to be important because it suppresses phantom objects, which are wrong moving objects as a result of motion artifacts. By leaving this term out, the overall labeling accuracy drops to 88.99%, even though the average per-class accuracy increases because the dominating static background class is not favored any longer.
The position term turns out to be advantageous especially for oncoming objects and objects driving ahead. In this case when taking into account this cue the segmentation accuracy rises significantly for these classes.
The detection of the unknown moving objects serves as input for the subsequent tracking step that is evaluated in the following subsection.

3.5.2 Known Moving Objects

In order to evaluate the performance of the full segmentation approach, the segmentation results were compared with a different manually labeled ground truth data set. This data set contains another about 80 000 images, the complete data from a test drive with a length of about one hour. The test data includes urban areas, rural roads and short highway parts. Every 80th image has been manually labeled to provide ground truth material as a representative sample. In this ground truth database, there are several (Stixel-wise) labeled moving objects in addition to labeled stationary background. The experimental results are summarized in Figure 3.16 and Figure 3.17. There, the x-axis specifies the required minimum overlap: objects are considered to be segmented correctly if they overlap more than $x\%$ with a labeled object in the image plane. For example, the PASCAL criterion suggests an overlap of 50% [Wojek et al., 2010, Everingham et al., 2010], [Gehrig et al., 2012] require an overlap of 60%. Besides that, the figures differentiate between various distances. Figure 3.16 shows the detection rate of moving objects for the vision-only solution and in Figure 3.17 for the Radar-assisted approach. Adding the Radar information increases the detection rate by about ten to fifteen percent in comparison to the vision-only solution. Especially for large distances, it is extremely difficult to separate oncoming cars from stationary background, based on their motion. The accuracy for measuring parallel traffic and hence the sensitivity (for a constant false positive rate) of the Radar sensor for parallel traffic is significantly higher as discussed in Subsection 3.4.2. Accordingly, the performance of the segmentation can be improved taking into account the Radar information.
For a better grading of the results and to discuss some of the remaining error cases, see Figure 3.18(a) and Figure 3.18(b). In Figure 3.18(b), a pedestrian walking slowly in front of a wall is not detected but such slowly moving pedestrians appear in the ground truth database. Usually, the measurement motion noise is higher than the pedestrian movement, so the pedestrian cannot be reliably detected. In future work, the intention could be to take into account a pedestrian classification step in order to increase the sensitivity of the system.
If requesting for a very high overlap ($\geq 90\%$), the detection rate drops significantly. This decrease is comprehensible and mostly corresponds to - depending on the distance - one or two Stixels at the border of objects due to fluctuations in the input data and due to an increased uncertainty at object borders, see Subsection 3.6.

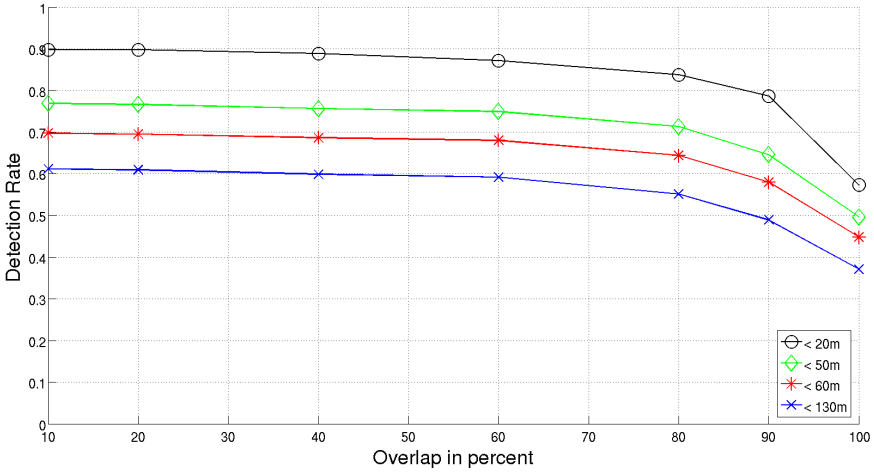


Figure 3.16: Moving object detection rate based on the vision-only solution. The x-axis specifies the minimum required overlap of the segmentation result with a ground truth object. A distinction is made between different distance ranges.

approach	correct background
with Radar	99.18 %
without Radar	99.64 %

Table 3.3: The correct labeled stationary background Stixel percentage.

Complementary to this investigation, the correctly labeled stationary background (false positives) has been examined. The statistical findings are summarized in Table 3.3. The low phantom rate observed in the experiments is a direct consequence of the strong regularization applied in this approach. See Figure 3.18(a) for an example of a remaining false positive detection. For the vision-only solution, the phantom rate is roughly one phantom Stixel every twentieth image. Using the Radar-assisted approach, the phantom rate is slightly higher, it is about one phantom Stixel every six images. There are many false positive measurements, especially due to erroneous Radar reflections at crash barriers, cf. Figure 3.18(a). However, vision-based motion segmentation is difficult for crash barriers, too. Due to weak texture and periodicities, optical flow estimation often fails. See [Pfeiffer, 2012,Schneider et al., 2012] for discussions on crash barrier tracking.

3.6 Conclusion

An EM-like CRF model for traffic scene segmentation has been presented. The difficulty of an (theoretically) uncountable infinite number of object classes is solved in

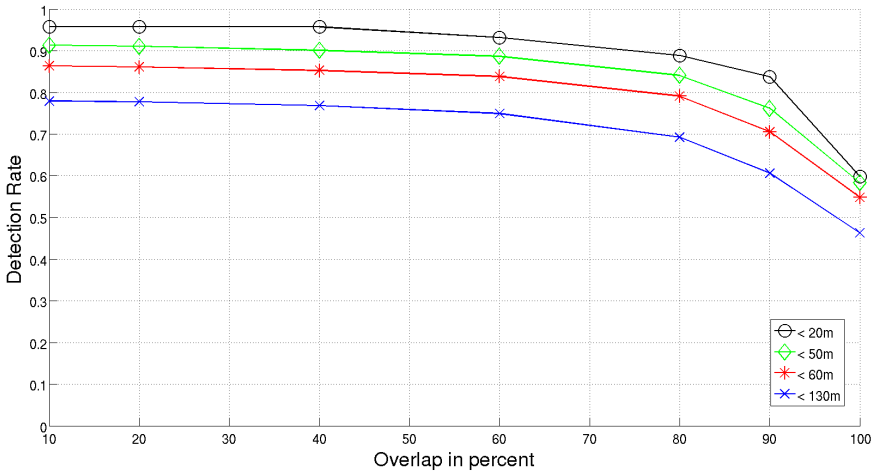
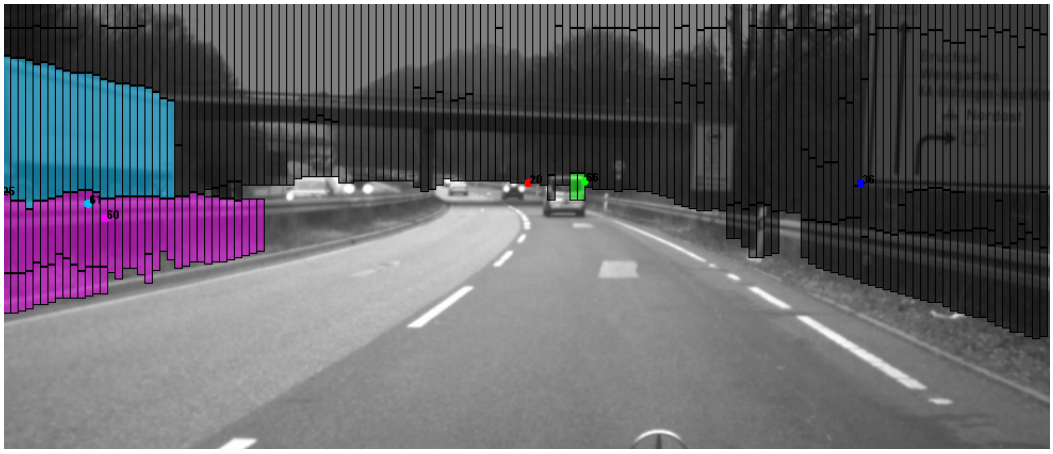


Figure 3.17: Moving object detection rate based on the Radar-assisted solution. The x-axis specifies the minimum required overlap of the segmentation result with a ground truth object. A distinction is made between different distance ranges.

a time-recursive fashion. The effectiveness of the proposed method has been demonstrated on the basis of ground truth data in various, challenging traffic scenes. The presented real-time capable approach has been extensively tested in the experimental vehicle.
This work focuses primarily on urban traffic scenes. However, the approach can easily be adapted to other scenarios such as highways or rural roads. In this case, scenario specific knowledge like the positional occurrence statistics shown in Figure 3.3 or the sensor model have to be adapted.
Using the Stixel World instead of dense stereo and pixel-wise motion information yields significant improvements with respect to stability and real-time capability because the amount of input data is reduced considerably and the Stixel World is largely insensitive to noise. Errors due to a wrong Stixel segmentation were found to be very seldom. The assumption of a planar street surface turns out to be one of the most significant difficulties which also limits the maximum object distance that can be resolved. More detailed object and street models are required to increase the range of the Stixel World.

There are ways to further develop this approach towards an increasingly powerful vision system. One intention is to take into account appearance cues, e.g. pedestrian classification. This step will help to further increase the sensitivity of the system especially for slowly moving pedestrians.
Besides that, incorporating further scenario-specific knowledge like lane markings or from externally provided maps has the potential to yield significant improvements. Perhaps these maps could be generated by multiple passes of the same route.
Thirdly, it might be beneficial to integrate higher-order information from a scene clas-



(a) Phantom example. A crash barrier is incorrectly segmented as a moving object due to erroneous Radar reflections and weak texture that complicates the vision-based tracking.

(b) False negative example. A slowly moving pedestrian is not separated from the stationary background.

Figure 3.18: Error cases to visualize the discussion in the main text.

sification step that classifies traffic scenarios similar to [Ess et al., 2009, Geiger et al., 2011, Heracles et al., 2010] or to couple both approaches.

A different approach based on dynamic programming is discussed in the next Chapter 4. Improvements there aim towards a global optimization of Equation 3.27 and enable to take into account higher-order object information which is undisputedly intractable using the present graphcut-based optimization scheme. The alternating optimization proposed in this chapter can get stuck in local optima since the object parameters $\mathbf{\Theta}^t_{map}$ and the labeling $\mathbf{L}^t$ cannot change simultaneously. Higher-order object properties such as object dimensions are difficult to handle with the present approach. The dynamic programming-based approach can additionally take into account such higher-order properties in a global optimization and thus yields superior results.
Further remaining difficult scenarios include object occlusions, especially for high degrees of occlusions. Object occlusions will be a central feature of the approach presented in the next Chapter 4.
Furthermore, Stixels at object borders often cause problems, cf. Figure 3.16 and 3.17 for high degrees of overlap. In extreme cases, these error-prone border Stixels may cause emergency braking maneuvers. Insofar, a successful object segmentation takes into account this increased uncertainty by means of prior sensor and object knowledge, see Chapter 4
Finally, the question arises if it is possible to increase the reach of the segmentation when taking into account additional object knowledge in comparison with this rather local MRF approach, cf. equation 3.36 and see also Figures 3.16 and 3.17. These and other questions will be part of the following chapter.

4 Dynamic Programming-based Object Segmentation

4.1 Introduction

The previous Chapter 3 has introduced a Conditional Random Field-based approach for moving object traffic scene segmentation. However, the kind of object information that can be taken into account global optimally is mostly limited to local pairwise labeling decisions. The problem is that the information that can be extracted from such local image patches is limited [Wojek and Schiele, 2008]. Furthermore, there are little control possibilities for the final segmentation since large areas of the segmentation cannot be constrained efficiently.
The difficulty of taking into account higher-order image information is that such long range interactions in general increase complexity exponentially, see Subsection 2.6.2 and [Ishikawa, 2009, Rother et al., 2009] and they are far less general than low-level properties. This is a major problem not only for reasons of computing time, but also with respect to modeling since in general, an adequate and individual probability measure needs to be specified or learned for an exponentially rising number of label configurations.
Nevertheless, it is incontestable that such higher-order regularization offers high potential when striving for powerful segmentations. Especially for noisy data and under adverse weather conditions such regularization can yield superior results. Object properties that require a higher-order description for example include object dimensions, object shape information or spatially varying appearance information.
In this chapter, an approach for object segmentation based on dynamic programming is introduced. Initially, the approach is limited to the first Stixel row, see Figure 4.1 for an impression. In most cases, this is not a severe restriction since typically the first Stixel row addresses the closest and most relevant obstacles. Furthermore, approximations are necessary since the underlying segmentation task is NP-complete but needs to be computable within a few milliseconds on standard hardware. For these reasons, it is important to know which simplifications are permissible to still solve the posed problem as efficiently as possible. Nevertheless, the proposed approach has great potential for possible extensions. Hence, at the end of this chapter, a possible extension to multiple Stixel rows is proposed in Subsection 4.6. This extension allows to take into account some occlusion scenarios which cannot be addressed otherwise, see Figure 4.22 for a first impression.
Both approaches based on dynamic programming are one-dimensional optimization approaches. However, they offer several advantages over the more general two-dimensional case. Firstly, in an undirected 2D Markov Random Field model there is no explicit causal dependency of the random variables as in case of one-dimensional problems that

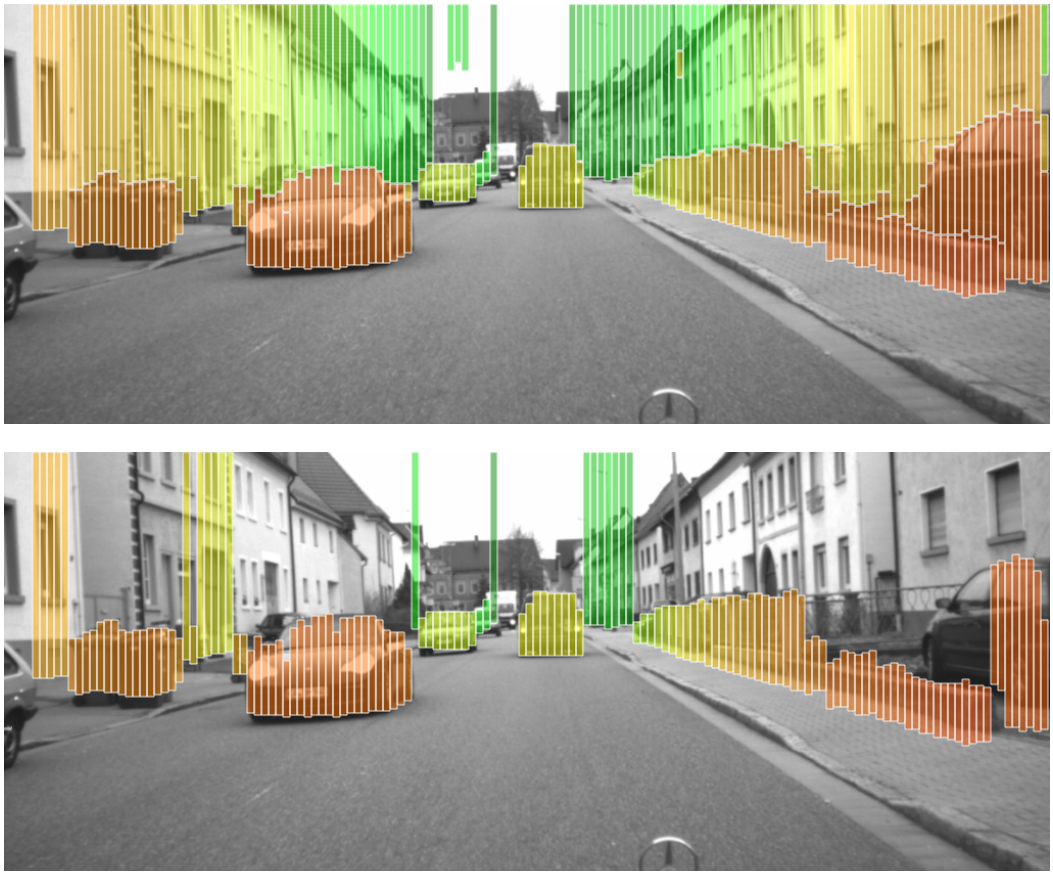


Figure 4.1: The full Stixel information above is reduced to the first Stixel row shown below. This way, in most cases, the closest and most relevant object is addressed.

can frequently be expressed as a directed acyclic graphical model, often referred to as Bayesian Network [Pearl, 1985]. Remember that a MRF factorizes into a product of potential functions over the maximal cliques of the graph. However, these potential functions do not have an obvious probabilistic interpretation such as the conditional distribution in case of a Bayesian Network. Such a probabilistic interpretation only emerges by introducing a global normalization constant Ω, see Equation 3.12. However, in most cases this normalization constant is too expensive to compute and thus these undirected models are difficult to access. In contrast, a one-dimensional Bayesian Network provides a valid factorization of a true probability distribution [Koller et al., 2007]. In directed acyclic graphs, each factor represents the conditional distribution of the corresponding variables, conditioned on the state of its parents [Bishop, 2007].
Secondly, there are efficient and exact inference methods such as Belief Propagation [Pearl, 1988] or dynamic programming [Bellman, 1954] for one-dimensional problems that do not readily generalize to higher dimensions and loopy graph structures. These arguments motivate the choice of the proposed approach.
It shall not be concealed that one-dimensional optimization alone often yields inferior results as a full two-dimensional optimization, see [Veksler, 2005] for the case of stereo reconstruction. The problem, however, in that work is probably the fact that the simpler approach (dynamic programming) did not exploit any additional information and thus its full potential. In this case, it is understandable that it performs worse. In the present work, however, further object and scenario knowledge are exploited to improve the segmentation. This additional knowledge is difficult to take into account by means of the full two-dimensional optimization.
The key advantage of the proposed approach is the possibility to take into account non-local, higher-order information such as shape or object size knowledge. It is shown explicitly that segmentation can be improved by taking into account this kind of information. Furthermore, the approach addresses the difficulties that were outlined at the end of Section 3.6, namely object occlusions and error-prone object border Stixels.
The request for higher-order regularization stems from the difficulties arising with stereo-based motion estimation. Being confronted with noisy and error-prone motion information, most local approaches prove to be insufficient to cope with the errors and the noise level of the input data. Another challenge is to extend the range of the object segmentation by means of strong regularization. More generally, to what extent is a stereo camera system currently capable of motion-based object segmentation and how stable are the segmentation results? In short, is motion segmentation applicable and suitable for driver assistance?
This chapter is structured as follows: Section 4.2 introduces the underlying segmentation problem and 4.3 proposes a possible implementation. A possible link between object segmentation and object tracking is sketched in Section 4.4. Section 4.5 presents experimental results and Section 4.6 gives a first indication how to generalize the dynamic programming step to multiple Stixel rows. Finally Section 4.7 concludes this chapter.

4.2 Optimization Problem

In this section, the segmentation task is formulated as a Bayesian optimization problem. Basically, this section is oriented towards the modeling from Chapter 3, but there are distinct differences, especially with regard to more general, non-local energy terms. A big further advantage of dynamic programming is the fact that the overall objective function does not need to be submodular for dynamic programming, thus it allows for greater modeling freedom.
In the following, the complete segmentation pipeline is outlined similar to Section 3.2. First of all, the given stereo camera system records an image sequence $\mathcal{I}$ with dense stereo information [Hirschmuller, 2005, Gehrig et al., 2009]. These dense disparity maps are subsequently segmented into the multi-layered Stixel World [Pfeiffer, 2012, Pfeiffer and Franke, 2011] partitioning an input image $I^t \in \mathcal{I}$ column-wise into several layers of one of the two classes $\mathcal{C}_{\text{Stixel}} \in \{\text{street, obstacle}\}$. In the following, the street area is left unchanged and the focus is on obstacle Stixels. See Section 2.4 for a short introduction to the Stixel World.
Subsequently, the Stixels are tracked over time in order to estimate their motion state. This Dynamic Stixel World [Pfeiffer and Franke, 2010] has been introduced in Section 2.5.
In summary, each Stixel with index i ($i = 1...N \in \mathbb{N}$) is defined by five observations including its 3D position $(u_i^t,\, Y_i^t,\, d_i^t)$ where d_i^t denotes its disparity value, u_i^t denotes the image column of the Stixel center and Y_i^t is the height of its top point in meter and its velocity $\left(\dot{X}_i^t,\, \dot{Z}_i^t\right)$ from the tracking. These five observations form a feature vector for each Stixel,

$$\vec{z}_i^{\,t} = \left(\dot{X}_i^t,\, \dot{Z}_i^t,\, u_i^t,\, Y_i^t,\, d_i^t\right)^{\mathrm{T}} \tag{4.1}$$

which in turn are again combined in a measurement array

$$\mathcal{Z}^t = \left(\vec{z}_1^{\,t},\, \ldots,\, \vec{z}_N^{\,t}\right). \tag{4.2}$$

Instead of definition 3.4, Equation 4.1 is partially defined in the image plane, which makes clear that this approach exploits even more the quantization in the image plane than Chapter 3.
Let $\mathcal{M}^t$ denote additional map knowledge, for example from externally provided maps or from statistical models and let

$$\mathbf{L}^t = \left(l_1^t,\, \ldots,\, l_N^t\right)^{\mathrm{T}} \tag{4.3}$$

denote a labeling as defined formally in Subsection 2.6.1 for a given input image I^t containing N dynamic Stixels. Again, the number of object classes varies dynamically as does the number of moving objects in real traffic scenes (denoted as M in Subsection 3.2). Additionally, there are B stationary background objects. A valid labeling $\mathbf{L}^t$ assigns each Stixel to exactly one moving object class, to the object outlier classes Out^L or Out^R, to the occluded object classes Occ^L or Occ^R, to static background Bg or to the background phantom class $\mathrm{Bg}^{\mathrm{out}}$,

$$l_i^t \in \{O_n,\, \mathrm{Occ}_n^L,\, \mathrm{Occ}_n^R,\, \mathrm{Out}_n^L,\, \mathrm{Out}_n^R,\, \mathrm{Bg}_n, \mathrm{Bg}_n^{\mathrm{out}}\}, \quad n = 1...M + B =: \mathcal{F}. \tag{4.4}$$

Figure 4.2: Example traffic scene with 5 moving vehicles. An oncoming car is shown yellow, the leading car is shown in purple. Two further moving objects are occluded from the leading vehicle. They are shown in red. Finally, there is a far distant fifth moving object Occ^R that is occluded from the traffic sign shown in green. The other cars far away cannot be resolved by the Stixel World. One Stixel on the left side of the oncoming car is classified as a background phantom Bg^{out} shown in brown. Object borders are marked with vertical, bold black lines.

At the moment, it suffices to know that there are *two* outlier classes and *two* occlusion classes referred to as "left" and "right". This class choice is due to the property of optimal substructure, see Section 4.3 for a discussion.
In the present approach, the most probable segmentation previously referred to as $\mathbf{L}^*$ in Equation 3.3 is computed taking into account the current observations $\mathcal{Z}^t$, any possibly existing map knowledge $\mathcal{M}^t$, optionally orthogonal Radar measurements $\mathcal{R}^t$ and the previous segmentation $\mathbf{L}^{t-1}$. More formally,

$$
\begin{aligned}
p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathcal{M}^t, \mathcal{R}^t, \mathbf{L}^{t-1}\right) &\propto p\left(\mathcal{Z}^t, \mathcal{M}^t, \mathcal{R}^t, \mathbf{L}^{t-1} \mid \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^t\right) \\
&= p\left(\mathcal{M}^t \mid \underbrace{\mathcal{Z}^t, \mathcal{R}^t, \mathbf{L}^{t-1}}, \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^{t-1} \mid \mathcal{Z}^t, \underbrace{\mathcal{R}^t}, \mathbf{L}^t\right) \cdot \\
&\quad p\left(\mathcal{Z}^t \mid \mathcal{R}^t, \mathbf{L}^t\right) \cdot p\left(\mathcal{R}^t \mid \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^t\right) \\
&\overset{*}{\approx} p\left(\mathcal{M}^t \mid \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^{t-1} \mid \mathcal{Z}^t, \mathbf{L}^t\right) \cdot \\
&\quad p\left(\mathcal{Z}^t \mid \mathcal{R}^t, \mathbf{L}^t\right) \cdot p\left(\mathcal{R}^t \mid \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^t\right) \\
&\propto p\left(\mathcal{M}^t \mid \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^{t-1} \mid \mathcal{Z}^t, \mathbf{L}^t\right) \cdot \\
&\quad p\left(\mathcal{Z}^t \mid \mathcal{R}^t, \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^t \mid \mathcal{R}^t\right)
\end{aligned} \tag{4.5}
$$

is maximized.
For the approximations marked with a star *, the following assumptions were made: Firstly, the map knowledge $\mathcal{M}^t$ is assumed to be independent of the Radar measurements $\mathcal{R}^t$, independent of the Stixel observations $\mathcal{Z}^t$ and independent of the informa-

tion of the previous segmentation $\mathbf{L}^{t-1}$ given knowledge of the current segmentation $\mathbf{L}^t$. This assumption seems to be reasonable since $\mathcal{M}^t$ does not know anything about the potential observations $\mathcal{R}^t$ and $\mathcal{Z}^t$ and the current labeling $\mathbf{L}^t$ is assumed to be *complete* [Thrun et al., 2005], that is $\mathbf{L}^{t-1}$ can be omitted. Secondly, the radar measurements $\mathcal{R}^t$ are assumed to be redundant to explain the old labeling $\mathbf{L}^{t-1}$, given $\mathcal{Z}^t$ and $\mathbf{L}^t$. This assumption also yields an important simplification allowing for an easier formulation without being too restrictive.
The correspondences for $p\left(\mathbf{L}^{t-1} \mid \mathcal{Z}^t, \mathbf{L}^t\right)$ between both segmentations are established via optical flow measurements.
For some scenarios it is desirable to ignore any potential Radar object knowledge $\mathcal{R}^t$ for reasons of sensor redundancy. For that reason, the symbol $\mathcal{R}^t$ is omitted in the following. Any resulting differences are pointed to in the following in the corresponding subsections.
Next, similar to Equation 3.18 the hidden parameter vector $\mathbf{\Theta}^t$ is introduced for all real moving and stationary objects, but not for the outlier classes. The property of being an outlier extends the object concept, but it just exists in the context of real objects. $\mathbf{\Theta}^t$ includes

$$\begin{aligned}\mathbf{\Theta}^t &= \{\Theta_1^t, \ldots, \Theta_{\mathcal{F}}^t\} \text{ and} \\ \Theta_n^t &= \{\mathcal{H}_n^t, \mathcal{U}_n^{Ref,t}, \mathcal{D}_n^{Ref,t}, \dot{\mathcal{X}}_n^t, \dot{\mathcal{Z}}_n^t, |\Delta\mathcal{U}|_n^t, |\Delta\mathcal{D}|_n^t\}. \end{aligned} \tag{4.6}$$

Without loss of generality the leftmost Stixel is defined to be the reference point of an object. The introduction of the parameter vector $\mathbf{\Theta}^t$ results in

$$\begin{aligned} p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathcal{M}^t, \mathbf{L}^{t-1}\right) &\overset{4.5}{\propto} \int_{\mathbf{\Theta}} p\left(\mathcal{Z}^t, \mathbf{\Theta}^t \mid \mathbf{L}^t\right) \mathrm{d}\mathbf{\Theta} \cdot \\ &\quad p\left(\mathcal{M}^t \mid \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^{t-1} \mid \mathcal{Z}^t, \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^t\right) \\ &\overset{3.24,3.26}{=} p\left(\mathcal{Z}^t \mid \mathbf{\Theta}_{map}^t, \mathbf{L}^t\right) \cdot p\left(\mathcal{M}^t \mid \mathbf{L}^t\right) \cdot \\ &\quad p\left(\mathbf{\Theta}_{map}^t \mid \mathbf{L}^t\right) \cdot \frac{(2\pi)^{K/2}}{N^{K/2}} \cdot \\ &\quad p\left(\mathbf{L}^{t-1} \mid \mathcal{Z}^t, \mathbf{L}^t\right) \cdot p\left(\mathbf{L}^t\right). \end{aligned} \tag{4.7}$$

Note that this result is completely analogous to the result 3.27. Again, K denotes the feature space dimension of $\mathbf{\Theta}^t$.

4.2.1 Parameter prior

The hidden parameter prior distribution is modeled as a first order Markov chain [Bishop, 2007], thus

$$\begin{aligned} p\left(\mathbf{\Theta}_{map}^t \mid \mathbf{L}^t\right) = p\left(\Theta_1^t, ..., \Theta_{\mathcal{F}}^t \mid \mathbf{L}^t\right) &= p\left(\Theta_1^t \mid \mathbf{L}^t\right) \cdot p\left(\Theta_2^t \mid \Theta_1^t, \mathbf{L}^t\right) \cdot \ldots \cdot \\ &\quad p\left(\Theta_{\mathcal{F}}^t \mid \Theta_1^t, ..., \Theta_{\mathcal{F}-1}^t, \mathbf{L}^t\right) \\ &\approx p\left(\Theta_1^t \mid \mathbf{L}^t\right) \cdot \prod_{n=2}^{\mathcal{F}} p\left(\Theta_n^t \mid \Theta_{n-1}^t, \mathbf{L}^t\right). \end{aligned} \tag{4.8}$$

Next, the term $p\left(\Theta_n^t \mid \Theta_{n-1}^t, \mathbf{L}^t\right)$ is analyzed in more detail. It is decomposed into various terms of lower dimensionality which are easier to model. The time index is omitted in the following since only the current time step t is considered and there is no risk of confusion.

$$
\begin{aligned}
p\left(\Theta_n^t \mid \Theta_{n-1}^t, \mathbf{L}^t\right) =& p\Big(\mathcal{U}_n^{Ref},\, \mathcal{D}_n^{Ref},\, \dot{\mathfrak{X}}_n,\, \dot{\mathcal{Z}}_n,\, |\Delta\mathcal{U}|_n\,,\, \mathcal{H}_n,\, |\Delta\mathcal{D}|_n \mid \\
& \qquad \mathcal{U}_{n-1}^{Ref},\, \mathcal{D}_{n-1}^{Ref},\, \dot{\mathfrak{X}}_{n-1},\, \dot{\mathcal{Z}}_{n-1},\, |\Delta\mathcal{U}|_{n-1}\,,\, \mathcal{H}_{n-1},\, |\Delta\mathcal{D}|_{n-1}\,, \mathbf{L}^t\Big) \\
\approx & \underbrace{p\left(\mathcal{H}_n \mid \mathbf{L}^t\right)}_{\text{height prior}} \cdot \\
& \underbrace{p\left(\dot{\mathfrak{X}}_n,\, \dot{\mathcal{Z}}_n \mid \mathbf{L}^t\right)}_{\text{velocity prior}} \cdot \\
& \underbrace{p\left(|\Delta\mathcal{U}|_n \mid \mathcal{H}_n, \dot{\mathfrak{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, \mathbf{L}^t\right)}_{\text{width prior}} \cdot \\
& \underbrace{p\left(|\Delta\mathcal{D}|_n \mid \mathcal{H}_n, \dot{\mathfrak{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, \mathbf{L}^t\right)}_{\text{length prior}} \cdot \\
& \underbrace{p\left(\mathcal{U}_n^{Ref} \mid \mathcal{U}_{n-1}^{Ref}, |\Delta\mathcal{U}|_{n-1}\,, \mathbf{L}^t\right)}_{\text{image distance prior}} \cdot \\
& \underbrace{p\left(\mathcal{D}_n^{Ref} \mid \mathcal{D}_{n-1}^{Ref}, |\Delta\mathcal{D}|_{n-1}\,, \mathbf{L}^t\right)}_{\text{depth distance prior}} .
\end{aligned}
\tag{4.9}
$$

The probability distribution is assumed to factorize as given in Equation 4.9. This special factorization property requires certain conditional independence assumptions that will be discussed below. These kind of conditional independence assumptions are the main source of tractability for most of the algorithms presented in this work. The parameter prior is assumed to factorize, among others, into a class dependent height prior $p\left(\mathcal{H}_n \mid \mathbf{L}^t\right)$ that will be addressed below 4.11.

Furthermore, the factorization includes a velocity prior $p\left(\dot{\mathfrak{X}}_n,\, \dot{\mathcal{Z}}_n \mid \mathbf{L}^t\right)$ which addresses the question what a moving object actually is. Since a moving object can be arbitrarily slow, a prior on object velocities might encode scenario specific knowledge. Informative priors can be specified for example in a highway scenario or with traffic sign information at hand.

$p\left(|\Delta\mathcal{U}|_n \mid \mathcal{H}_n, \dot{\mathfrak{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, \mathbf{L}^t\right)$ encodes a high-dimensional prior on apparent object sizes in the image plane. Definitely, this term depends on the orientation of the corresponding object which is assumed to be given by $\dot{\mathfrak{X}}_n$ and $\dot{\mathcal{Z}}_n$ for simplicity and on its distance represented by $\mathcal{D}_n^{Ref}$. Of course, there is a great variety of moving traffic participants including cars, bicycles, trams or trucks. In principle, all of these objects must define a separate class in the optimization. However, the computational effort rises quadratically in the number of classes. To avoid this drawback, a simplified access is chosen. The great object variability is taken into account by restricting the object width term to the observed fact that there is a strong correlation between object heights

and dimensions. Trucks for example are significantly larger than usual cars but they are also significantly higher. This consideration justifies the height dependency of the object width term.
Similarly, there is a comparable object length prior given by
$p\left(|\Delta\mathcal{D}|_n \mid \mathcal{H}_n, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, \mathbf{L}^t\right)$. The length is given in disparity units (pixels) here.
Finally there are two inter-object distance priors that explicitly model object-object interactions like occlusions. Additionally, rigid objects cannot interpenetrate but they usually keep a certain safety distance.

4.2.2 Data term

The data term $p\left(\mathcal{Z}^t \mid \mathbf{\Theta}_{map}^t, \mathbf{L}^t\right)$ is analyzed next. Similar to Equation 3.29, the observations are partially assumed to be independent yielding

$$
\begin{aligned}
p\left(\mathcal{Z}^t \mid \mathbf{\Theta}_{map}^t, \mathbf{L}^t\right) &= \prod_{i=1}^{N} p\left(\dot{X}_i^t,\, \dot{Z}_i^t, u_i^t, Y_i^t, d_i^t \mid \mathbf{\Theta}_{map}^t, \mathbf{L}^t\right) \\
&\overset{*}{\approx} \prod_{i=1}^{N} p\left(Y_i^t \mid \underbrace{\dot{X}_i^t,\, \dot{Z}_i^t, u_i^t, d_i^t}, \mathbf{\Theta}_{map}^t, \mathbf{L}^t\right) \cdot \\
&\quad \prod_{i=1}^{N} p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \underbrace{u_i^t, d_i^t}, \mathbf{\Theta}_{map}^t, \mathbf{L}^t\right) \cdot \\
&\quad p\left(u_1^t, d_1^t \mid \mathbf{\Theta}_{map}^t, \mathbf{L}^t\right) \cdot \prod_{i=2}^{N} p\left(u_i^t, d_i^t \mid u_{i-1}^t, d_{i-1}^t, \mathbf{\Theta}_{map}^t, \mathbf{L}^t\right) \\
&\overset{**}{\approx} \prod_{i=1}^{N} p\left(Y_i^t \mid \mathcal{H}_n^t, l_i^t\right) \cdot \prod_{i=1}^{N} p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, l_i^t\right) \cdot \\
&\quad p\left(u_1^t \mid \mathcal{U}_1^{Ref}, l_1^t\right) \cdot \prod_{i=2}^{N} p\left(u_i^t \mid u_{i-1}^t, \mathcal{U}_n^{Ref}, |\Delta\mathcal{U}|_n, l_i^t, l_{i-1}^t\right) \\
&\quad p\left(d_1^t \mid \mathcal{D}_1^{Ref}, l_1^t\right) \cdot \prod_{i=2}^{N} p\left(d_i^t \mid d_{i-1}^t, \mathcal{D}_n^{Ref}, |\Delta\mathcal{D}|_n, l_i^t, l_{i-1}^t\right). \qquad (4.10)
\end{aligned}
$$

For the approximation marked with *, the observations u_i and d_i, $i = 1...N$, were assumed to factorize in a first order Markov chain. Thus any long-range correlations between those observations were dropped for simplicity. For the approximation marked with **, the height observation was assumed to be independent of the position and independent of the velocity given the object class and its parameters. Definitely, these dependencies dominate. As long as the distance is not too large, the dependency on the distance which modifies the height noise via error propagation is indeed negligible. Secondly, the velocities are assumed to be dependent on the object velocities and its Kalman filter covariances. Any additional noise terms are assumed to be captured by these filter covariance terms.
Thirdly, the image position measurement and the disparities are assumed to be independent, see [Rabe, 2011, Pfeiffer et al., 2010] for experimental evaluations.

The unary factor $p\left(Y_i^t \mid \mathcal{H}_n^t, l_i^t\right)$ is transformed into

$$\prod_{i=1}^{N} p\left(Y_i^t \mid \mathcal{H}_n^t, l_i^t\right) =: p\left(\vec{h}^t \mid \mathcal{H}_n^t, \mathbf{L}^t\right) = \frac{p\left(\mathcal{H}_n^t \mid \vec{h}^t, \mathbf{L}^t\right) \cdot p\left(\vec{h}^t \mid \mathbf{L}^t\right)}{p\left(\mathcal{H}_n^t \mid \mathbf{L}^t\right)}, \tag{4.11}$$

where the denominator cancels the height prior defined in Equation 4.9. In summary, the data term

$$\begin{aligned} p\left(\mathcal{Z}^t \mid \mathbf{\Theta}_{map}^t, \mathbf{L}^t\right) \propto & \underbrace{\prod_{i=1}^{N} p\left(Y_i^t \mid l_i^t\right)}_{\overset{4.11}{=} p\left(\vec{h}^t \mid \mathbf{L}^t\right)} \cdot \\ & \prod_{i=1}^{N} p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, l_i^t\right) \cdot \\ & p\left(u_1^t \mid \mathcal{U}_1^{Ref}, l_1^t\right) \cdot \prod_{i=2}^{N} p\left(u_i^t \mid \mathcal{U}_n^{Ref}, \left|\Delta\mathcal{U}\right|_n, u_{i-1}^t, l_i^t, l_{i-1}^t\right) \\ & p\left(d_1^t \mid \mathcal{D}_1^{Ref}, l_1^t\right) \cdot \prod_{i=2}^{N} p\left(d_i^t \mid \mathcal{D}_n^{Ref}, \left|\Delta\mathcal{D}\right|_n, d_{i-1}^t, l_i^t, l_{i-1}^t\right) \end{aligned} \tag{4.12}$$

and the parameter prior

$$\begin{aligned} p\left(\mathbf{\Theta}_n^t \mid \mathbf{\Theta}_{n-1}^t, \mathbf{L}^t\right) \approx & \; p\left(\mathcal{H}_n^t \mid \vec{h}^t, \mathbf{L}^t\right) \cdot \\ & p\left(\dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t \mid \mathbf{L}^t\right) \cdot \\ & p\left(\left|\Delta\mathcal{U}\right|_n^t \mid \mathcal{H}_n^t, \dot{\mathcal{X}}_n^t, \dot{\mathcal{Z}}_n^t, \mathcal{D}_n^{Ref,t}, \mathbf{L}^t\right) \cdot \\ & p\left(\left|\Delta\mathcal{D}\right|_n^t \mid \mathcal{H}_n^t, \dot{\mathcal{X}}_n^t, \dot{\mathcal{Z}}_n^t, \mathcal{D}_n^{Ref,t}, \mathbf{L}^t\right) \cdot \\ & p\left(\mathcal{U}_n^{Ref,t} \mid \mathcal{U}_{n-1}^{Ref,t}, \left|\Delta\mathcal{U}\right|_{n-1}^t, \mathbf{L}^t\right) \cdot \\ & p\left(\mathcal{D}_n^{Ref,t} \mid \mathcal{D}_{n-1}^{Ref,t}, \left|\Delta\mathcal{D}\right|_{n-1}^t, \mathbf{L}^t\right) \end{aligned} \tag{4.13}$$

are optimized as objective function.
The classes O_n and Bg_n are similar to Chapter 3. The outlier classes Out^L, Out^R and $\mathrm{Bg}^{\mathrm{out}}$, however, are new. This concept is introduced next.

4.2.3 Outlier Stixels

Outlier Classes: Introduction The classes Out^L and Out^R denote phantom Stixels due to gross stereo measurement errors as they frequently appear at object borders, see Figure 4.3 for an example. In the surrounding of object borders, it might happen that stereo measurements are spread backwards along the viewing ray with grossly incorrect depth information. This phenomenon is hereinafter referred to as tear-off edges. These phantom measurements can possibly generate phantom obstacles that may lead to unwanted emergency brakings. It is important to take into account these measurement errors as part of a realistic sensor model in order not to end up with hopelessly inconsistent scene interpretations.
For both phantom classes prior knowledge exists:

- Phantom Stixels have low stereo confidences.
- Phantom Stixels arise at object borders.
- Phantom Stixel groups are small.
- Phantom Stixels increase object dimensions significantly.
- Phantom Stixels are behind real objects.
- Phantom Stixels are not stable over time in most cases.
- Phantom Stixels are difficult to track and have unreliable motion states.

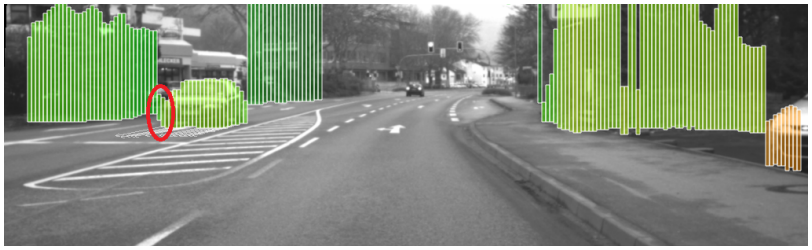
Stixel confidences as mentioned in the first point can be best described as an existence probability. Various approaches have been proposed [Pfeiffer et al., 2013,Scharwaechter, 2012]. In order to describe tear-off Stixels, the Stixel confidences proposed in [Pfeiffer et al., 2013] resulting from averaged stereo confidences are augmented with a stereo density cue. Broadly speaking, the stereo confidence describes the unambigousness of the optimum in the SGM cost cube, see [Pfeiffer et al., 2013] for details. However, the mean stereo confidence in [Pfeiffer et al., 2013] ignores all invalid pixels that were removed in an upstream left-right consistency check [Hirschmuller, 2005, Gehrig and Franke, 2007] since the average stereo confidence is calculated on the basis of the valid disparity values only. This fact is contrary to an uniform confidence description. For that reason, in the present work invalid stereo points that failed the left-right consistency check are taken into account by multiplying the resulting Stixel confidence with the stereo density value,

$$\mathfrak{C}_{\text{stixel}} = \cdot\bar{\mathfrak{C}}_{\text{stereo}} \cdot \mathfrak{D}_{\text{stereo}}, \tag{4.14}$$

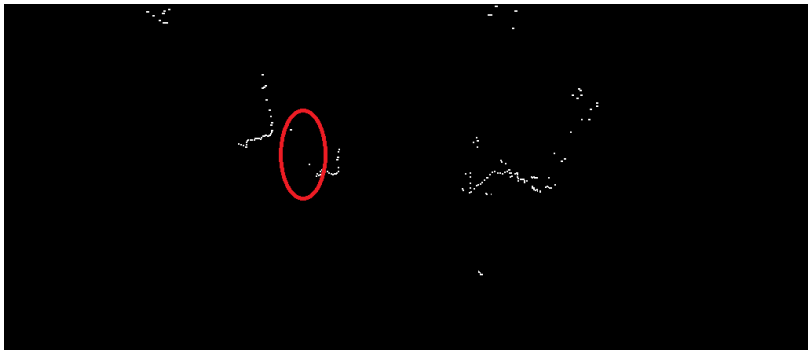
where $\bar{\mathfrak{C}}_{\text{stereo}}$ denotes the mean stereo confidence value inside a Stixel and $\mathfrak{D}_{\text{stereo}} \in [0...1]$ reflects the percentage of valid disparity measurements. This way, invalid stereo points are taken into account with a confidence of zero. See Figure 4.4 for a visualization.

Outlier Stixels: Implementation In the present work, traffic scenes are described by pairwise object interactions for reasons of efficiency, see Equation 4.8. Thus it is necessary to introduce two phantom classes, phantom objects on the left side of a moving car and phantom objects on the right side referred to as Out^L and Out^R respectively. In this case, left and right refers to the position in the image plane. The main reason for the necessity to introduce *two* outlier classes is the principle of dynamic programming. Dynamic programming as introduced in Subsection 2.6.1 computes recursively the optimal path for all Stixels from index 1 to j, $j = 1...N$, taking into account all the observations $\vec{z}_1^t$ until $\vec{z}_j^t$. It is important to note that dynamic programming does not anticipate any "future" observations, $\vec{z}_{j+1}^t$ until $\vec{z}_N^t$, and future labelings, l_j^t until l_N^t. This would be contrary to the principle of optimal substructure. This fact constitutes a difficulty for the scenario of left phantom Stixels, see Figure 4.3, since at the moment of computation in the dynamic programming it is not clear yet whether there will be a corresponding object on the right side l_{j+1}^t until l_N^t that belongs to the left phantom Stixels. By definition, outlier Stixels can only exist in the context of real object. That is, at the time of computation, the left phantom Stixels $i = 1...j$ need a confirmation

(a) Grey value image showing the traffic situation: another vehicle is oncoming on the left side.



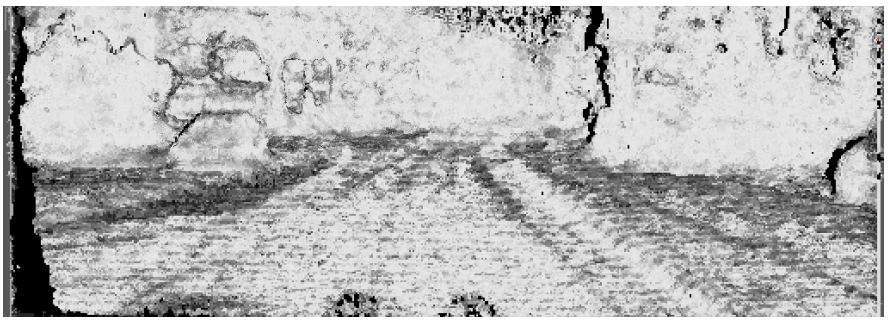
(b) Dynamic Stixel World. The distance is color encoded ranging from red (close) to green (far away).The tear-off edge is marked red.



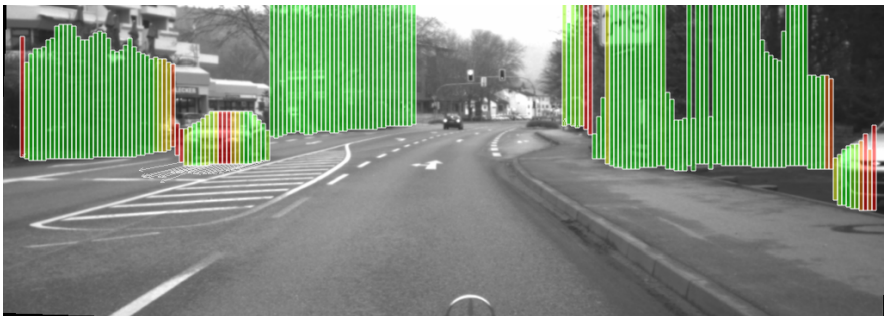
(c) Birds Eye View of the Stixel World. The tear-off edge is marked red.

(d) The solution. The phantom Stixels are correctly classified as Out_1^L outlier Stixels, the corresponding Stixel are marked brown.

Figure 4.3: Visualization of the phantom Stixel concept. Without taking into account the stereo error characteristics, phantom Stixels can arise that are difficult to associate with any of the moving objects and may lead to unwanted emergency brakings. The Stixel phantom concept yields a consistent picture of the traffic scene.



(a) Stereo confidence visualization. The color encodes the confidence (0=black, 1=white).



(b) Resulting Stixel confidence $\mathfrak{C}_{\text{stixel}}$. The color encodes the confidence (0.65=red, 0.8=green).

Figure 4.4: Visualization of the Stixel confidence concept. Typically, the outlier Stixels have a low confidence value $\mathfrak{C}_{\text{stixel}}$.

of an appropriate object on the right side $i = j + 1...N$, but this confirmation does not exist due to the principle of optimal substructure. For that reason, this case has to be provided by the introduction of two special outlier classes, Out^L and Out^R. Left outlier Stixels Out^L still need a confirmation of future observations $i = j + 1...N$ and future labelings, l_j^t until l_N^t whereas right outlier Stixels Out^R refer to "past" measurements and labelings, $i = 1...j$.
There are different transition probabilities between these classes, see Section 4.3.3. An alternative would be to extend the order of the Markov chain to multiple object configurations (at least to fourth order to capture the same effect), but this would be computationally more expensive.
The same reason makes it necessary to introduce two occluded object classes. An occluded object on the left side, Occ^L, still needs a confirmation from an object on the right side, that is there needs to be a closer object on the right that occludes the Occ^L object.
Finally, a similar phantom class is introduced for the static background, referred to as Bg^{out}. However, there is no real notion of an object for background and there are no real object borders. Accordingly, phantom Stixels can occur everywhere in static background. Insofar, there is only one background outlier class Bg^{out}.

4.3 Definition of the Energy Terms

This section elaborates in more detail the modeling of the energy terms introduced in Section 4.2. In Equation 4.4, the used object classes, given by different instances n of the moving object class O_n, of stationary background Bg_n, of the outlier classes Out_n^L and Out_n^R and Bg_n^{out} and the occluded object classes Occ_n^L and Occ_n^R, were introduced.

4.3.1 Statistical Map knowledge

The term $p\left(\mathcal{M}^t \mid \mathbf{L}^t\right)$ modeling the occurrence of background and moving objects has been introduced already in Subsection 3.3.1. The map $\mathcal{M}^t$ is assumed to contain a location-dependent occurrence probability for all object classes. So

$$p\left(\mathcal{M}^t \mid \mathbf{L}^t\right) = \prod_{(X,Z)} p\left(m_{X,Z}^t \mid \mathbf{L}^t\right) \tag{4.15}$$

is assumed to be the product over all cells $m_{X,Z}^t$ inside the map $\mathcal{M}^t$ modeling their occupancy probability as given in position-related part of Subsection 3.3.1.

4.3.2 Data term

Height data term

The height data term $p\left(Y_i^t \mid l_i^t\right)$ has been introduced already in Subsection 3.3.1. The probability distributions for moving objects and stationary background were collected from a ground truth database containing about 38000 manually labeled images, see Figure 3.4 for a visualization.

Velocity data term

The velocity data term $p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, l_i^t\right)$ is estimated as a parametric Gaussian distribution. This approach is chosen for mainly two reasons:
Firstly, it requires significantly less training data than learning the complete probability distribution. It is almost impossible to actually learn high-dimensional probability distributions completely from limited ground truth data, especially the tails of the distribution. However, the tails are important for noisy data. Since it is only possible to estimate the mean and its variance from the limited ground truth data, the principle of maximum entropy [Jaynes, 1957,Sivia, 1996] justifies to choose a Gaussian distribution. In order to deal with gross outliers, a uniform socket over the possible range of the data is added to this distribution.
Secondly, the parametric approach allows to react easier to changes in the Stixel tracking. Otherwise, the complete distribution has to be relearned. For these reasons, the velocity distribution is estimated similar to Subsection 3.3.2 as

$$p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, l_i^t\right) = (1 - p_{out}) \cdot \eta \cdot \mathcal{N}\left(\left(\dot{X}_i^t, \dot{Z}_i^t\right), \left(\dot{\mathcal{X}}_n^t, \dot{\mathcal{Z}}_n^t\right), \mathbf{\Sigma}_i^t\right) + \frac{p_{out}}{|V_x^{max} - V_x^{min}| \cdot |V_z^{max} - V_z^{min}|}. \tag{4.16}$$

In this equation, η is a normalization constant, p_{out} models the outlier probability for robustness and V_x^{min}, V_x^{max} and V_z^{min}, V_z^{max} limit the value range of the uniform distribution. Finally $\mathbf{\Sigma}_i^t$ is directly given by the covariance estimate of the Kalman filter. The outlier probability is estimated once on a validation set and is kept fixed in the experiments. For static background, $\dot{\mathcal{X}}_n^t = \dot{\mathcal{Z}}_n^t := 0$ is not a parameter but is constant.

Multiple Filter Hypotheses The Stixel tracking can use multiple Kalman filter hypotheses for faster convergence, see [Rabe, 2011]. In this case, there are several reactive filter hypotheses and other static, slowly-converging filters. The currently best filter is selected by the minimal Normalized Innovation Squared (NIS) criterion or by Maximum Likelihood. After some measurements, the filters converge and the initial state estimates no longer play a role. The problem is that after a few time steps only, the initial estimates play a very real role. The question is how this initial uncertainty can be modeled.
First, in case of multiple velocity measurements an adequate probability expression is questionable. Typically, the Stixel tracking yields F Gaussian filter hypotheses on the estimated velocity and a filter agreement given by the NIS. In this sense, the current observation is given by the respective flow and disparity raw track. The filter assignment step is a correspondence problem, since it is not clear which filter hypothesis is correct and has generated the observation. The usual procedure would be to select the most likely data association based on the NIS. However, an early decision has some disadvantages and it introduces an unwanted nonlinearity that is difficult to model. Instead, it is better to maintain all hypotheses.
For this purpose, the likelihood for the velocity measurements $\dot{X}_i^t$ and $\dot{Z}_i^t$ is expressed as a marginal over the latent filter variable $f_i = 1...F$, which associates the i-th mea-

surements with the 1st...F-th filter hypothesis:

$$\begin{aligned} p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, l_i^t\right) &= \sum_{f_i} p\left(\dot{X}_i^t,\, \dot{Z}_i^t, f_i \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, l_i^t\right) \\ &= \sum_{f_i} p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid f_i, \underbrace{\dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t}, l_i^t\right) \cdot p\left(f_i \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, l_i^t\right) \\ &\approx \sum_{f_i} p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid f_i, l_i^t\right) \cdot p\left(f_i \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, l_i^t\right). \end{aligned} \tag{4.17}$$

In the third line, any unnecessary conditioning statements have been omitted: the observations $\dot{X}_i^t$ and $\dot{Z}_i^t$ are assumed to be specified by the Stixel filter hypothesis f_i completely. Again, each single Stixel measurement scatters around the object velocity $\dot{\mathcal{X}}_n^t$ and $\dot{\mathcal{Z}}_n^t$, but the single Stixel description is assumed to be more precise, e.g. for turning objects where the Stixels' velocities are non-constant over the whole object. Furthermore, f_i provides a covariance estimate that helps to bound the expected Stixel noise.
The current filter likelihood is assumed to be specified by the current NIS similar to the 1 filter case 4.16

$$\begin{aligned} p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid f_i, l_i^t\right) = (1 - p_{out}) \cdot \eta \cdot \exp\left(-\mathrm{NIS}_{f_i}\right) + \\ \frac{p_{out}}{|V_x^{max} - V_x^{min}| \cdot |V_z^{max} - V_z^{min}|}, \end{aligned} \tag{4.18}$$

and the different filter object likelihoods are given by

$$p\left(f_i \mid \dot{\mathcal{X}}_n^t,\, \dot{\mathcal{Z}}_n^t, l_i^t\right) = \frac{\mathcal{N}\left(\left(\dot{X}_{f_i}^t, \dot{Z}_{f_i}^t\right), \left(\dot{\mathcal{X}}_n^t, \dot{\mathcal{Z}}_n^t\right), \mathbf{\Sigma}_{f_i}^t\right)}{\sum_{f_j=1}^{F} \mathcal{N}\left(\left(\dot{X}_{f_j}^t, \dot{Z}_{f_j}^t\right), \left(\dot{\mathcal{X}}_n^t, \dot{\mathcal{Z}}_n^t\right), \mathbf{\Sigma}_{f_j}^t\right)}, \tag{4.19}$$

where $\dot{X}_{f_i}^t$ and $\dot{Z}_{f_i}^t$ denote the velocity estimate of the f_i-th filter hypothesis and $\mathbf{\Sigma}_{f_i}^t$ is the corresponding covariance estimation.

Optional Radar Measurements In the case of additional Radar measurements, the object velocities $\vec{\mathcal{V}}_n := \left(\dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n\right)^T$ can be replaced by the Radar velocities defined in Equation 4.61. However, the Radar sensor has an increased uncertainty with respect to crossing traffic. This uncertainty has to be taken into account in Equation 4.17 by marginalization over the true, but unknown object velocities $\left(\hat{\dot{\mathcal{X}}}_r, \hat{\dot{\mathcal{Z}}}_r\right)$:

$$\begin{aligned} p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \dot{\mathcal{X}}_r^t,\, \dot{\mathcal{Z}}_r^t, l_i^t\right) &= \int\int p\left(\dot{X}_i^t,\, \dot{Z}_i^t, \hat{\dot{\mathcal{X}}}_r, \hat{\dot{\mathcal{Z}}}_r \mid \dot{\mathcal{X}}_r^t,\, \dot{\mathcal{Z}}_r^t, l_i^t\right) \mathrm{d}\hat{\dot{\mathcal{X}}}_r\, \mathrm{d}\hat{\dot{\mathcal{Z}}}_r \\ &= \int\int p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \hat{\dot{\mathcal{X}}}_r, \hat{\dot{\mathcal{Z}}}_r, \underbrace{\dot{\mathcal{X}}_r^t,\, \dot{\mathcal{Z}}_r^t}, l_i^t\right) \cdot p\left(\hat{\dot{\mathcal{X}}}_r, \hat{\dot{\mathcal{Z}}}_r \mid \dot{\mathcal{X}}_r^t,\, \dot{\mathcal{Z}}_r^t, l_i^t\right) \mathrm{d}\hat{\dot{\mathcal{X}}}_r\, \mathrm{d}\hat{\dot{\mathcal{Z}}}_r \\ &\approx \int\int p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \hat{\dot{\mathcal{X}}}_r, \hat{\dot{\mathcal{Z}}}_r, l_i^t\right) \cdot p\left(\hat{\dot{\mathcal{X}}}_r, \hat{\dot{\mathcal{Z}}}_r \mid \dot{\mathcal{X}}_r^t,\, \dot{\mathcal{Z}}_r^t, l_i^t\right) \mathrm{d}\hat{\dot{\mathcal{X}}}_r\, \mathrm{d}\hat{\dot{\mathcal{Z}}}_r. \end{aligned} \tag{4.20}$$

Thus, the measurement probability is convolved with the uncertainty of the Radar sensor. The Radar sensor provides Gaussian uncertainties,

$$p\left(\hat{\mathfrak{X}}_r, \hat{\dot{\mathcal{Z}}}_r \mid \dot{\mathcal{X}}_r^t,\, \dot{\mathcal{Z}}_r^t, l_i^t\right) = \mathcal{N}\left(\hat{\mathfrak{X}}_r, \hat{\dot{\mathcal{Z}}}_r, \dot{\mathcal{X}}_r^t,\, \dot{\mathcal{Z}}_r^t, \mathbf{\Sigma}_{\text{Radar}}\right), \tag{4.21}$$

and the likelihood term is similar to Equation 4.19

$$p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \hat{\mathfrak{X}}_r, \hat{\dot{\mathcal{Z}}}_r, l_i^t\right) = (1 - p_{out}) \cdot \eta \cdot \mathcal{N}\left(\left(\dot{X}_i^t,\, \dot{Z}_i^t\right), \left(\hat{\mathfrak{X}}_r, \hat{\dot{\mathcal{Z}}}_r\right), \mathbf{\Sigma}_i^t\right) + \frac{p_{out}}{|V_x^{max} - V_x^{min}| \cdot |V_z^{max} - V_z^{min}|}, \tag{4.22}$$

so the convolution of Equation 4.20 can be done analytically resulting in

$$p\left(\dot{X}_i^t,\, \dot{Z}_i^t \mid \dot{\mathcal{X}}_r^t,\, \dot{\mathcal{Z}}_r^t, l_i^t\right) = (1 - p_{out}) \cdot \eta \cdot \mathcal{N}\left(\dot{X}_i^t,\, \dot{Z}_i^t, \dot{\mathcal{X}}_r^t,\, \dot{\mathcal{Z}}_r^t, \left(\mathbf{\Sigma}_i^2 + \mathbf{\Sigma}_{\text{Radar}}^2\right)^{1/2}\right) + \frac{p_{out}}{|V_x^{max} - V_x^{min}| \cdot |V_z^{max} - V_z^{min}|}, \tag{4.23}$$

where the time index of the covariance matrix has been omitted for the sake of better readability.

Distance data terms

For the position terms,

$$p\left(u_1^t \mid \mathcal{U}_1^{Ref}, l_1^t\right) := \begin{cases} 1, & \text{if } u_1^t = \mathcal{U}_1^{Ref} \\ 0, & \text{otherwise.} \end{cases} \tag{4.24}$$

and

$$p\left(d_1^t \mid \mathcal{D}_1^{Ref}, l_1^t\right) := \begin{cases} 1, & \text{if } d_1^t = \mathcal{D}_1^{Ref} \\ 0, & \text{otherwise.} \end{cases} \tag{4.25}$$

are set. This choice is reasonable due to the definition of the reference point in Section 4.2 as the leftmost object Stixel.
The terms $p\left(u_i^t \mid u_{i-1}^t, \mathcal{U}_n^{Ref}, |\Delta\mathcal{U}|_n, l_i^t, l_{i-1}^t\right)$ and $p\left(d_i^t \mid d_{i-1}^t, \mathcal{D}_n^{Ref}, |\Delta\mathcal{D}|_n, l_i^t, l_{i-1}^t\right)$ from Equation 4.12 describe distances in image coordinates, representing a simple object *shape model*. Typically, these terms are highly supermodular for larger distances, making impossible inference via graphcut for example. This simple shape model takes into account the fact that moving objects like cars or bicyclists are compact and do not have large holes.
For static background, it turns out to be more difficult to formulate an adequate shape model because the static infrastructure is so wide-ranging. Furthermore, the desired degree of detail of the background remains an open question. For those reasons, the used background model follows a broader distribution which overlaps significantly with the moving object distributions since shape alone cannot classify the motion state.
For all classes, $p\left(u_i^t \mid u_{i-1}^t, \mathcal{U}_n^{Ref}, |\Delta\mathcal{U}|_n, l_i^t, l_{i-1}^t\right)$ encodes the probability to miss Stixels on the same physical object. The Stixels are spaced on a fixed grid along the horizontal image coordinate u, there is no real measurement noise for this coordinate. Holes can

occur with respect to this fixed grid for example due to low stereo confidences. In this case, an obstacle is missed (false negative example). Furthermore, it might happen that there are simply no obstacles, for example on an empty street or the obstacles are far away outside a user-defined region of interest.
The probability for the cases that both l_{i-1}^t and l_i^t belong to same physical object (moving or background, which also includes the transitions from or to outliers, and occluded objects) is estimated as

$$p\left(u_i^t \mid u_{i-1}^t, \mathcal{U}_n^{Ref}, |\Delta\mathcal{U}|_n, l_i^t = l_{i-1}^t\right) = \begin{cases} \eta \cdot \mathcal{N}\left(u_i^t, u_{i-1}^t + w_{\text{stixel}}, \mathbf{\Sigma}_u\right), & \text{if } u_i^t \geq u_{i-1}^t + w_{\text{stixel}} \wedge u_i^t \leq \mathcal{U}_n^{Ref} + |\Delta\mathcal{U}|_n \\ 0, & \text{otherwise.} \end{cases} \tag{4.26}$$

The normalizer η evaluates to

$$\eta = \left(\int_{u_{i-1}^t + w_{\text{stixel}}}^{\mathcal{U}_n^{Ref} + |\Delta\mathcal{U}|_n} \mathcal{N}\left(u_i^t, u_{i-1}^t + w_{\text{stixel}}, \mathbf{\Sigma}_u\right) \mathrm{d}u_i^t \right)^{-1}, \tag{4.27}$$

where w_{stixel} denotes the fixed width of one Stixel in pixels. Note that $\mathbf{\Sigma}_u$ can be different for the different classes.
The remaining cases describing the transitions between different physical objects are modeled according to Equation 4.24

$$p\left(u_i^t \mid u_{i-1}^t, \mathcal{U}_n^{Ref}, |\Delta\mathcal{U}|_n, l_i^t \neq l_{i-1}^t\right) = \begin{cases} 1, & \text{if } u_i^t = \mathcal{U}_n^{Ref} \\ 0, & \text{otherwise.} \end{cases} \tag{4.28}$$

Similarly, $p\left(d_i^t \mid d_{i-1}^t, \mathcal{D}_n^{Ref}, |\Delta\mathcal{D}|_n, l_i^t, l_{i-1}^t\right)$ encodes the object depth compactness assumption. The probability for the cases that both l_{i-1}^t and l_i^t belong to same physical object, excluding the outlier classes, is

$$p\left(d_i^t \mid d_{i-1}^t, \mathcal{D}_n^{Ref}, |\Delta\mathcal{D}|_n, l_{i-1}^t = l_i^t\right) = \frac{p_{out}}{d_{max} - d_{min}} + (1 - p_{out}) \cdot \eta \cdot \mathcal{N}\left(d_i^t, d_{i-1}^t, \mathbf{\Sigma}_d\right). \tag{4.29}$$

In this expression, p_{out} describes an uniform distribution percentage which can be different for different classes. For the used stereo camera setup the domain for valid disparities extends from $d_{min} = 0$ to $d_{max} = 128$ pixels. Again, η is a normalization constant.
The disparity measurement probability describing the transition from any class l_{i-1}^t to a right phantom Stixel $l_i^t = \text{Out}^R$ is modeled via a sigmoid function

$$p\left(d_i^t \mid d_{i-1}^t, \mathcal{D}_n^{Ref}, |\Delta\mathcal{D}|_n, l_{i-1}^t, \text{Out}^R\right) = \frac{\eta \cdot (1 - p_{out})}{1 + \exp\left(-\left(d_{i-1}^t - d_i^t - \mu_d\right) / \mathbf{\Sigma}_d\right)} + \frac{p_{out}}{d_{max} - d_{min}}. \tag{4.30}$$

This probability function is large if $d_i^t \ll d_{i-1}^t$, i.e. Stixel i is significantly behind Stixel $i-1$, see the phantom class specification 4.2.3. The transition from a left phantom Stixel $l_{i-1}^t = \text{Out}^L$ to any other class is defined analogously, where $\mathbf{\Sigma}_d$ is replaced with $-\mathbf{\Sigma}_d$. The expected depth-jump μ_d is oriented to the measurement accuracy of the used stereo system, typically $\mu_d = 3 \cdot |\mathbf{\Sigma}_d|$.
Finally, the inter-object transitions $l_{i-1}^t \neq l_i^t$ between two different, physical objects are defined similar to Equation 4.28 and 4.25 as

$$p\left(d_i^t \mid d_{i-1}^t, \mathcal{D}_n^{Ref}, |\Delta\mathcal{D}|_n, l_{i-1}^t \neq l_i^t\right) = \begin{cases} 1, & \text{if } d_i^t = \mathcal{D}_n^{Ref} \\ 0, & \text{otherwise.} \end{cases} \tag{4.31}$$

Next, the parameter prior defined in Equation 4.13 is specified.

4.3.3 Parameter prior

Velocity prior term

The velocity prior $p\left(\dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n \mid \mathbf{L}^t\right)$ encodes any prior knowledge on object velocities. In this work, without knowing anything better, the naive assumption of a uniform velocity prior is made. This means that the true object velocity is assumed to lie between the limits $[V_{x,min}, V_{x,max}] \otimes [V_{z,min}, V_{z,max}]$. The limits are plausibility values and should depend on the traffic scenario under investigation. In particular, there is no velocity threshold at this point for moving objects. In certain traffic scenarios as highways, prior knowledge on object velocities exists. However, this is far more difficult for urban traffic scenes with starting and braking objects. For this reason, a more contextual approach is chosen instead of hard prior assumptions. The velocity prior is defined very general as

$$p\left(\dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n \mid O_n\right) = \frac{1}{|V_x^{max} - V_x^{min}| \cdot |V_z^{max} - V_z^{min}|}. \tag{4.32}$$

A similar prior needs not to be specified for the static background class since the object velocity is excluded in the integration 4.7. Static background is known to be static with $\dot{\mathcal{X}}_n = 0$ and $\dot{\mathcal{Z}}_n = 0$.

Height prior term

For the height prior probability $p\left(\mathcal{H}_n^t \mid \vec{h}^t, \mathbf{L}^t\right)$, the prior variable $\mathcal{H}_n^t$ is defined to be the observed mean height. For one object O_n, the probability of a certain mean object height with N associated observations is given by

$$p\left(\mathcal{H}_n^t \mid \vec{h}^t, \mathbf{L}^t\right) = \left(\frac{1}{\sigma_h\sqrt{2\pi}}\right)^N \exp\left[-\frac{1}{2}\sum_{i=1}^{N}\left(\frac{Y_i^t - \mathcal{H}_n^t}{\sigma_h}\right)^2\right], \tag{4.33}$$

assuming independent height measurements. In this case, the most probable height is given by the mean value

$$\mathcal{H}_{n,map}^t = \frac{1}{N}\sum_{i=1}^{N} Y_i^t, \tag{4.34}$$

with the uncertainty

$$\sigma^2_{\mathcal{H}^t_{n,map}} = \frac{1}{N \cdot (N-1)} \sum_{i=1}^{N} \left(Y_i^t - \mathcal{H}^t_{n,map} \right)^2 . \tag{4.35}$$

In total, the height prior aggregates objects with similar heights and tends to break highly heterogeneous objects.

Object size prior terms

The width prior $p\left(|\Delta\mathcal{U}|_n \mid \mathcal{H}_n, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, \mathbf{L}^t\right)$ is assumed to depend on the height $\mathcal{H}_n$. The higher an object is, the wider it usually is. This simplifying assumption makes a special treatment of buses or trucks unnecessary.
Formally, the dependence on $\mathcal{H}_n$ increases the dimension of the width prior. In order to reduce complexity, a simple height threshold H^T is specified. Mean object heights higher than this threshold suggest using a truck model whereas lower heights indicate that the car model might be a better object description

$$\begin{aligned} &p\left(|\Delta\mathcal{U}|_n \mid \mathcal{H}_n, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, O_n\right) \\ &\qquad = \begin{cases} p\left(|\Delta\mathcal{U}|_n \mid \text{truck}, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, O_n\right), & \text{if } \mathcal{H}_n \geq H^T \\ p\left(|\Delta\mathcal{U}|_n \mid \text{vehicle}, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, O_n\right), & \text{otherwise.} \end{cases} \end{aligned} \tag{4.36}$$

Apart from that, it is assumed that the other dependencies, namely the distance of the object $\mathcal{D}_n^{Ref}$ and its orientation given by its velocity components $\dot{\mathcal{X}}_n$ and $\dot{\mathcal{Z}}_n$ can be determined with sufficient precision. Small variations in these parameters lead to small variations of the apparent object width due to the continuity of the image projection equations. In this case, it is admissible to assume that essentially the width of the object width distribution $p\left(|\Delta\mathcal{U}|_n \mid \mathcal{H}_n, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, O_n\right)$ results from the inherent car width distribution. This distribution is assumed to be Gaussian for simplicity and therefore also

$$\begin{aligned} p\left(|\Delta\mathcal{U}|_n \mid \mathcal{H}_n, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, O_n\right) = (1-\pi_{bike}) \cdot \mathcal{N}\left(|\Delta\mathcal{U}|_n, \Delta\mathcal{U}_{n,exp}, \sigma_{|\Delta\mathcal{U}|_n}\right) + \\ \pi_{bike} \cdot \mathcal{N}\left(|\Delta\mathcal{U}|_n, \Delta\mathcal{U}^{bike}_{n,exp}, \sigma^{bike}_{|\Delta\mathcal{U}|_n}\right), \end{aligned} \tag{4.37}$$

as a first approximation, where $\Delta\mathcal{U}_{n,exp}$ denotes the expected image width and the term with π_{bike} accounts for bicyclists. In cases where the remaining parameters cannot be estimated with sufficient precision, a more complex error propagation becomes necessary. The width $\sigma_{|\Delta\mathcal{U}|_n}$ can be estimated by error propagation as follows.
An object point $\vec{x}^t$ of the vehicle cuboid shown in Figure 4.5 is assumed to have the object coordinates specified in this figure. In general, the object has a certain orientation specified by the gear angle ψ. In this work, a left-handed coordinate system is assumed where positive angles correspond to clockwise rotations. In Figure 4.5, it is assumed for simplicity that the object rotation axis on the vehicle rear axis coincides with the object center indicated with a blue dot in the middle of the object. However, the following considerations can be easily transferred to another rotation point. The

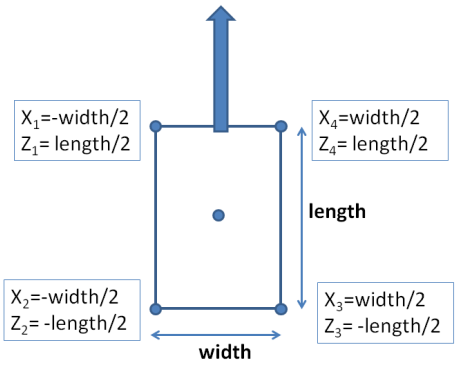


Figure 4.5: Figure illustrating the width expectation described in the main text. The arrow indicates the moving direction of the vehicle. The coordinates of the corner points are given in object coordinates relative to the object center $\mathcal{X}_n^t$ and $\mathcal{Z}_n^t$ in the middle.

object points ${}^o\vec{x}_k^t = \left({}^oX_k^t, {}^oY_k^t, {}^oZ_k^t, 1\right)^T$ given in homogeneous coordinates with respect to the respective object coordinate system can be transferred to the camera coordinate system

$$ {}^c\vec{x}_k^t = {}^c\mathbf{M}_o \cdot {}^o\vec{x}_k^t, \tag{4.38} $$

where ${}^c\mathbf{M}_o$ is a homogeneous matrix given by

$$ {}^c\mathbf{M}_o = \begin{pmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{pmatrix} \cdot \begin{pmatrix} \cos\psi & 0 & \sin\psi & \mathcal{X}_n^t \\ 0 & 1 & 0 & \mathcal{Y}_n^t \\ -\sin\psi & 0 & \cos\psi & \mathcal{Z}_n^t \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{4.39} $$

and

$$ \psi := \arctan 2 \left(\dot{\mathcal{X}}_n^t, \dot{\mathcal{Z}}_n^t \right). \tag{4.40} $$

The first matrix defines the extrinsic camera parameters that describe the location and orientation of the camera frame with respect to the world coordinate system. The second matrix defines the pose (position and orientation) of the respective camera coordinate system. A point in the camera coordinate system can be transformed into image coordinates u_k^t and d_k^t via the projection equations given by

$$ \begin{aligned} u_k^t &= \frac{f_x \cdot {}^cX_k^t}{{}^cZ_k^t} + u_{\text{hor}} \\ d_k^t &= \frac{b \cdot f_x}{{}^cZ_k^t}. \end{aligned} \tag{4.41} $$

The assumptions made above justify to assume that these transformations are known exactly.
Now, the apparent cuboid width in image coordinates can be defined as

$$ \Delta\mathcal{U}_{n,exp} := \max u_k - \min u_k, \;\; k = 1...4, \tag{4.42} $$

and the apparent cuboid length is given by

$$\Delta\mathcal{D}_{n,exp} := \left|d_{k_{max}} - d_{k_{min}}\right|, \tag{4.43}$$

where

$$\begin{aligned} k_{max} &:= \arg\max_k u_k, \\ k_{min} &:= \arg\min_k u_k \end{aligned} \tag{4.44}$$

are defined.
Since this is clearly a non-linear relation, the width and length variations are estimated via finite differences,

$$\begin{aligned} \left|\sigma_{|\Delta\mathcal{D}|_n}\right| \approx &\left|\left(\frac{\Delta\mathcal{D}_{n,exp}\left(\text{width}+\epsilon,\text{length}\right) - \Delta\mathcal{D}_{n,exp}\left(\text{width},\text{length}\right)}{\epsilon}\right)\right| \cdot \sigma_{\text{width}} + \\ &\left|\left(\frac{\Delta\mathcal{D}_{n,exp}\left(\text{width},\text{length}+\epsilon\right) - \Delta\mathcal{D}_{n,exp}\left(\text{width},\text{length}\right)}{\epsilon}\right)\right| \cdot \sigma_{\text{length}} \end{aligned} \tag{4.45}$$

and

$$\begin{aligned} \left|\sigma_{|\Delta\mathcal{U}|_n}\right| \approx &\left|\left(\frac{\Delta\mathcal{U}_{n,exp}\left(\text{width}+\epsilon,\text{length}\right) - \Delta\mathcal{U}_{n,exp}\left(\text{width},\text{length}\right)}{\epsilon}\right)\right| \cdot \sigma_{\text{width}} + \\ &\left|\left(\frac{\Delta\mathcal{U}_{n,exp}\left(\text{width},\text{length}+\epsilon\right) - \Delta\mathcal{U}_{n,exp}\left(\text{width},\text{length}\right)}{\epsilon}\right)\right| \cdot \sigma_{\text{length}}. \end{aligned} \tag{4.46}$$

The width σ_{width} and length variations σ_{length} are model variations that can be estimated with reasonable accuracy based on prior knowledge.
For the truck model, different width and length variations are assumed. The resulting distributions can be precomputed, normalized and stored for discrete variations of $\mathcal{D}_n^{Ref}$ and the orientation given by ψ.
Partially occluded objects are smaller than regular objects. The size expectation is modeled

$$\begin{aligned} p\left(|\Delta\mathcal{U}|_n \mid \mathcal{H}_n, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, Occ\right) = \\ \begin{cases} p\left(|\Delta\mathcal{U}|_n \mid \text{truck}, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, Occ\right), & \text{if } \mathcal{H}_n \geq H^T \\ p\left(|\Delta\mathcal{U}|_n \mid \text{vehicle}, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, Occ\right), & \text{otherwise,} \end{cases} \end{aligned} \tag{4.47}$$

where for example

$$\begin{aligned} p\left(|\Delta\mathcal{U}|_n \mid \text{vehicle}, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, Occ\right) = (1-p_{out}) \cdot \eta \cdot p_{Size}\left(|\Delta\mathcal{U}|_n, \Delta\mathcal{U}_{n,exp}, \sigma_{|\Delta\mathcal{U}|_n}\right) + \\ \frac{p_{out}}{u_{max} - u_{min}}, \end{aligned} \tag{4.48}$$

where

$$\begin{aligned} p_{Size}\left(|\Delta\mathcal{U}|_n, \Delta\mathcal{U}_{n,exp}, \sigma_{|\Delta\mathcal{U}|_n}\right) = \\ \begin{cases} p_{high} \cdot \exp\left(\frac{-\left(|\Delta\mathcal{U}|_n - \Delta\mathcal{U}_{n,exp}\right)^2}{2\sigma^2_{|\Delta\mathcal{U}|_n}}\right), & \text{if } |\Delta\mathcal{U}|_n \geq \Delta\mathcal{U}_{n,exp} \\ p_{high}, & \text{otherwise.} \end{cases} \end{aligned} \tag{4.49}$$

The quadratic exponent as opposed to an ordinary sigmoid-shaped function ensures that $p\left(|\Delta\mathcal{U}|_n \mid \text{vehicle}, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, Occ\right)$ has the same asymptotic scaling behavior as $p\left(|\Delta\mathcal{U}|_n \mid \text{vehicle}, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, O_n\right)$. p_{high} is a normalization constant. In analogy, the same applies for $p\left(|\Delta\mathcal{D}|_n \mid \text{vehicle}, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, Occ\right)$, $p\left(|\Delta\mathcal{U}|_n \mid \text{truck}, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, Occ\right)$ and $p\left(|\Delta\mathcal{D}|_n \mid \text{truck}, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, Occ\right)$.
The static background class has much broader dimension distributions. The width of this distribution depends on the desired level of detail. Which objects or structures shall be resolved? In the present work, a low level of detail was striven for since this choice delivered the most stable results.
For that reason, initially a uniform distribution for the background object dimensions was assumed. However, problems occurred especially for parking vehicles since the uniform distribution is very crude and a parking vehicle can be described much better by the vehicle shape model as described above. The respective dimension probability distributions differ by several orders of magnitude. The developed system made many mistakes even for moderate noise. In principle, the question rises whether a separation of moving and stationary objects can be performed based on their dimensions at all or if it is necessary to introduce a special stationary vehicle class.
The solution to this problem is simply the fact that the stationary background dimension distribution cannot be described by a simple uniform distribution. Instead, the described ambiguity needs to be taken into account by the background dimension distribution. Accordingly, the background dimension distribution is formulated as a mixture distribution

$$\begin{aligned} p\left(|\Delta\mathcal{U}|_n \mid \mathcal{H}_n, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, Bg\right) =& \pi_{vehicle} \cdot p\left(|\Delta\mathcal{U}|_n \mid \mathcal{H}_n, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, O_n\right) + \\ & \pi_{occ} \cdot p\left(|\Delta\mathcal{U}|_n \mid \mathcal{H}_n, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, Occ\right) + \\ & \frac{1 - \pi_{vehicle} - \pi_{occ}}{u_{max} - u_{min}}. \end{aligned} \tag{4.50}$$

The mixture coefficients $\pi_{vehicle}$ and π_{occ} specify the expected percentage of parking vehicles/trucks and stationary occluded vehicles/trucks respectively. The same holds analogously for $p\left(|\Delta\mathcal{D}|_n \mid \mathcal{H}_n, \dot{\mathcal{X}}_n, \dot{\mathcal{Z}}_n, \mathcal{D}_n^{Ref}, Bg\right)$.

Object distance prior

The image distance prior is defined as

$$p\left(\mathcal{U}_n^{Ref} \mid \mathcal{U}_{n-1}^{Ref}, |\Delta\mathcal{U}|_{n-1}, \mathbf{L}^t\right) = \begin{cases} \frac{1}{W - \mathcal{U}_{n-1}^{Ref} - |\Delta\mathcal{U}|_{n-1} - w_{\text{stixel}}}, & \text{if } \mathcal{U}_n^{Ref} \geq \mathcal{U}_{n-1}^{Ref} + |\Delta\mathcal{U}|_{n-1} + w_{\text{stixel}} \\ 0, & \text{otherwise,} \end{cases} \tag{4.51}$$

where the image width W was already introduced in Section 2.4. This uniform inter-object image distance prior competes with the intra-object distance term defined in equation 4.26.

Next, the depth distance prior $p\left(\mathcal{D}_n^{Ref} \mid \mathcal{D}_{n-1}^{Ref}, |\Delta\mathcal{D}|_{n-1}, \mathbf{L}^t\right)$ is specified. Typically, different objects keep a certain distance. This distance expectation is modeled via a symmetric sigmoid function

$$p\left(\mathcal{D}_n^{Ref} \mid \mathcal{D}_{n-1}^{Ref}, |\Delta\mathcal{D}|_{n-1}, O \leftrightarrows Bg\right) = \frac{\eta \cdot (1 - p_{out})}{1 + \exp\left(-\left|\mathcal{D}_n^{Ref} - \mathcal{D}_{n-1}^{Ref} - |\Delta\mathcal{D}|_{n-1} - \mu_{disp}\right| / \sigma_{disp}\right)} + \frac{p_{out}}{d_{max} - d_{min}}. \tag{4.52}$$

σ_{disp} models the disparity uncertainty and has been introduced already in Equation 3.32. $\mu_{disp} \geq 0$ models the expected disparity deviance between neighboring objects. Equation 4.52 is large when $\mathcal{D}_n^{Ref}$ is much larger or smaller than $\mathcal{D}_{n-1}^{Ref}$. This probability distribution models the transitions from moving objects to stationary objects and vice versa.

The transitions to occluded objects take into account a *depth ordering constraint*. Simultaneously, both objects need to be adjacent to each other in the image plane (*adjacency constraint*). These are the only constraints that can be formulated for occlusion detection from a geometric viewpoint. In this case, the distance expectation is modeled via a signed sigmoid function since occluded objects have a greater depth than the object which occludes them:

$$p\left(\mathcal{D}_n^{Ref} \mid \mathcal{D}_{n-1}^{Ref}, |\Delta\mathcal{D}|_{n-1}, O \rightarrow \mathrm{Occ}^R / \mathrm{Occ}^L \rightarrow O\right) = \frac{\eta \cdot (1 - p_{out})}{1 + \exp\left(-sign\left(\mu_{disp}\right) \cdot \left(\mathcal{D}_n^{Ref} - \mathcal{D}_{n-1}^{Ref} - |\Delta\mathcal{D}|_{n-1} - |\mu_{disp}|\right) / \sigma_{disp}\right)} + \frac{p_{out}}{d_{max} - d_{min}}, \tag{4.53}$$

where $\mu_{disp} > 0$ for the transition from Occ^L to another object, $\mu_{disp} < 0$ for the transition from a foreground object to Occ^R and

$$sign\left(\mu_{disp}\right) := \begin{cases} 1, & \text{if } \mu_{disp} \geq 0 \\ -1, & \text{otherwise.} \end{cases} \tag{4.54}$$

Furthermore, there are various forbidden transitions. For example, two objects cannot simultaneously occlude each other. All the possible class transitions are summarized in Table 4.1.

As shown there, the transition from background Bg to a right phantom Out^R is forbidden since a right phantom needs a preceding moving object to be defined at all. The same holds analogously for the transitions from Out^L to Out^L, Out^R and $\mathrm{Bg}^{\mathrm{out}}$. In all of these cases, Out^L would not be defined.

The transition from Out^L to Occ^R is forbidden since the occluding object on the left side must be closer, but a left phantom Stixel due to a Stereo tear-off edge is behind the following object.

Similarly, the transitions from Out^R to Out^R are excluded since a right phantom group

Transition from/to	O	Bg	Out^L	Out^R	Bg^{out}	Occ^L	Occ^R
O	A	A	O	B	A	C	A
Bg	A	O	X	B	O	C	A
Out^L	C	C	X	X	C	C	C
Out^R	O	X	X	X	X	X	O
Bg^{out}	A	O	X	B	O	C	A
Occ^L	A	A	O	B	A	C	A
Occ^R	B	B	X	B	B	X	B

Table 4.1: Geometric transitions between object classes. The transition "A" is modeled via Equation 4.52, "B" is given by Equation 4.53 with $\mu_{disp} \leq 0$, "C" is given by Equation 4.53 with $\mu_{disp} \geq 0$, "X" is forbidden and "O" is free since both classes belong to the same physical object.

needs a preceding object.
Finally the transitions from Occ^L to Out^R and to Occ^R are physically implausible. In the first case, the Occ^L object raises the expectation that there will be a subsequent object with smaller depth, but Out^R Stixels suggest to have greater depth. The second case does not make sense since two objects cannot mutually occlude.
Looking at the class transitions of Bg and Bg^{out} it can be noticed that both classes can replace each other. This symmetry can be exploited to exclude the Bg^{out} class from the dynamic programming step and to optimize subsequently for this class by replacing some Bg labels with Bg^{out}.

Segmentation prior

The segmentation prior $p\left(\mathbf{L}^t\right)$ is assumed to factorize into the product of pairwise cliques (l_{i-1}^t, l_i^t)

$$p\left(\mathbf{L}^t\right) \propto \prod_{i=2}^{N} p\left(l_{i-1}^t, l_i^t\right) = \prod_{i=2}^{N} p\left(l_i^t \mid l_{i-1}^t\right) \cdot p\left(l_{i-1}^t\right), \tag{4.55}$$

which needs to be normalized explicitly. However, since this normalization constant does not change the MAP solution it is ignored in the following. The term $p\left(l_{i-1}^t\right)$ modeling the global occurrence probability for each class was already introduced in Subsubsection 3.3.1. Furthermore, the pairwise term $p\left(l_i^t \mid l_{i-1}^t\right)$ models the transition probabilities between different objects, e.g. in order to take into account preferred class transitions. Nevertheless, its most important function is to restrict the transitions between objects and outliers to physical plausible configurations. These restrictions enable a scene description based on pairwise object interactions in the first place. This term is defined by means of Table 4.1, the forbidden transitions are marked with an "X", see Subsection 4.3.3 for details. Since no adequate ground truth was available for all classes, the corresponding values have to be estimated. In order to get consistent estimates, a very simple model is applied which describes each class transition via a binary variable Δl_i^t which takes on the value 0 if both Stixels belong to the same physical

$p\left(\xi_{out} \mid \Delta l_i^t\right)$	$\Delta l_i^t = 0$	$\Delta l_i^t = 1$
$\xi_{out} = 0$	0.9	0.6
$\xi_{out} = 1$	0.1	0.4

Table 4.2: Estimated probability table $p\left(\xi_{out} \mid \Delta l_i^t\right)$ which describes the outlier probability given the information that there is a new object or not.

object and one otherwise

$$\Delta l_i^t := \begin{cases} 0, & \text{if } l_i^t = l_{i-1}^t \\ 1, & \text{otherwise.} \end{cases} \tag{4.56}$$

Furthermore the binary variable ξ_{out} models the prior probability for outliers. ξ_{out} is 1 if l_i^t belongs to an outlier class like Out^L, Out^R or Bg^{out} and zero otherwise. This way, the transition $p\left(l_i^t \mid l_{i-1}^t\right)$ is categorized into

$$p\left(l_i^t \mid l_{i-1}^t\right) \approx \eta \cdot p\left(\Delta l_i^t, \xi_{out}\right) = \eta \cdot p\left(\xi_{out} \mid \Delta l_i^t\right) \cdot p\left(\Delta l_i^t\right). \tag{4.57}$$

The idea behind this concept is to reduce the effective number of parameters in the absence of ground truth data. Multiple transitions can be handled in the same way solely based on the two characteristics defined above. The estimated values for the outlier probability $p\left(\xi_{out} \mid \Delta l_i^t\right)$ are summarized in table 4.2. The object change probability $p\left(\Delta l_i^t\right)$ is estimated

$$p\left(\Delta l_i^t\right) := \begin{cases} 0.95, & \Delta l_i^t = 0 \\ 0.05, & \Delta l_i^t = 1. \end{cases} \tag{4.58}$$

The normalizer η from Equation 4.57, for example for $l_{i-1}^t = O_n$ is chosen

$$\begin{aligned} \eta^{-1} = & \underbrace{p\left(\xi_{out} = 0 \mid \Delta l_i^t = 0\right) \cdot p\left(\Delta l_i^t = 0\right)}_{\text{same object}} + \\ & \underbrace{p\left(\xi_{out} = 0 \mid \Delta l_i^t = 1\right) \cdot p\left(\Delta l_i^t = 1\right)}_{\text{bg object}} + \\ & \underbrace{p\left(\xi_{out} = 1 \mid \Delta l_i^t = 1\right) \cdot p\left(\Delta l_i^t = 1\right)}_{\text{left phantom}} + \\ & \underbrace{p\left(\xi_{out} = 1 \mid \Delta l_i^t = 0\right) \cdot p\left(\Delta l_i^t = 0\right)}_{\text{right phantom}} + \\ & \underbrace{p\left(\xi_{out} = 1 \mid \Delta l_i^t = 1\right) \cdot p\left(\Delta l_i^t = 1\right)}_{\text{bg phantom}} + \\ & \underbrace{p\left(\xi_{out} = 0 \mid \Delta l_i^t = 1\right) \cdot p\left(\Delta l_i^t = 1\right)}_{\text{left occluded}} + \\ & \underbrace{p\left(\xi_{out} = 0 \mid \Delta l_i^t = 1\right) \cdot p\left(\Delta l_i^t = 1\right)}_{\text{right occluded}}. \end{aligned} \tag{4.59}$$

to ensure that the probability is properly normalized, see Table 4.1. For the other classes, the forbidden transitions are excluded from the normalizer.

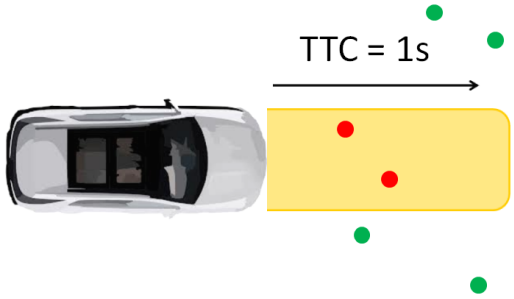


Figure 4.6: Stixel TTC violation constraint. Stationary Stixels shown in red inside the the predicted driving corridor shown in orange are unlikely. Outside the corridor, the learned prior 3.3.1 applies.

Optional Radar Prior If there is additional Radar object information available, the second term $p\left(l_{i-1}^{t}\right)$ defined in Equation 4.55 is adapted. The used Radar sensor (Continental ARS300 long range RADAR [Continental Automotive Industrial Sensors, 2011]) provides object hypotheses as discussed already in Subsection 3.4.2. Each object hypothesis Θ_r is described by its geometric object position

$$\vec{x}_r^t = \left(\mathcal{X}_r^t, 0, \mathcal{Z}_r^t\right)^T,\ r = 1... \left|\mathcal{R}^t\right|, \tag{4.60}$$

its object velocity $\vec{V}_r^t$

$$\vec{V}_r^t = \left(\dot{\mathcal{X}}_r^t, 0, \dot{\mathcal{Z}}_r^t\right)^T,\ r = 1... \left|\mathcal{R}^t\right| \tag{4.61}$$

and a width ${}^{o}\left|\Delta\mathcal{X}\right|_r^t$ defined in the object coordinate system. Further Radar-specific object information is not taken into account. This way,

$$p\left(l_{i-1}^t \mid \mathcal{R}^t\right) = \begin{cases} 1 - p_{out}, & \vec{x}_{i-1}^t \in \bigcup_r \mathrm{Corr}\left(\Theta_r\right) \wedge \\ & l_{i-1}^t \in \left\{O,\ \mathrm{Occ}^L,\ \mathrm{Occ}^R,\ \mathrm{Out}^L,\ \mathrm{Out}^R\right\}, \\ p_{out}, & \vec{x}_{i-1}^t \in \bigcup_r \mathrm{Corr}\left(\Theta_r\right) \wedge \\ & l_{i-1}^t \in \left\{\mathrm{Bg},\ \mathrm{Bg}^{\mathrm{out}}\right\}, \\ p\left(l_{i-1}^t\right), & \text{else}, \end{cases} \tag{4.62}$$

where is $\mathrm{Corr}\left(\Theta_r\right)$ denotes the predicted driving corridor of object r. The idea is visualized in Figure 4.6. Stationary Stixels inside the predicted driving corridor of a moving object are unlikely. The moving objects "clear" the path ahead.
In Equation 4.62 a simplified two-class model is used. All moving classes like O, Occ^L, Occ^R, Out^L or Out^R are treated as "moving", stationary classes like Bg or $\mathrm{Bg}^{\mathrm{out}}$ are summarized as stationary.

Temporal Consistency Term

As indicated in Section 3.2, the previous segmentation $\mathbf{L}^{t-1}$ constitutes an important prior for the current segmentation $\mathbf{L}^t$. The motion state of objects rarely changes, at most due to starting and braking. In so far it is quite obvious to couple segmentation results temporally.

The expression $p\left(\mathbf{L}^{t-1} \mid \mathcal{Z}^t, \mathbf{L}^t\right)$ takes into consideration the current Stixel observations $\mathcal{Z}^t$, besides the current segmentation $\mathbf{L}^t$. This is important in order to model an uncertainty in the old class decision. Note that this uncertainty is modeled in the classical discrete Bayes filter, see Chapter 2 of [Thrun et al., 2005]. However, direct application of this concept is not feasible since a probabilistic recursive class estimation would require to solve a high-dimensional Chapman-Kolmogorov integral [Papoulis, 1984] which is intractable in general. For that reason, a non-recursive formulation will be pursued. Under the assumption that the current segmentation $\mathbf{L}^t$ and the observations $\mathcal{Z}^t$ are complete for the old labeling $\mathbf{L}^{t-1}$, the probability $p\left(\mathbf{L}^{t-1} \mid \mathcal{Z}^t, \mathbf{L}^t\right)$ can be factorized into

$$
\begin{aligned}
p\left(\mathbf{L}^{t-1} \mid \mathcal{Z}^t, \mathbf{L}^t\right) &= p\left(l_1^{t-1}, ..., l_N^{t-1} \mid \mathcal{Z}^t, l_1^t, ..., l_N^t\right) \\
&= p\left(l_1^{t-1} \mid \mathcal{Z}^t, l_1^t, ..., l_N^t\right) \cdot p\left(l_2^{t-1} \mid \mathcal{Z}^t, l_1^{t-1}, l_1^t, ..., l_N^t\right) \cdot \\
&\quad p\left(l_3^{t-1} \mid \mathcal{Z}^t, l_1^{t-1}, l_2^{t-1}, l_1^t, ..., l_N^t\right) \cdot ... \cdot \\
&\quad p\left(l_N^{t-1} \mid \mathcal{Z}^t, l_1^{t-1}, l_2^{t-1}, ..., l_{N-1}^{t-1}, l_1^t, ..., l_N^t\right) \\
&\overset{*}{\approx} p\left(l_1^{t-1} \mid \mathcal{Z}^t, l_1^t, ..., l_N^t\right) \cdot p\left(l_2^{t-1} \mid \mathcal{Z}^t, l_1^t, ..., l_N^t\right) \cdot \\
&\quad p\left(l_3^{t-1} \mid \mathcal{Z}^t, l_1^t, ..., l_N^t\right) \cdot ... \cdot p\left(l_N^{t-1} \mid \mathcal{Z}^t, l_1^t, ..., l_N^t\right) \\
&\overset{**}{\approx} p\left(l_1^{t-1} \mid \vec{z}_1^t, l_1^t\right) \cdot p\left(l_2^{t-1} \mid \vec{z}_2^t, l_2^t\right) \cdot \\
&\quad p\left(l_3^{t-1} \mid \vec{z}_3^t, l_3^t\right) \cdot ... \cdot p\left(l_N^{t-1} \mid \vec{z}_N^t, l_N^t\right) \\
&\overset{***}{\approx} p\left(l_1^{t-1} \mid \Delta v_{res,1}^t, l_1^t\right) \cdot p\left(l_2^{t-1} \mid \Delta v_{res,2}^t, l_2^t\right) \cdot \\
&\quad p\left(l_3^{t-1} \mid \Delta v_{res,3}^t, l_3^t\right) \cdot ... \cdot p\left(l_N^{t-1} \mid \Delta v_{res,N}^t, l_N^t\right). \qquad (4.63)
\end{aligned}
$$

For the approximation marked with a star *, the aforementioned completeness assumption of $\mathbf{L}^t$ and $\mathcal{Z}^t$ has been used. This step ignores any spatial correlations for the old labeling $\mathbf{L}^{t-1}$ but it allows to break down the high-dimensional probability distribution.

Secondly, for the approximation marked with a double star **, it has been assumed that the old class of each Stixel just depends on the class choice of the corresponding Stixel in the current time step. This is another simplification for the sake of tractability. The correspondence between Stixels of consecutive images is determined via optical flow. This approximation holds as long as the optical flow is sufficiently accurate, that is it has at least Stixel accuracy. In this case, the global dependencies $p\left(l_i^{t-1} \mid \mathcal{Z}^t, l_1^t, ..., l_N^t\right) \approx p\left(l_i^{t-1} \mid \mathcal{Z}^t, l_i^t\right)$ can be simplified since not all possible transitions have to be taken into account.

Finally, the approximation marked with three stars takes into account the most evident source of class changes, namely braking and acceleration of objects. A global statistical temporal coupling such as $p\left(l_i^{t-1} \mid l_i^t\right)$ that ignores any potential observations bears the risk that a few mistakes in the segmentation might persist for a long time. The problem resides in the fact that the labeling uncertainty is not taken into account in the MAP solution. In order to alleviate this unwanted drawback, the velocity residual $\Delta v_{res,i}^t$

from the zero velocity hypothesis given by

$$\Delta v_{res,i}^t := \left(\dot{X}_i^t,\, \dot{Z}_i^t\right) \cdot \mathbf{\Sigma}_i^t \cdot \left(\dot{X}_i^t,\, \dot{Z}_i^t\right)^T \tag{4.64}$$

is taken into account. The idea is very simple: the higher the Stixel velocity, the more likely is a class change from stationary to moving since probably a previously standing object is starting. The velocity residual has proven to be significantly more stable than the acceleration for example. On the basis of these motion residuals, conclusions are drawn on the correctness of the old labeling $\mathbf{L}^{t-1}$. For this purpose, the binary variable ζ is introduced that is equal to zero if the old labeling l_i^{t-1} is probably not correct and equal to one otherwise

$$\zeta := \begin{cases} 1, & l_i^{t-1}\text{correct} \\ 0, & \text{else.} \end{cases} \tag{4.65}$$

This way Equation 4.63 becomes

$$\begin{aligned} p\left(l_i^{t-1} \mid \mathcal{Z}^t, l_i^t\right) &= \sum_{\zeta=0}^{1} p\left(l_i^{t-1}, \zeta \mid \Delta v_{res,i}^t, l_i^t\right) \\ &= \sum_{\zeta=0}^{1} p\left(l_i^{t-1} \mid \zeta, \Delta v_{res,i}^t, l_i^t\right) \cdot p\left(\zeta \mid \Delta v_{res,i}^t, l_i^t\right) \\ &\approx \sum_{\zeta=0}^{1} p\left(l_i^{t-1} \mid \zeta, l_i^t\right) \cdot p\left(\zeta \mid \Delta v_{res,i}^t, l_i^t\right). \end{aligned} \tag{4.66}$$

The transition probabilities $p\left(l_i^{t-1} \mid \zeta, l_i^t\right)$ are modeled

$$p\left(l_i^{t-1} \mid \zeta, l_i^t\right) := \begin{cases} \frac{1}{J}, & \text{if } \zeta = 0 \\ \hat{p}\left(l_i^{t-1} \mid l_i^t\right), & \text{otherwise.} \end{cases} \tag{4.67}$$

In this equation, again J denotes the cardinality of the object classes and $\hat{p}\left(l_i^{t-1} \mid l_i^t\right)$ are learned transition probabilities as specified in Table 3.1. $p\left(\zeta \mid \Delta v_{res,i}^t, l_i^t\right)$ denotes the temporal class confidence. For simplicity,

$$\begin{aligned} p\left(\zeta = 1 \mid \Delta v_{res,i}^t, l_i^t = 0\right) &= 1 - \exp\left(-\Delta v_{res,i}^t\right), \\ p\left(\zeta = 0 \mid \Delta v_{res,i}^t, l_i^t = 0\right) &= \exp\left(-\Delta v_{res,i}^t\right), \\ p\left(\zeta = 1 \mid \Delta v_{res,i}^t, l_i^t = Bg\right) &= \exp\left(-\Delta v_{res,i}^t\right), \\ p\left(\zeta = 0 \mid \Delta v_{res,i}^t, l_i^t = Bg\right) &= 1 - \exp\left(-\Delta v_{res,i}^t\right) \end{aligned} \tag{4.68}$$

is assumed. Depending on the confidence in the old class decision the temporal coupling is switched on or off. For $\zeta = 0$, the temporal coupling is switched off. The advantage of this model in comparison with Subsubsection 3.3.1 is its flexibility to better describe object braking and acceleration maneuvers. The resulting unary term is a temporal prior which explicitly helps to enforce temporally consistent labeling results.

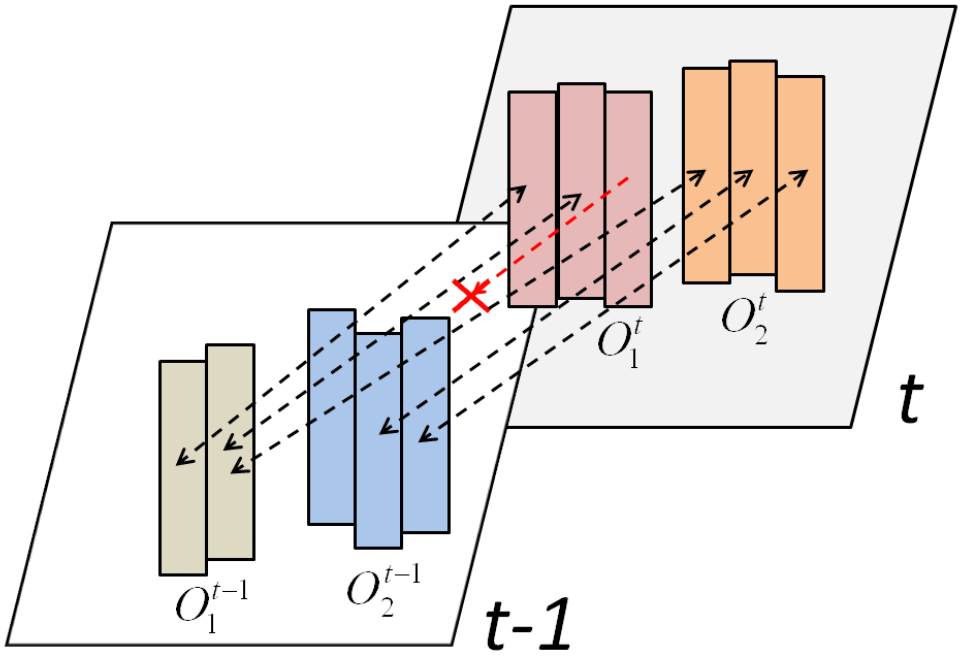


Figure 4.7: Temporal object ID transfer between frame $t-1$ and t. In both frames, two moving objects are found shown with different colors. Valid correspondences between Stixels are indicated by a black, dashed line. A red line indicates that the Stixel has no predecessor Stixel. The global Hamming distance is minimized via the assignments $\Delta\left(O_1^{t-1}, O_1^t\right) = 0$ and $\Delta\left(O_2^{t-1}, O_2^t\right) = 1$.

Note that the expression given in Equation 4.66 does not separate between different object instances but solely enforces labeling class consistency. So the introduced concept does not yield out of the box a temporally consistent object identification number (ID) for each object that is desirable for some applications.
To circumvent this drawback, the object ID is transferred from frame to frame via an object association step based on the Hungarian algorithm [Munkres, 1957]. The Stixel World finds the correspondences between Stixels from different time steps via optical flow measurements. Additionally, the Stixels have an object ID from the segmentation. The best match between different objects from different time steps can be found using the Hungarian method. This best match minimizes the Hamming distance between the objects. The Hamming distance between two objects $\Delta\left(O_n^{t-1}, O_{n'}^t\right)$ is the number of Stixels at which the object ID is different. See Figure 4.7 for a simple example.
Although the complexity of this algorithm scales very badly $\mathcal{O}\left(M^3\right)$, where M denotes the number of moving objects in both time steps, this association step typically does not play a role for the overall computing time due to the typical low number of objects. The performance and stability of this temporal coupling are investigated in 4.5.1.
The advantage of this class-specific coupling temporal coupling in comparison with the more object-specific coupling introduced in the previous Chapter 3 is that complexity

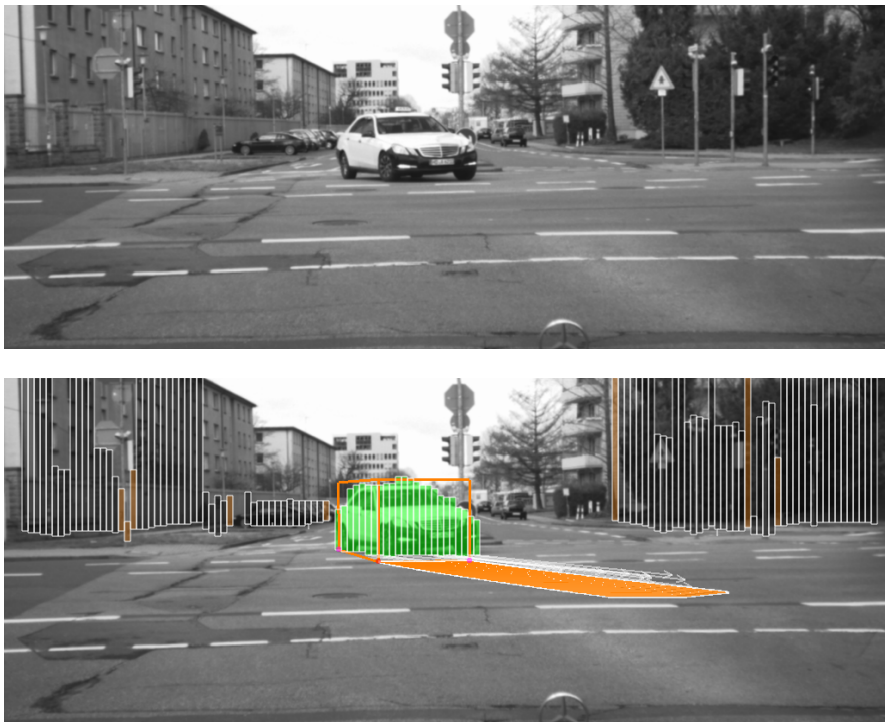


Figure 4.8: Object tracking initialization via the proposed Stixel segmentation. The orange bounding box shows the estimated object dimensions, the carpet on the ground shows the expected driving path within the next second.

does not rise as the number of moving objects rises. Furthermore, wrong segmentations from the past have less impact on the current segmentation since the strength of the coupling can be better controlled. The approach in Equation 3.3.1 finds a local optimum, starting from the old labeling result whereas this approach finds the global optimum taking into account the class decision like moving or stationary from the previous time step.
Note that the exact cut position between different moving objects is not coupled temporally by this approach. This can be advantageous or not. In practice, attempts to constrain the cut position often lead to over-segmentations since the Stixel measurements are noisy and temporally uncorrelated, see [Scharwaechter, 2012].
The segmentation results can be also used as noisy observations in a recursive Kalman filter-based object tracking scheme. This concept is introduced next.

4.4 Object Kalman Filtering

Optionally, the lateral movement of the segmented moving objects can be restricted to a circular path according to the well-known bicycle model [Zomotor, 1987, Barth and Franke, 2008]. This way, a full dynamic object model including the object acceleration and yaw rate can be estimated. The results of the segmentation act as noisy input data for a Kalman filter which basically combines the observations from different time steps

taking into account their uncertainties.
For the tracking, the object state defined by Θ_n^t from Equation 4.6 is augmented to contain

$$\tilde{\Theta}_n^t = \left[{}^e\mathcal{X}_n^t, {}^e\mathcal{Z}_n^t, {}^oX_{rot,n}^t, {}^oZ_{rot,n}^t, {}^o|\Delta\mathcal{X}|_n^t, {}^o|\Delta\mathcal{Z}|_n^t, \mathcal{H}_n^t, \psi^t, \|\vec{V}\|_n^t, \dot{\psi}_n^t, a_n^t \right]^T, \tag{4.69}$$

where ${}^e\mathcal{X}_n^t$ and ${}^e\mathcal{Z}_n^t$ denote the object reference point in the ego vehicle coordinate system that is assumed to be placed at the vehicle rear axis center, ${}^oX_{rot,n}^t$ and ${}^oZ_{rot,n}^t$ estimate the position of the object rotation point near the vehicle rear axis and ${}^o|\Delta\mathcal{X}|_n^t$, ${}^o|\Delta\mathcal{Z}|_n^t$ and $\mathcal{H}_n^t$ denote the object dimensions (width, length, height) in the respective object coordinate system. Finally, ψ_n^t and $\dot{\psi}_n^t$ are the vehicle gear angle and the object yaw rate and $\|\vec{V}\|_n^t$ and a_n^t are the vehicle velocity and acceleration [Barth and Franke, 2008].
The system model is given by the following set of differential equations

$$\Delta\tilde{\Theta}_n^t = \begin{bmatrix} \Delta\mathcal{X} \\ \Delta\mathcal{Z} \\ \Delta X_{rot} \\ \Delta Z_{rot} \\ \Delta\,|\Delta\mathcal{X}| \\ \Delta\,|\Delta\mathcal{Z}| \\ \Delta\mathcal{H} \\ \Delta\psi \\ \Delta\dot{\psi} \\ \Delta\|\vec{V}\| \\ \Delta a \end{bmatrix}_n^t = \begin{bmatrix} \mathbf{R}_{y,1}^{-1}(\psi) \cdot {}^o\tilde{\vec{x}}_{ref} \\ \mathbf{R}_{y,3}^{-1}(\psi) \cdot {}^o\tilde{\vec{x}}_{ref} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \dot{\psi}\Delta t \\ 0 \\ a\Delta t \\ 0 \end{bmatrix}, \tag{4.70}$$

where $\mathbf{R}_{y,i}(\psi)$ is the i-th row of the 3×3 rotation matrix around the height axis given by Equation 2.24,

$${}^o\tilde{\vec{x}}_{ref} = \mathbf{R}\left(\dot{\psi}\Delta t\right) \cdot ({}^o\vec{x}_{ref} - {}^o\vec{x}_{rot}) + {}^o\vec{x}_{rot} + {}^o\vec{T} \tag{4.71}$$

is the predicted position of the object reference point ${}^o\vec{x}_{ref} = (0,0,0)^T$ and

$${}^o\vec{T} = \begin{bmatrix} \int_0^{\Delta t} \left(\|\vec{V}\| + a \cdot t\right) \cdot \sin\left(\dot{\psi}t\right)\, \mathrm{d}t \\ 0 \\ \int_0^{\Delta t} \left(\|\vec{V}\| + a \cdot t\right) \cdot \cos\left(\dot{\psi}t\right)\, \mathrm{d}t \end{bmatrix} = \begin{bmatrix} \frac{\|\vec{V}\|}{\dot{\psi}} + \frac{a}{\dot{\psi}^2} \sin\left(\dot{\psi}\Delta t\right) - \frac{\|\vec{V}\| + a \cdot \Delta t}{\dot{\psi}} \cdot \cos\left(\dot{\psi}\Delta t\right) \\ 0 \\ \frac{\|\vec{V}\| + a \cdot \Delta t}{\dot{\psi}} \cdot \sin\left(\dot{\psi}\Delta t\right) + \frac{a}{\dot{\psi}^2} \cdot \left(\cos\left(\dot{\psi}\Delta t\right) - 1\right) \end{bmatrix} \tag{4.72}$$

is the translation vector in object coordinates [Barth and Franke, 2009].
The measurement equations are given by

$$\vec{z}^t = \begin{bmatrix} \mathcal{U}_n^{Ref,t} - u_0 \\ \mathcal{D}_n^{Ref,t} \\ {}^o|\Delta\mathcal{X}|_n^t \\ {}^o|\Delta\mathcal{Z}|_n^t \\ \mathcal{H}_n^t \\ \psi^t \\ \|\vec{V}\|_n^t \end{bmatrix} = \mathbf{H}^t \cdot {}^c\tilde{\Theta}_n^t, \tag{4.73}$$

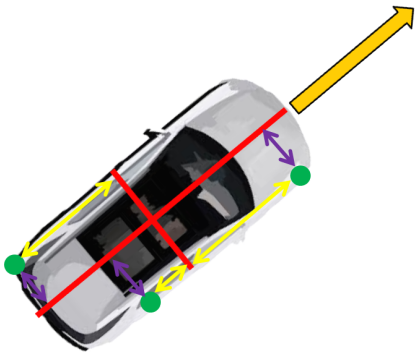


Figure 4.9: Object dimension estimation in the object coordinate system. The Stixel observations indicated by green dots are projected onto the direction of motion to determine the lateral dimension (shown in purple) and to a plane perpendicular to the motion direction to determine the longitudinal dimension (shown in yellow).

where the non-linear measurement matrix $\mathbf{H}^t$ (EKF) is given by

$$\mathbf{H}^t = \begin{pmatrix} f_x/{}^c\mathcal{Z}_n^t & & & & & & & & \\ & bf_x/{}^c\mathcal{Z}_n^{t\,2} & & & & & & & \\ & & 0 & & & & & & \\ & & & 0 & & & & & \\ & & & & 1 & & & & \\ & & & & & 1 & & & \\ & & & & & & 1 & & \\ & & & & & & & 1 & \\ & & & & & & & & 0 \\ & & & & & & & & & 1 \\ & & & & & & & & & & 0 \end{pmatrix}. \tag{4.74}$$

Note that the tracking is formulated in the ego coordinate system, however the observations are in image coordinates. Therefore

$${}^c\tilde{\Theta}_n^t = {}^c\mathbf{M}_e \cdot {}^e\tilde{\Theta}_n^t \tag{4.75}$$

where ${}^c\mathbf{M}_e$ is given by the extrinsic camera calibration matrix, see 4.39.
The reference point $\mathcal{U}_n^{Ref,t}$ and $\mathcal{D}_n^{Ref,t}$ are extracted from the segmentation. The only difficulty is to extract the object dimensions ${}^o|\Delta\mathcal{X}|_n^t$ and ${}^o|\Delta\mathcal{Z}|_n^t$ from the observations ψ^t via Equation 4.40 and from the observed dimensions $|\Delta\mathcal{U}|_n^t$ and $|\Delta\mathcal{D}|_n^t$, see Equation 4.6. For that reason, the associated Stixel observations are projected onto the direction of motion of the respective object and onto a plane perpendicular to it, see Figure 4.9 for a visualization. The maximum and minimum signed distance define the object dimensions. The remaining observations from Equation 4.73 can be directly observed. Note that this kind of converted measurement Kalman filter requires a bias removal for large distances [Sibley et al., 2007, Lerro and Bar-Shalom, 1993].

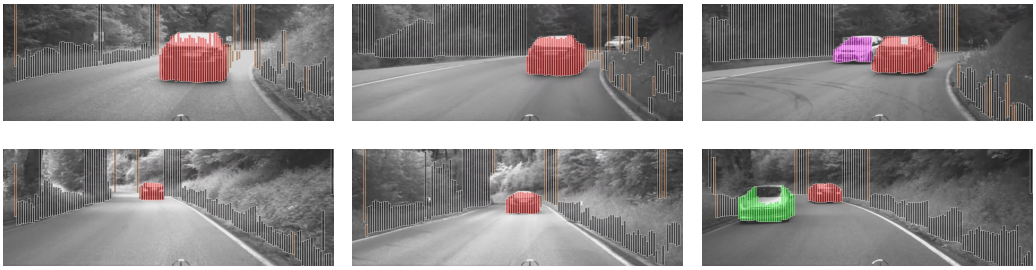


Figure 4.10: Selection of images from a real-world sequence used for the evaluation of motion accuracy estimation. For this purpose, the motion state of the leading vehicle is estimated using the Stixel segmentation and compared to its groundtruth recordings based on its inertial motion sensors. Especially in curves the importance of the outlier concept becomes evident.

4.5 Results

The following section focuses on a quantitative evaluation of the proposed object segmentation approach. In order to be able to rate the approach with respect to subsequent planning or tracking modules, it is necessary to provide statements regarding reliability, robustness and availability. However, since only limited groundtruth data was available for the experiments, the following evaluations focus on various aspects of the segmentation. Interesting here is especially an experimental comparison with the graphcut-based approach from Chapter 3 since both approaches work with the same input data.
This section is structured as follows. At first, an Adaptive Cruise Control (ACC [Winner et al., 2012]) like leading vehicle scenario with groundtruth information is investigated 4.5.1. This experiment rates the accuracy of the motion estimation step and the stability of the tracking.
Secondly, the detection rate of moving objects and stationary background is analyzed on the basis of groundtruth material 4.5.2. A comparison is drawn between the dynamic programming-based approach and the graphcut-based solution. These experiments rate the availability and false-alarm rates of both systems.
Thirdly, the approach is evaluated in the context of a higher-level planning module which takes into account the planned driving corridor of the ego vehicle. This way, the segmentation step is evaluated in the context of the whole processing chain. This kind of evaluation is of particular interest to autonomous driving. These results are summarized in Subsection 4.5.3.

4.5.1 Motion estimation ground truth

In order to rate the accuracy of the segmentation-based motion estimation, a real- world leading vehicle scenario is chosen.
In this scenario, a vehicle equipped with a stereo camera system follows another leading vehicle which records its own velocity and yaw rate measured by its inertial motion sensors, see Figure 4.10 for an impression of the scenario. The leading vehicle drives around serpentines and performs alternating acceleration and braking maneuvers to

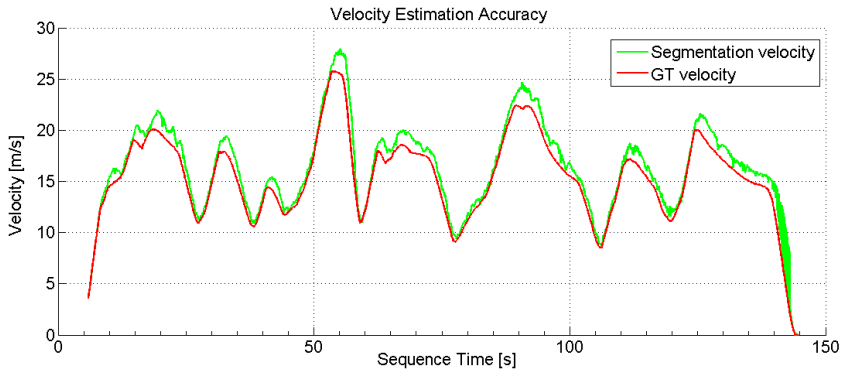


Figure 4.11: Direct comparison of the estimated velocity of the leading vehicle shown in green to the inertial motion sensor-based velocity measurements shown in red. The inertial motion sensor data was considered as ground truth in this evaluation.

cover a high dynamic range. The velocity measurements of the leading vehicle are considered as ground truth in the following. The following vehicle records the trajectory of the leading vehicle using a stereo camera system with a resolution of 1024×440 pixels and a base line of 23 cm. The duration of the evaluation is about 3500 pictures which corresponds to about $140\,s$. The segmentation results of the following vehicle are quantitatively compared to the ground truth data of the leading vehicle. The results are plotted in Figure 4.11. In this figure, absolute values for the velocities are plotted. Apparently, the curves coincide pretty well.
More precisely, the standard deviation with Bessel's correction defined by

$$\mathrm{std}\left(\vec{v}_{est} - \vec{v}_{gT}\right) := \sqrt{\frac{1}{T-1}\sum_{t=1}^{T}\left(v_{est,t} - \vec{v}_{gT,t}\right)^2} \tag{4.76}$$

between the estimated velocity measurements $\vec{v}_{est} = \{v_{est,t}, t = 1...T\}$ and the ground truth measurements $\vec{v}_{gT} = \{v_{gT,t}, t = 1...T\}$ is computed as a measure for the deviation of both quantities. The standard deviation is

$$\mathrm{std}\left(\vec{v}_{est} - \vec{v}_{gT}\right) = 0.72\ m/s. \tag{4.77}$$

The motion estimate $v_{est,t}$ is defined by those values $\left(\dot{\mathcal{X}}_n^t,\ \dot{\mathcal{Z}}_n^t\right)$ that maximize Equation 4.17.
A more detailed analysis of the error distribution is given in Figure 4.12. As can be seen from Figure 4.12, on average the segmentation-based motion estimation slightly overestimates the absolute value of the velocity. Especially at the end of the test drive, the Stixel filters are too slow to follow the strong braking of the leading vehicle. During that period, the largest deviation between the estimated velocity and the actual ground truth velocity

$$\max\left(\vec{v}_{est} - \vec{v}_{gT}\right) = 5.87\ m/s \tag{4.78}$$

is observed. The Stixel tracking assumes a constant velocity motion model which only adapts slowly to strong braking and acceleration maneuvers.

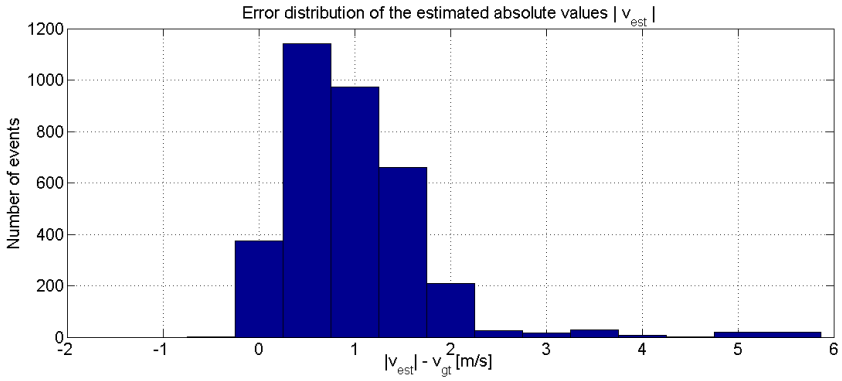


Figure 4.12: Error distribution of the estimated velocities $|\vec{v}_{est}| - \vec{v}_{gT}$.

Further error sources include ego-velocity errors which directly deteriorate the accuracy of the Stixel motion estimation step. In addition, a remaining referencing error between the inertial sensor data and the segmentation results might also downgrade the results. For referencing of the data, the start of the drive of the leading vehicle has been additionally indicated by an optical signal that can be identified by the following vehicle.
Nevertheless, on average the accuracy of motion estimation is pretty high and results are satisfactory.
Besides motion estimation accuracy, a stability analysis of the segmentation and the possible occurrence of phantom is of interest.
Stability is rated in terms of new object initializations. However, the tracked object did not need be reinitialized during the whole tracking sequence. This fact demonstrates the temporal consistency of the segmentation.
Finally, no additional phantom objects besides the leading vehicle were observed in this scenario. Any noise in the background is suppressed completely in this scenario by means of the applied regularization.
To sum up, the scenario under investigation has proven the performance of the proposed segmentation approach. The approach yields temporally consistent segmentation results, it can estimate the object velocities of other traffic participants with an accuracy greater than 1 m/s and it is insensitive to noise, that is it generates very little false-positive objects.

4.5.2 Labeling ground truth

Besides the accuracy analysis of the object velocity parameters $\dot{\mathcal{X}}_n$ and $\dot{\mathcal{Z}}_n$ presented in Subsection 4.5.1, a comparison with a manually labeled groundtruth is of special interest - particularly to deduce performance characteristics such as availability and false alarm rates. For this purpose, the same groundtruth data as presented for the graph-cut solution in Subsection 3.5.2 is considered. This data set contains the complete data from a test drive with a length of about one hour with about 80 000 images. Every 80th image has been manually labeled to provide groundtruth material as representative samples and to circumvent the strong correlation between neighboring frames. In this ground truth database, there are several (Stixel-wise) labeled moving objects in

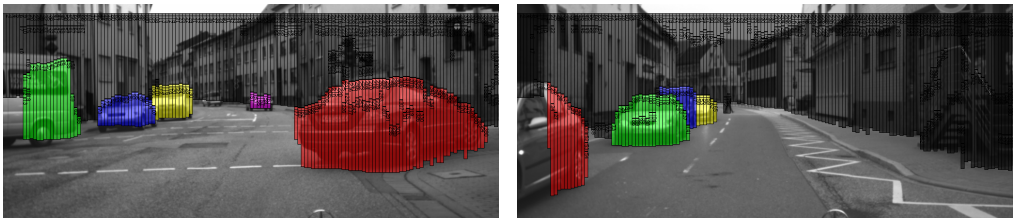
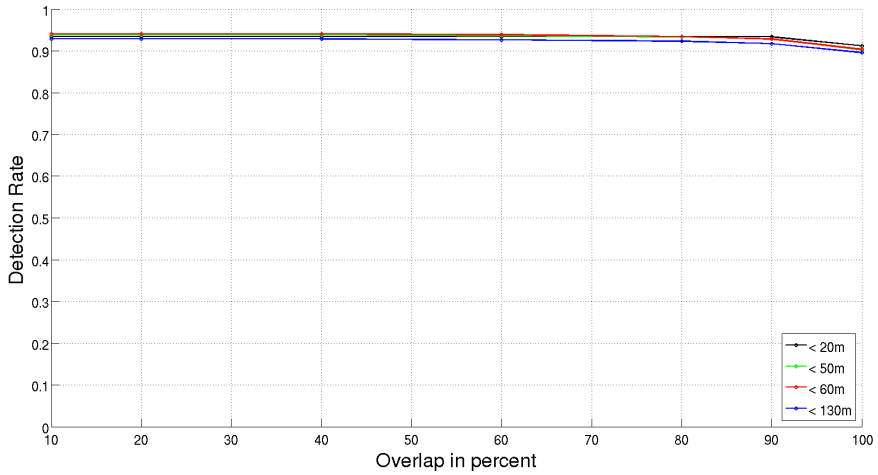
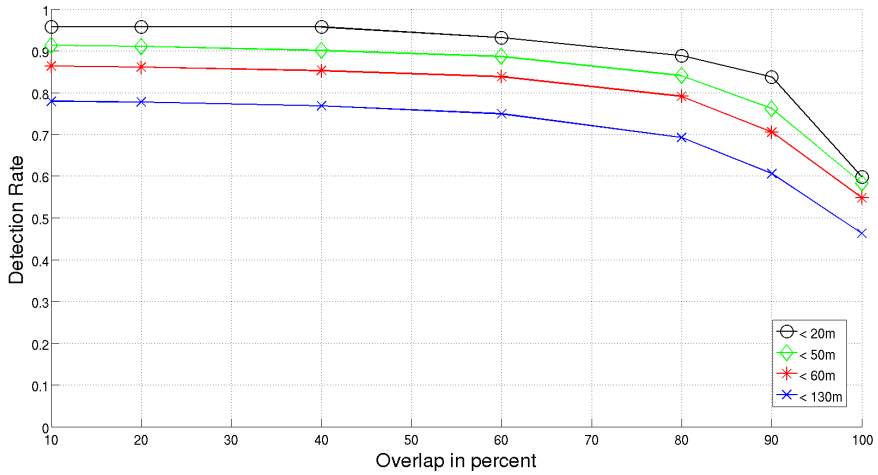


Figure 4.13: Sample images from the labeling groundtruth. The groundtruth contains manually Stixel-wise labeled traffic scene images. Stationary background is shown in black, different moving objects are shown with different colors.

addition to labeled stationary background. See Figure 4.13 for example images. The data set roughly consists half of rural roads and half of urban scenarios. Objects are included in the evaluation up to the detection limit of the Stixel World in the groundtruth (about 130 m), i.e. no moving objects are left out intentionally. Objects that are farther away cannot be resolved by the Stixel World. In this case, since these objects are not represented by the Stixels, they are counted as background. See Figure 4.13(a) for an example. A leading vehicle far away is not detected by the Stixel World, so it was labeled background. Experimental results are summarized in Figures 4.14 and 4.15.
Figure 4.14 shows the results of the Radar-assisted solution. The idea behind this concept has been explained above, see Paragraph 4.3.2 and 4.3.3. Figure 4.15 shows the results for the vision-only solution.
The most interesting curves are the blue curves, since they show the detection rate for moving objects for distances up to roughly 130 m. The other curves are a more detailed breakdowns for smaller distances.
In these figures, the detection rate is shown for various degrees of overlap. The x-axis specifies the required minimum overlap: objects are considered to be segmented correctly if they overlap more than $x\%$ with a labeled object. As stated in Subsection 3.5.2, the PASCAL criterion suggests using an overlap of 50% [Wojek et al., 2010, Everingham et al., 2010]. However, it is questionable whether this low overlap is sufficient in practice to preclude ambiguities and misinterpretations. Since the complete 3D object state vector of the moving objects is not available, a state difference to the groundtruth object state similar to [Schuhmacher et al., 2008] cannot be determined. A comparison in the image plane is the only comparison that is possible in this case. Nevertheless, together with the statement of false alarm rates, this kind of evaluation allows to draw conclusions about the performance of both approaches. The breakdown into various degrees of overlap circumvents the introduction of any unmotivated thresholding values that might obscure important information in the data.
It can be seen clearly from Figures 4.14 and 4.15 that the dynamic programming-based approach significantly outperforms the graphcut solution. In the following, a number of issues is discussed regarding these results.
First note that if requesting for a very high overlap ($\geq 90\%$), the detection rate drops significantly for the graphcut solution. This descent was already discussed in Subsection 3.5.2 and 3.6. In most cases, this decrease corresponds to - depending on the distance - one or two Stixels at the border of objects. Object borders are very unstable over

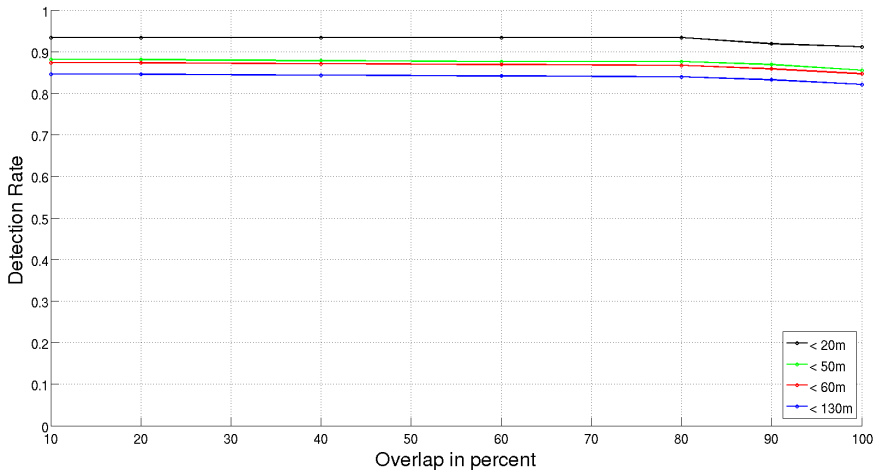


(a) Dynamic programming-based detection rate with Radar assistance.

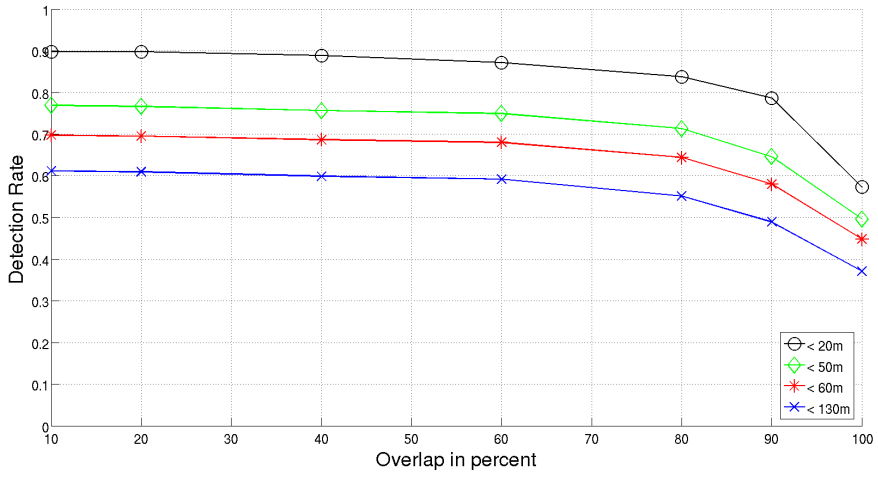


(b) Graphcut-based detection rate with Radar assistance.

Figure 4.14: Experimental comparison of dynamic programming (above) and graphcut-based methods (below) with Radar assistance.

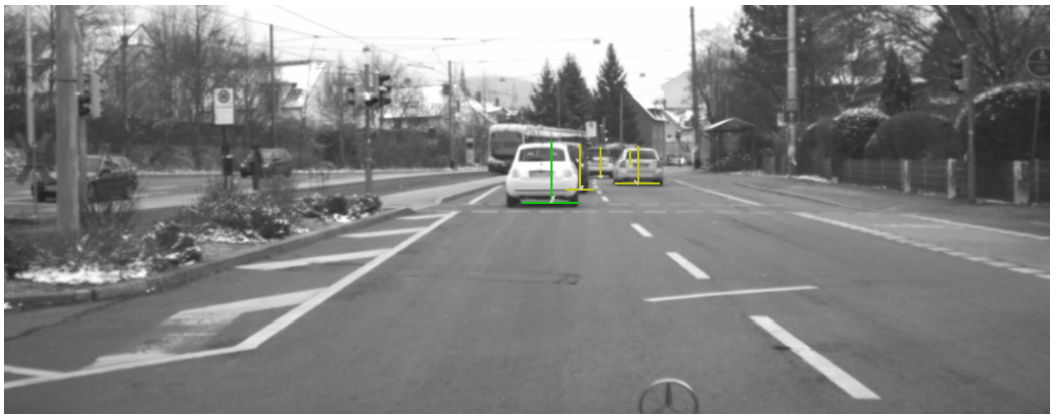


(a) Dynamic programming-based detection rate without Radar assistance.

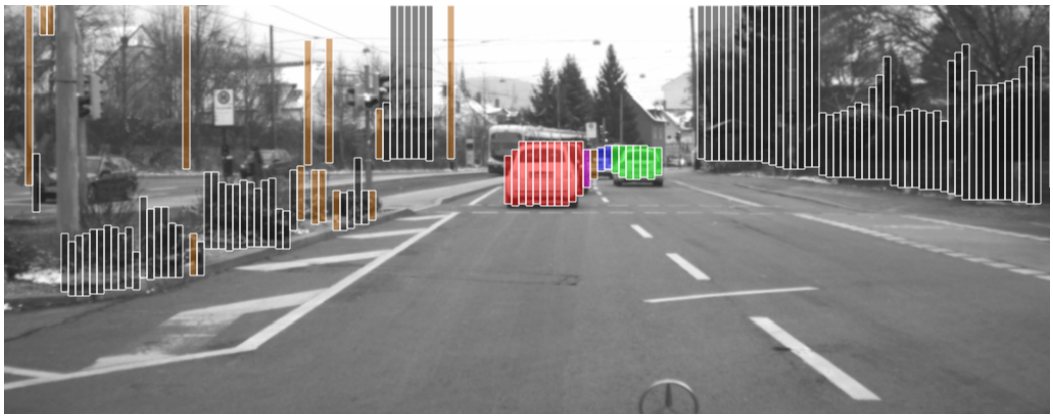


(b) Graphcut-based detection rate without Radar assistance.

Figure 4.15: Experimental comparison of dynamic programming (above) and graphcut-based methods (below) without Radar assistance.



(a) Traffic scene with four moving objects marked with radar targets.



(b) Segmentation result. Especially the separation between occlusions and outliers turns out to be extremely difficult. Note that the second object in the queue shown in magenta is covered by a single Stixel.

Figure 4.16: On the separation between occluded objects and outliers. Due to the low number of observations, object detection is challenging.

time, thus they are difficult to track and to segment due to stereo measurement errors and due to the fixed Stixel width. It therefore becomes clear that the proposed outlier classes are not superfluous, but they are essential for the used stereo system. Depending on the subsequent interpretation of the segmentation results, such errors can be fatal causing unwanted emergency brakings. The dynamic programming-based approach and especially the introduction of the outlier concept offer advantages since they take into account this heightened uncertainty at object borders. This step is important when striving for complete and highly accurate segmentation results.
Secondly, the non-submodular terms as object sizes and the occlusion modeling in the dynamic programming approach help substantially to avoid such errors at object borders because they massively constrain the possible solution space. It is clear that segmentation results as shown in Figure 4.16(b) are not possible without vehicle shape and dimension information. Disregarding the occlusion classes, the detection rate shown in Figure 4.14 and 4.15 drops significantly. The occluded state is important especially

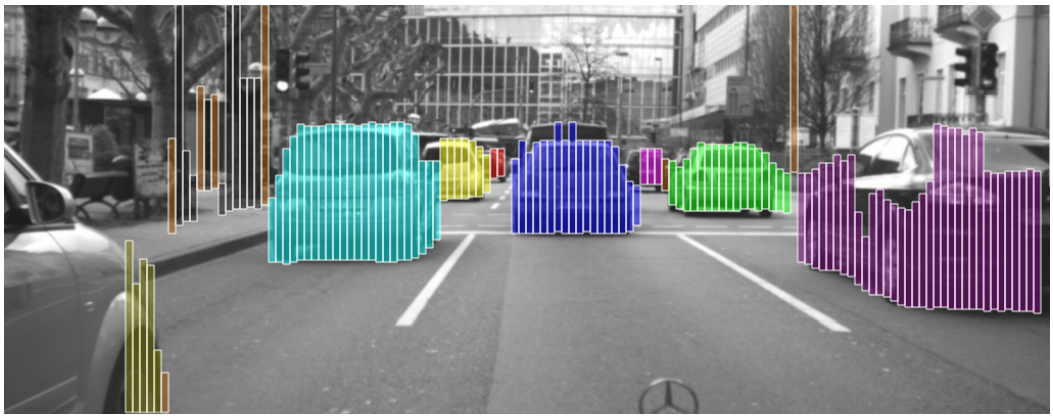


Figure 4.17: Example traffic scene with 8 moving vehicles, stationary background and brown phantom Stixels. Despite significant occlusions, the moving objects are segmented correctly.

motion source	gc	dp
with radar	99.2 %	98.2 %
without radar	99.6 %	97.3 %

Table 4.3: The correct labeled stationary background Stixel percentage for the graphcut segmentation (gc) and for the dynamic programming-based (dp) approach.

for crowded traffic scenes as shown in Figure 4.17. The occluded state was necessary for autonomous driving since misclassified objects lead to emergency brakings. Classifying really every Stixel is challenging as shown in Figure 4.16(b), but it is probably the only chance to interpret more complex scenes. However, it should be pointed out that the fixed Stixel width reaches its limits for situations as shown in Figure 4.16(b). The Stixel width limits the resolution of single objects. However, lowering this width reduces stability. In the future, a better understanding of the Stixel data and of their errors might help to achieve progress here.
On the other side, the dynamic programming approach produces slightly more false alarms, see Table 4.3. For the radar assisted solution, the phantom rate slightly increases by about 1 percent, the phantom rate for the vision-based solution increases by about 2 percent.
One reason for this is the introduction of the occluded object state. In many cases, occluded objects have no statistical significance in the sense of the BIC, see Equation 3.26. The reason for this is that in many cases there are simply not enough observations, see Figure 4.16(b). In this case, the requirements of the BIC are not fulfilled. In the future, it might be expedient to consult a part-based object detector for verification. The geometric viewpoint on occlusions is sometimes not sufficient.
Secondly, the more contextual reasoning based on object dimensions for larger distances can also be wrong in some situations.

Nevertheless, it should be pointed out that the increase in the phantom rate is minimal and is more than compensated by the significant increase in the detection rate. For larger distances up to 130 m (the blue curves in 4.14 and 4.15), the detection rate is increased by more than 20%. Note that the blue curve includes all distances up to 130 m, i.e. the increase for larger distances is even more pronounced. This enhancement is not possible without contextual knowledge.
Most remaining errors occur due to unmodeled Stixel tracking errors. There are several unmodeled sources of error in the tracking such as the uncertainty of the ego velocity, errors due to wrong filter adaptations or associations in the Stixel tracking, ignorance of confidences or unmodeled hidden correlations between observations. These effects are not captured by the Kalman filter covariance estimate. As a consequence, the actual uncertainties are unknown in most cases. These unmodeled effects should be a focus for future research.
Besides unmodeled tracking effects, a long-term temporal aggregation offers the opportunity to improve robustness significantly. Especially of interest here would be a combined state and existence estimation as offered by a PHD-filter [Vo and Ma, 2006] that would extend the Kalman filter-based object tracking proposed in Subsection 4.4.

4.5.3 Phantom evaluation

Robustness of segmentation results is essential in the context of safety-critical vision-based driver assistance. For that reason, in this subsection the proposed segmentation framework is evaluated with respect to the error behavior of a subsequent, higher-order planning module similar to [Schneider et al., 2012].
The planning module evaluates the planned ego vehicle driving corridor which is of uttermost importance for collision avoidance for autonomous driving. In order to know the planned driving corridor, the vehicle's driven path through the three-dimensional scene is reconstructed before the evaluation. This is achieved by looking ahead and integrating the vehicle's odometry information (velocity and yaw rate) from the recorded sequence meta-data [Schneider et al., 2012]. See Figure 4.19(a) for an example. The driving carpet shown on the ground illustrates the actual driven path of the ego vehicle. Since all sequences in the used database were recorded without any collisions, it is assumed that the Time To Collision (TTC) of all objects is always larger than 1 s. So if there is such a situation in which the predicted Stixel object position will collide with the predicted ego vehicle position within the next second, an error in the overall algorithmic processing chain is registered. Such an error might be due a mistake of the segmentation module which for example misclassifies the motion state of the Stixels or due to an error in the proceeding algorithms. This could be tracking errors of the Dynamic Stixel World, phantom obstacle Stixels inside the driving corridor of the Stixel World or ego motion estimation errors. Insofar the phantom evaluation presented in this subsection is a performance analysis of all algorithmic components simultaneously. In contrast to [Schneider et al., 2012], the driving corridor of the ego vehicle is taken into account for the collision prognosis. A Stixel with TTC = $1s$ will hit the predicted ego vehicle bounding box within one second. See Figure 4.18 for a visualization of this concept. In [Schneider et al., 2012], the TTC is computed based on a linear prediction of both the ego vehicle and of the Stixels. However, this simplification was found to be

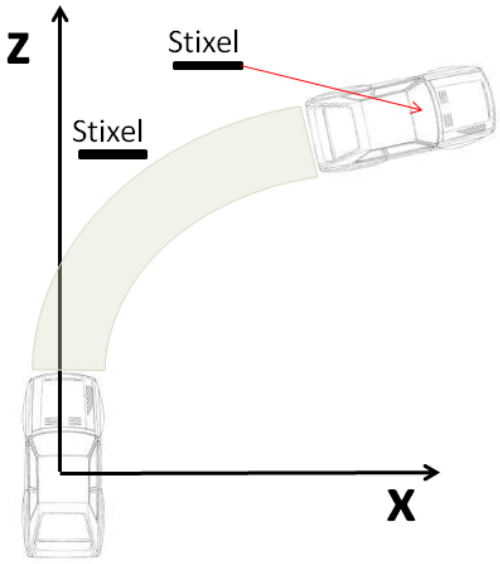
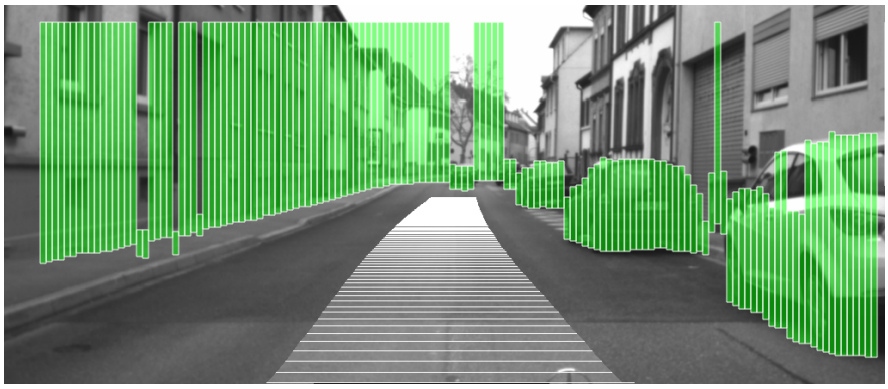
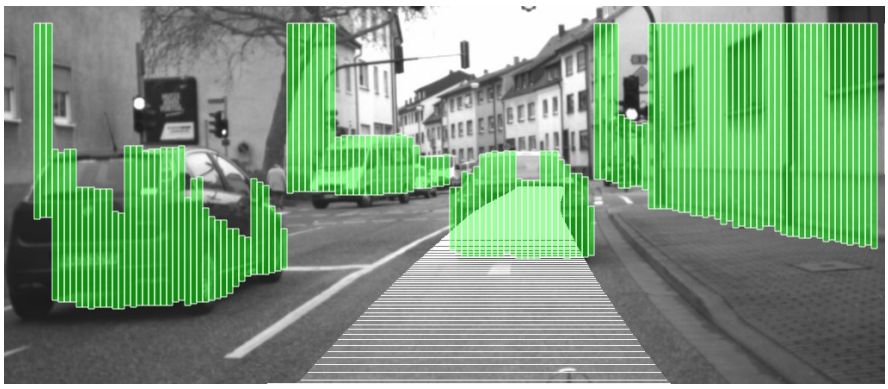


Figure 4.18: The TTC of each object Stixel is computed by predicting the ego vehicle according to the actual reconstructed driving corridor shown in gray and the linear prediction of the the Stixel position shown with a red arrow.

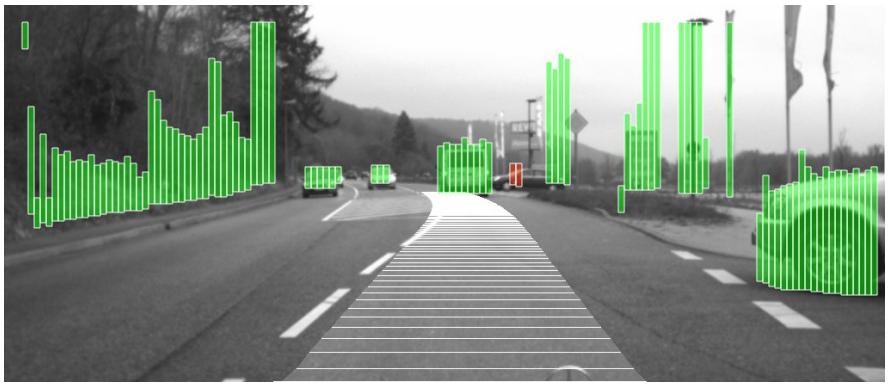
insufficient especially for curves.
The velocity of the ego vehicle is intentionally not taken from the driving corridor since the driver that was present during the recordings might have reacted to any potential obstacles inside the corridor. Especially with a view of autonomous driving, this treatment is not admissible. The autonomous vehicle cannot react to any obstacles inside the driving corridor that are not captured by the installed sensors. Instead, the rough assumption is made that the ego vehicle continues driving with the same speed. Of course, this is a crude approximation especially for curves where it is necessary to slow down. However, a complete planning module taking into account comfort, efficiency and safety is beyond the scope of the present work.
In order to be able to better assess the obtained results they are compared to the original Dynamic Stixel World [Pfeiffer and Franke, 2010], see Table 4.4 and Figure 4.20. This way, any possible deterioration due to the object segmentation gets clear and the results of the original Dynamic Stixel World serve as a baseline for comparison. As given in Table 4.4, the number of immediate collisions is reduced by a factor of 70 compared to the original Dynamic Stixel World. This reduction is significant. Figure 4.20 shows the complete TTC distribution of both approaches. For the histograms, the TTC range from TTC $= 0$ s to TTC $= 10$ s has been binned. Larger TTC values (TTC $> 10s$) are not shown since they would overshadow the whole histogram. Note that this reduction is significantly greater than the influence of different optical flow methods that has been investigated in [Schneider et al., 2012].
One remaining collision case is shown in Figure 4.19(c). In this case, a collision is wrongly predicted since the proposed system cannot assess the vigilance of the driver on the right side and it does not take into account higher-level prior information such as



(a) Stixel TTC visualization. The Stixel color encodes the expected TTC ranging from green (TTC $> 10s$) to red (TTC $= 0s$). The planned driving corridor is shown as a carpet on the ground. There are no imminent collisions in this scene.



(b) Another traffic scene with a leading vehicle inside the driving corridor. In this case the leading object has to predicted to rule out a collision.



(c) Remaining error case. A moving vehicle is approaching from the right. Due to the used linear motion model and due to the fact that lane markings are not taken into account, a collision is predicted erroneously.

Figure 4.19: Stixel TTC evaluation results as explained in the main text.

approach	# Stixels TTC $< 1s$ per frame	relative frequency
Dynamic Stixel World	0.306	6976 %
Stixel segmentation	0.004	100 %

Table 4.4: Comparison of the number of immediate collisions (TTC < 1) of presented Stixel segmentation and the Dynamic Stixel World. The number of immediate collisions is reduced by a factor of 70 by using the Stixel segmentation.

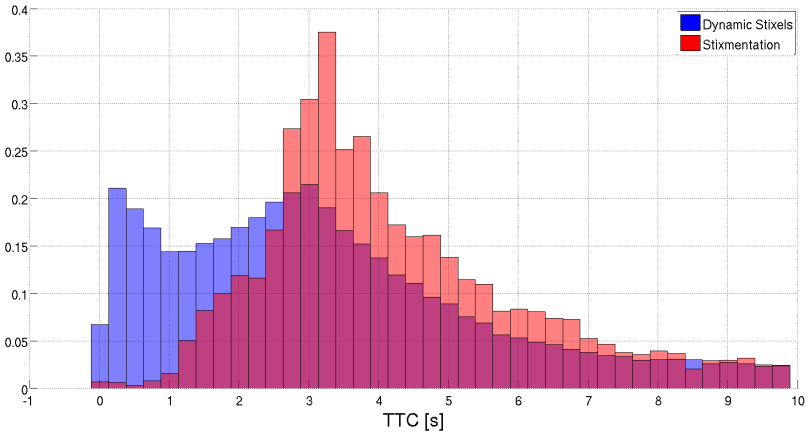


Figure 4.20: Comparison of the Stixel Time To Collision distribution $p(TTC)$ of the Stixel segmentation shown in red and the Dynamic Stixel World shown in blue. The number of immediate collisions (TTC < 1 s) is reduced significantly using the Stixel segmentation instead of the original Dynamic Stixel World. Larger TTC values (TTC $> 10s$) are not shown.

lane markings or the course of the road for example from digital maps. Such additional information might help to robustify the object segmentation. In Figure 4.19(c), a simple linear prediction foresees a collision since the object on the right-hand side has to brake. Further major error sources include phantom Stixels in the sky with a grossly wrong stereo information or phantom obstacles due to a wrong road profile estimation. In these cases, the Stixels might form a phantom obstacle in the planned driving corridor causing unwanted emergency brakings.

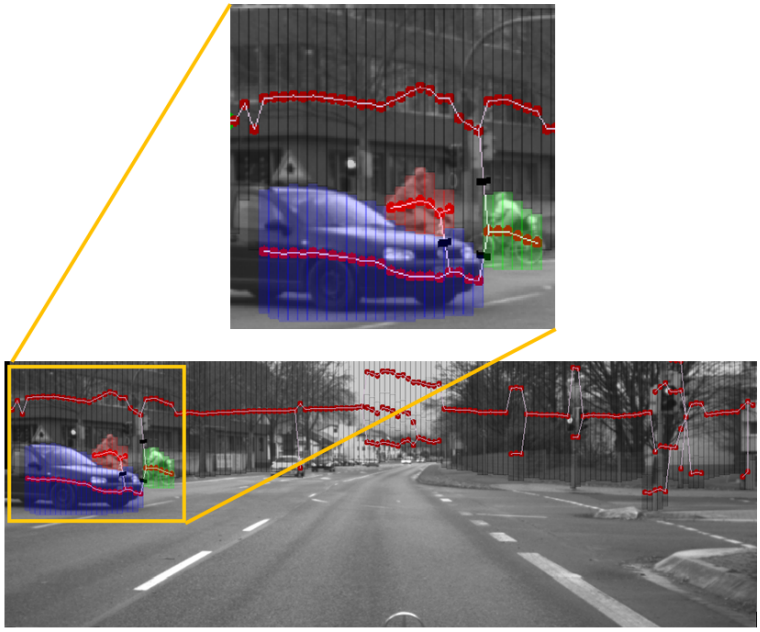
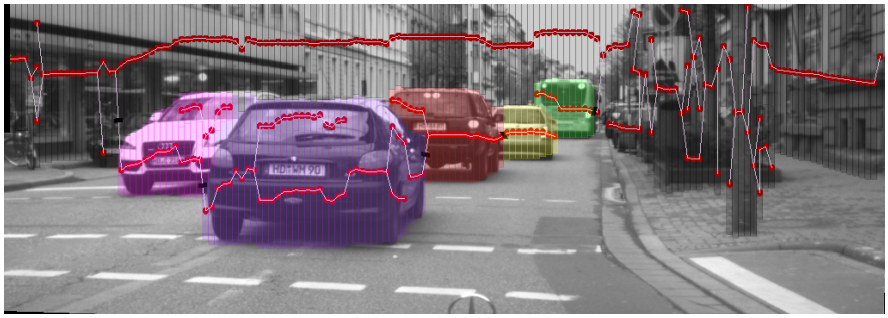


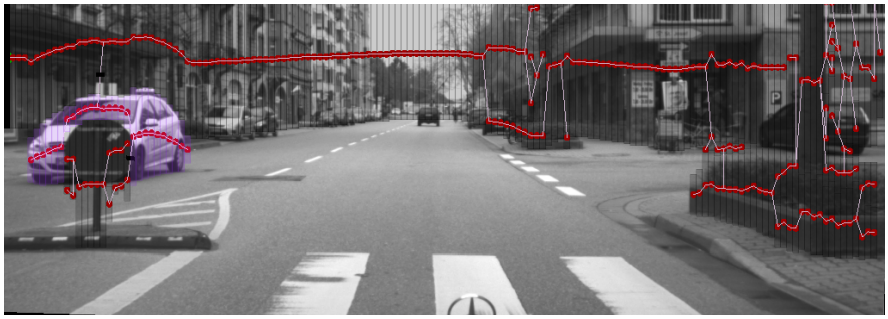
Figure 4.21: A possible extension of the dynamic programming concept to capture all Stixel rows. A tree is built from the Stixel World, connected Stixels are linked via white edges. This way, the segmentation is casted as tree cutting problem where the cuts are indicated by black lines between neighboring Stixels in the sense of the tree structure.

4.6 Outlook: Multi-layer Dynamic Programming

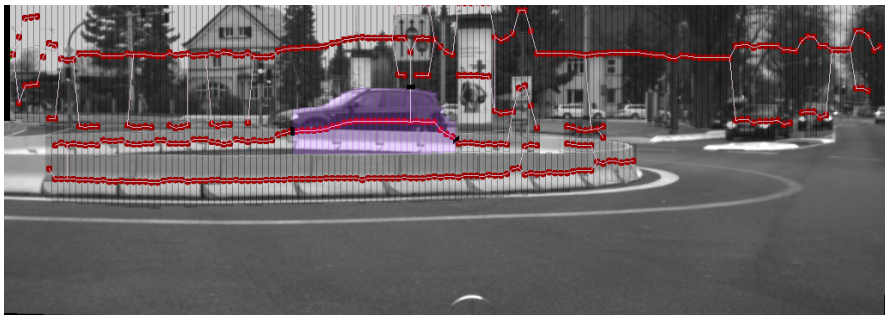
There are some object configurations that cannot be described completely solely relying on the first Stixel row information that corresponds to the closest obstacle along each line of sight. Consider Figure 4.21 for example. A bicyclist riding behind a crossing vehicle is missed by the first row Stixel World. Another example is given in Figure 4.22(c) where a crossing object is occluded by the roundabout infrastructure. In order to be able to address these scenarios, the Stixel segmentation has to be extended to multiple Stixel rows. For the graphcut-based solution presented in Chapter 3, this does not constitute a problem since the graphcut is out-of-the-box a 2D optimization technique. Dynamic programming, however, is a priori one-dimensional and cannot handle multiple Stixel rows efficiently. Nevertheless, dynamic programming is highly attractive due to the possible higher-order object regularization presented at the beginning of this chapter. In this section, a dynamic programming-based concept for multiple Stixel rows is discussed. The aim of this section is to show that the proposed dynamic programming approach can be generalized to the more complex 2D case.
To understand this generalization, consider the image section shown in Figure 4.21. In order to be able to apply dynamic programming, the underlying graphical model needs to be a tree. So instead of using the usual four-connected neighborhood, some edges have to be removed to obtain a tree. Typically, for N Stixels there are $2N$ edges when



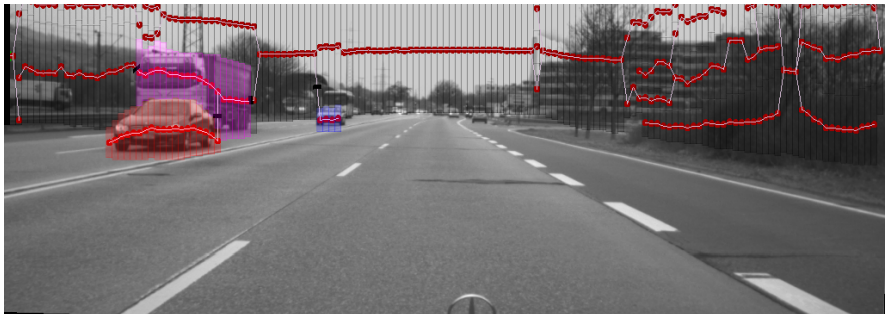
(a) Occlusion scenario of the left purple oncoming car and of the leading red object.



(b) Object occlusion due a closer traffic sign. The oncoming cannot be described completely solely relying on the first Stixel row information.



(c) Roundabout scenario where a crossing object is partially occluded by the roundabout structure.



(d) An oncoming truck that is not captured by the first row Stixels is occluded by a closer vehicle.

Figure 4.22: Example results of the tree-based dynamic programming.

considering a four-connected neighborhood but solely N edges in a tree. The question is which edges to remove best? Which edges carry the least information?
In order to answer this question, the original probability distribution $p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathcal{M}^t, \mathbf{L}^{t-1}\right)$ is relaxed to a different second-order, tree-structured distribution $q\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathcal{M}^t, \mathbf{L}^{t-1}\right)$ in the sense of a Chow-Liu tree [Chow and Liu, 1968, Kullback and Leibler, 1951]. The relaxed distribution $q\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathcal{M}^t, \mathbf{L}^{t-1}\right)$ has the minimum Kullback-Leibler distance defined by

$$D\left(p \parallel q\right) = -\sum_{\mathbf{L}^t} p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathcal{M}^t, \mathbf{L}^{t-1}\right) \log \frac{p\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathcal{M}^t, \mathbf{L}^{t-1}\right)}{q\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathcal{M}^t, \mathbf{L}^{t-1}\right)}. \tag{4.79}$$

In practice, computing any pairwise marginals is intractable. For that reason, $q\left(\mathbf{L}^t \mid \mathcal{Z}^t, \mathcal{M}^t, \mathbf{L}^{t-1}\right)$ is the minimum spanning tree found via Prim's algorithm [Prim, 1957] where the edge weight are defined based on the Stixel distances.
For most cases, this tree-relaxation is not a problem, see Figure 4.21 and 4.22 for example trees. It would be problematic if an object was not connected via the underlying tree. However, this case almost never occurs.
The idea to build a minimum spanning tree to define a Stixel neighborhood in the underlying graph makes sense as long as the Stixel sampling rate is sufficiently high, the object under consideration is spatially connected with a continuous contour, neighboring objects are spatially separated and the Stixel data is not dominated by noise. Under these conditions, the minimum spanning tree could also be regarded as a physical world model or a regularization (*connectivity prior*). For most cases these conditions are met. Choosing a tree structure offers several advantages in comparison with the first row approach presented before [Veksler, 2005].

- Keeping the **most important edges**. It is possible to keep those Stixel dependencies that carry most information by constructing the minimum weight spanning tree. This contrasts with the 1D approach that is limited to the horizontal edges only with no choice.
- For a tree, the labeling decision of each Stixel indirectly depends on the labeling decision of any other Stixel. This way, the optimal solution on a tree is a true **global optimal solution**.
- A tree keeps $g - 1$ **more edges** than a row-wise approximation where g is the number of Stixel rows.

Inference on trees has already been discussed in Section 2.6.1. For typical trees as shown in Figure 4.21 and 4.22, inference is intractable since complexity rises exponentially in the number of tree branches. For that reason, a *spider-approximation* is made, that is the maximum number of branch vertices is artificially limited to one by *pruning*, see Subsection 2.6.1. Usually, the spiders give a good and efficient object approximation, see Figure 4.21 and 4.22 for examples. This simplification lowers the the computational costs significantly. Nevertheless, the algorithm finds the global optimum for objects that can be represented by spiders.
Problems typically arise for close objects. In this case, the Stixel World often oversegments the close objects, especially for transparent panes. See Figure 4.22(a) for an

example. In this case, the object subtree is widened and inference would become more expensive. In this case, the spider-approximation just gives a local optimal solution. It will be part of future work to address oversegmentation at the Stixel level by defining a more appropriate object model.
In principle, the spider-based approach can be also used for other state-of-the-art super-pixel representations such as SLIC [Achanta et al., 2010] or turbopixels [Levinshtein et al., 2009]. Optimality of the approach, however, strongly depends on the compactness of the respective super-pixel representations and particularly on the number of branch vertices. Since complexity rises exponentially in the number of branch vertices, inference is only tractable for simple trees. Considering typical tree structures as shown in Figure 4.22, it is apparent that the Stixel tree representation has large linear chain-like tree structures without any branch vertices. Insofar, the Stixel world is particularly well-suited for the proposed spider-approximation.
First results show an average computing time of about 80 ms per frame but there is still enough room for accelerations, e.g. by efficient cost precomputations or parallelization. Besides this different graph construction step, the idea is exactly the same as presented in this chapter. Insofar, this approach is an extension of the first Stixel row concept. See [Erbs et al., 2014] and [Witte, 2013] for details and first results.

4.7 Conclusion

Is motion segmentation already today suited and efficient enough for driver assistance? The answer to this question from the introduction is affirmative. Motion information plays a key role for humans perceiving their environment [Funke and Frensch, 2006]. Insofar it is extremely important taking into account this information also for driver assistance systems.
The autonomous drive from Mannheim to Pforzheim has not least given a boost for the stereo camera sensor in the automotive environment. Until now the stereo camera and, in particular, motion information has played a minor role for previous autonomous driving projects such as the Google car [Thrun, 2010, Erico, 2013].
Now the first step has been taken and the stereo camera has established as an important full sensor for environment perception. It can be seen as a tremendous success that the autonomous drive was possible based on the proposed Stixel object segmentation.
Furthermore, the substantial reduction in the number of imminent collisions presented in Section 4.5.3 clearly shows the importance of object segmentation and of the proposed outlier concept for autonomous driving. An experimental comparison with the EM-like graphcut approach presented in the previous Chapter 3 clearly shows that object segmentation can be improved significantly by means of the higher-order object information that can be additionally taken into account by the dynamic programming-based approach.
Besides that, significant improvements were made with respect to partial occlusions and nearby moving objects, fundamental difficulties of segmentation and tracking [Barth, 2010]. Here it is most important to mention the introduction of a special occlusion class. The benefit of this object class is two-fold: firstly it takes into account object interactions, so it allows to use the maximum prior knowledge on occlusion scenarios. Occlusion is not considered as a perturbation of an ideal object model, but is taken into account explicitly. This way, powerful regularization terms such as the expected object dimensions can be specified more accurately and follow human reasoning.
Finally, in Section 4.6 an extension for taking into account multiple Stixel rows has been introduced. This approach is based on the setup of a tree that is supposed to capture the most important dependencies between individual Stixels. The proposed approach yields very promising first results. A so-called spider-approximation is proposed that is the basis for efficient inference. Unlike the much more complex junction tree algorithm [Korb and Nicholson, 2003, Jensen and Nielsen, 2007], the corresponding tree model is not inferred by introducing additional auxiliary nodes which speeds up the inference enormously. A remaining difficulty that has to be addressed in the future is the already discussed Stixel oversegmentation in the close range of the ego vehicle. It remains to be seen whether this goal can be achieved via parameter adaptations of the Stixels or whether extended object models or extended concepts for transparencies are required. Some occlusion scenarios as shown in Figure 4.22(b) cannot be handled by the first row approach as it is at the moment. In order to associate the right part of the object with its left part at least a second order Markov chain is required. Such higher-order models might also be part of future work.
The segmentation currently reaches its **limitations** for bad weather or illumination conditions where the optical flow fails completely. Furthermore, the approach is only

restrictedly suitable when considering a hardware-near implementation for example on microcontrollers due to its high complexity.

5 Conclusion and Future Work

In the present work, novel approaches for motion-based object segmentation on the basis of the Dynamic Stixel World representation were proposed.
The first approach segments the Dynamic Stixel World and estimates the relevant object parameters in an alternating expectation maximization sense. The labeling step is computed by means of the $\alpha-$expansion graphcut, the parameter estimation step via one-dimensional sampling and multidimensional gradient descent. Decoupling the parameter estimation from the labeling is a necessary prerequisite to achieve real-time capability. A lot of importance is placed on a sound probabilistic formulation that is often neglected in literature. The chosen probabilistic formulation allows to improve and stabilize object segmentation by incorporating spatio-temporal prior knowledge. Using the more compact and robust Stixel World instead of dense stereo and optical flow information makes it possible to compute the object segmentation within a few milliseconds on a single CPU core, thus enabling real-time operation in automobiles. To the knowledge of the author, there is currently no faster global optimization in a demonstrator car.
This first approach is used for comparison with a second approach based on dynamic programming. The key advantage of the dynamic programming approach is the fact that the object parameters and the Stixel labelings can be estimated simultaneously. This allows to take into account object knowledge explicitly in the optimization.
Besides the efficient optimization, two main challenges had to be resolved.
Firstly, the object segmentation needs to be able to deal with highly noisy and erroneous input data. Especially the presence of so called stereo tear-off edges has proven to be a difficult problem for a correct interpretation of the Stixel data. These faulty measurements are taken into account by means of a special outlier class which allows to include prior knowledge on their attributes and emergence. The outlier class turns out to be a mandatory necessity when striving for a complete and consistent interpretation of the Stixel data.
Secondly, occlusions are addressed explicitly. Especially in bustling cities occlusions constitute more the rule than the exception. A big advantage of the approach is the possibility to describe objects in their respective context rather than in isolation. Again, the introduction of a special occlusion class proves as an extremely important step towards complete and consistent scene interpretations and towards traffic scene understanding.
Both approaches, the alternating graphcut-based optimization and the dynamic programming-based approach, were compared and the results were discussed. Further experiments focus on complementary evaluation scenarios and investigate the accuracy and stability of the object segmentation - also with regard to the long-term goal of autonomous driving.
A main part of this work was the ongoing further development of the object segmen-

tation for the autonomous driving project Bertha Benz Memorial Drive. It can be considered a great success of this work that it was possible to drive the route from Mannheim to Pforzheim fully autonomously in real traffic and so to continue automotive history - the Bertha Benz drive 125 years ago can be seen as the hour of birth of the automobile. Although autonomous driving is still in its infancy and it is a long way to a possible market launch, hopefully the Bertha Benz project will be an important milestone on this path.
Finally, at the end of this thesis a short outlook is given on a possible extension of the dynamic programming-based object segmentation towards multiple Stixel rows.

Lessons Learned The performed evaluations of both approaches were focused on various selected real-world scenarios.
First, both approaches were compared by means of a labeling groundtruth for a long test drive with a duration of about one hour. Since the separation of moving objects and stationary background was the core task of the object segmentation for autonomous driving, this examination is of particular importance. It has been demonstrated that object segmentation can be improved significantly by means of higher-order object and scene knowledge, especially for large distances where input data is weak and objects are frequently partly occluded. Altogether, the obtained detection and false-alarm rates are impressive.
Secondly, the measurement accuracy and stability of object segmentation have been investigated in a leading vehicle scenario with groundtruth at hand. In this scenario, the proposed object segmentation has demonstrated both its outstanding stability against false-alarm objects and its high measurement accuracy with an average motion estimation error of less than 1 m/s.
Thirdly, the overall system has been tested extensively in the context of autonomous driving on the basis of a higher-level planning module which takes into account the planned driving corridor of the ego vehicle. It was demonstrated that by using the proposed object segmentation the number of false emergency braking maneuvers could be reduced significantly by a factor of 70 in comparison with the original Dynamic Stixel World. In this case, the numbers speak for themselves. This way, object segmentation could be established as an important building block for autonomous driving.
Summing up, object segmentation has taken a major step forwards during this work but it is not at its limits with its performance. The introduced outlier and occlusion concepts are valuable first steps in the right direction. However, the outlier concept could still be extended, in particular for bad illumination or weather conditions. Here the addressed tear-off edges just make up a small part of the occurring errors. A more comprehensive outlier concept that takes into account various confidence measures [Gehrig and Scharwachter, 2011, Hu and Mordohai, 2010, Pfeiffer et al., 2013] of previous algorithms is required under these conditions.
Moreover, it became clear in the experiments that object segmentation has significantly more degrees of freedom than conventional object tracking [Barth, 2010] and strongly depends on the underlying Stixel tracking. Accordingly, difficulties arise when the Stixel tracking fails, for example due to errors of the underlying road profile estimation. Currently, this dependency limits stability of the Stixel tracking and thus stability of object tracking. For that reason, the object Kalman Filtering step in Subsection 4.4

was proposed. Possibly this approach can further increase stability of the overall system. In this case, an extended object existence estimation [Gehrig et al., 2012] would be required, for example by means of PHD-filters [Vo and Ma, 2006], Dirichlet processes [Kooij et al., 2012] or multi-hypothesis tracking [Blackman, 2004].
Finally, detecting small and slow objects such as pedestrians or slow bicyclists has proven particularly difficult. Typically, these objects are difficult to detect mainly due to their low signal to noise ratio. In order to increase sensitivity of object segmentation especially in this low speed range, it might prove beneficial to take into account additional appearance information.

Outlook In order to improve beyond that what has been achieved so far, there are essentially two ways to move forward.
On the one hand, there is the possibility to take into account more and more scene information. Firstly, this might include the input from various classifier modules such as detectors for pedestrians or bicycles. Secondly, it might prove beneficial to consider the course of the road ahead via lane markings, curb reconstruction or map knowledge. Humans place object motion in context of the respective traffic scene specified by the street layout (intersection, T-junction, ...), lane markings, stationary infrastructure or simply based on empirical values. This wide field of *scene classification* [Ess et al., 2009, Geiger et al., 2011, Heracles et al., 2010] might help to better detect other objects and to predict their intended path.
On the other hand, it will be necessary to achieve a better understanding of the input data and of their uncertainties. This step is essential to extract the full information in the input data and forms the core of data analysis and model selection.
Finally, optimally taking into account all Stixel rows remains a challenging task. It is quite clear that in general inference has to search exhaustively over all possible solutions which is intractable. Insofar either a special solution structure has to be assumed as done in this work (spider-approximation) or the objective function needs to have certain properties such as submodularity. It definitely remains exciting to see how motion-based object segmentation will further develop in the future.

6 Appendix

6.1 Approximation of $Q\left(\mathcal{Z}^t, \Theta \mid \mathbf{L}^t\right)$

In this section, it is shown that a high-dimensional distribution $Q\left(\mathcal{Z}^t, \Theta \mid \mathbf{L}^t\right)$ can be approximated in the case of independent observations by a Gaussian distribution irrespective of the precise modeling of the underlying probability density. This observation underlies Laplace's method [Sivia, 1996].
The result is visualized in Figure 6.1. A complicated multimodal probability density function $p_1(\Theta) := \exp\left(f\left(\Theta\right)\right)$ with a maximum at $\Theta_{map} = -20$

$$\begin{aligned} p_1(\Theta \in [-50, 50]) =& 0.5 \cdot \mathcal{N}\left([-50, 50], 20, 4\right) + 0.3 \cdot \mathcal{N}\left([-50, 50], -20, 2\right) + \\ & 0.15 \cdot \mathcal{N}\left([-50, 50], 35, 5\right) + 0.05 \cdot \mathcal{N}\left([-50, 50], 0, 9\right) \end{aligned} \tag{6.1}$$

is shown for $N = 1$ in blue.
In the same Figure, $p_{50}(\Theta) = \exp\left(50 \cdot f\left(\Theta\right)\right)$ is shown in red, see Equation 6.2. It can be seen that the high dimensional probability density converges to an unimodal Gaussian distribution as the number of observations increases.

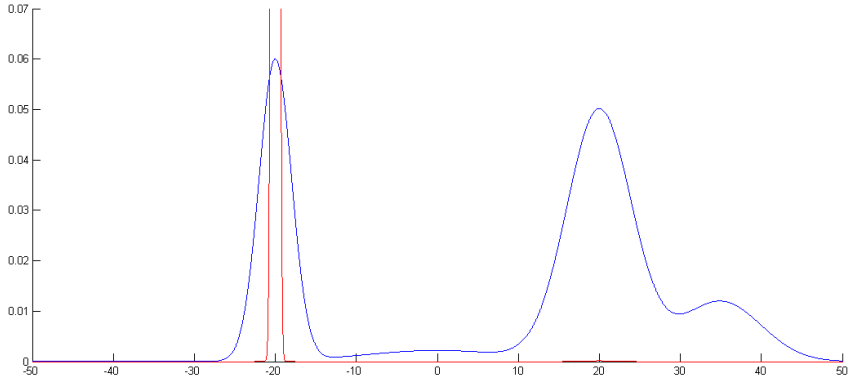


Figure 6.1: For an increasing number of observations, the probability distribution defined in Equation 6.2 better approximates a Gaussian distribution. In this Figure, the case for $N = 1$ is shown in blue and $N = 50$ in red.

Theorem 1 (Gaussian character of $Q\left(\mathcal{Z}^t, \Theta \mid \mathbf{L}^t\right)$)**.** *Assume that the probability* $Q\left(\mathcal{Z}^t, \Theta \mid \mathbf{L}^t\right)$ *is at least twice continuously differentiable on the interval* $[\Theta_{min}, \Theta_{max}]$ *and that it takes on its maximum value at* Θ_{map}*. Furthermore, assume that the observations in* $\mathcal{Z}^t$ *are independent and the number of observations* N *is large. In the case,*

$$\lim_{N\to\infty} \int_{\Theta_{min}}^{\Theta_{max}} Q\left(\mathcal{Z}^t, \Theta \mid \mathbf{L}^t\right) \mathrm{d}\Theta = \sqrt{\frac{2\pi}{\left|Q''\left(\mathcal{Z}^t, \Theta \mid \mathbf{L}^t\right)\right|}} \cdot Q\left(\mathcal{Z}^t, \Theta_{map} \mid \mathbf{L}^t\right).$$

Proof. The proof is done for the one-dimensional parameter case to keep to notation uncluttered. However the result can be readily generalized to an arbitrary dimension. By definition,

$$\begin{aligned}
Q\left(\mathcal{Z}^t, \Theta \mid \mathbf{L}^t\right) &= Q\left(\mathcal{Z}^t \mid \mathbf{L}^t, \Theta\right) Q\left(\Theta \mid \mathbf{L}^t\right) \\
&= \exp\left(-E\left(\mathcal{Z}^t \mid \mathbf{L}^t, \Theta\right) - E\left(\Theta \mid \mathbf{L}^t\right)\right) \\
&= \exp\left(\sum_{i=1}^{N}\left(-E\left(\vec{z}_i^t \mid \mathbf{L}^t, \Theta\right)\right) - E\left(\Theta \mid \mathbf{L}^t\right)\right) \\
&= \exp\left(N \cdot \left(\underbrace{-\bar{E}\left(\vec{z}_i^t \mid \mathbf{L}^t, \Theta\right) - \frac{1}{N} E\left(\Theta \mid \mathbf{L}^t\right)}_{f\left(\mathcal{Z}^t, \Theta, \mathbf{L}^t\right)}\right)\right) \\
&=: \exp\left(N \cdot f\left(\mathcal{Z}^t, \Theta, \mathbf{L}^t\right)\right)
\end{aligned} \tag{6.2}$$

holds where

$$E\left(\mathcal{Z}^t \mid \mathbf{L}^t, \Theta\right) := -\log Q\left(\mathcal{Z}^t \mid \mathbf{L}^t, \Theta\right), \tag{6.3}$$

$$E\left(\Theta \mid \mathbf{L}^t\right) := -\log Q\left(\Theta \mid \mathbf{L}^t\right) \text{ and} \tag{6.4}$$

$$\bar{E}\left(\vec{z}_i^t \mid \mathbf{L}^t, \Theta\right) := \frac{1}{N}\sum_{i=1}^{N}\left(E\left(\vec{z}_i^t \mid \mathbf{L}^t, \Theta\right)\right) \tag{6.5}$$

were defined.
$f\left(\mathcal{Z}^t, \Theta, \mathbf{L}^t\right)$ takes on its maximum value at $\Theta = \Theta_{map}$, accordingly the second derivative $f''\left(\Theta_{map}\right) < 0$.
In the following, the dependency of $f\left(\mathcal{Z}^t, \Theta, \mathbf{L}^t\right)$ from $\mathcal{Z}^t$ and $\mathbf{L}^t$ is omitted for the sake of better readability.
Let $\epsilon > 0$. It was assumed that $f''(\Theta)$ is continuous, so there exists $\delta > 0$ such that if $|\Theta - \Theta_{map}| < \delta$, then

$$f''(\Theta) \geq f''\left(\Theta_{map}\right) - \epsilon \tag{6.6}$$

holds.
Taylor's theorem implies that for $\Theta \in (\Theta_{map} - \delta, \Theta_{map} + \delta)$ there is $\xi \in (\Theta, \Theta_{map})$ such

that

$$\begin{aligned} f(\Theta) &= f(\Theta_{map}) + \frac{1}{2} f''(\xi)\left(\Theta - \Theta_{map}\right)^2 \\ &\overset{6.6}{\geq} f(\Theta_{map}) + \frac{1}{2}\left(f''(\Theta_{map}) - \epsilon\right)\left(\Theta - \Theta_{map}\right)^2. \end{aligned} \tag{6.7}$$

The first derivative has to vanish since Θ_{map} is a maximum by definition. So the integral in 6.2 can be bounded downwards exploiting the monotonicity of the exponential function

$$\begin{aligned} \int_{\Theta_{min}}^{\Theta_{max}} \underbrace{\exp\left(N \cdot f(\Theta)\right)}_{\geq 0} \mathrm{d}\Theta &\geq \int_{\Theta_{map}-\delta}^{\Theta_{map}+\delta} \exp\left(N \cdot f(\Theta)\right) \mathrm{d}\Theta \\ &\overset{6.7}{\geq} \exp\left(N \cdot f(\Theta_{map})\right) \cdot \\ &\int_{\Theta_{map}-\delta}^{\Theta_{map}+\delta} \exp\left(\frac{N}{2} \cdot \left(f''(\Theta_{map}) - \epsilon\right)\left(\Theta - \Theta_{map}\right)^2\right) \mathrm{d}\Theta. \end{aligned} \tag{6.8}$$

By substituting

$$\begin{aligned} u &:= \sqrt{N \cdot \left(\epsilon - f''(\Theta_{map})\right)} \cdot \left(\Theta - \Theta_{map}\right) \\ \rightarrow \frac{\mathrm{d}u}{\mathrm{d}\Theta} &= \sqrt{N \cdot \left(\epsilon - f''(\Theta_{map})\right)} \\ \rightarrow \mathrm{d}\Theta &= \frac{du}{\sqrt{N \cdot \left(\epsilon - f''(\Theta_{map})\right)}} \end{aligned} \tag{6.9}$$

the last integral in Equation 6.8 can be cast into a standard Gaussian integral as follows. Note that it is safe to take the square root in Equation 6.9 since $f''(\Theta_{map}) < 0$.

$$\begin{aligned} &\exp\left(N \cdot f(\Theta_{map})\right) \cdot \int_{\Theta_{map}-\delta}^{\Theta_{map}+\delta} \exp\left(\frac{N}{2} \cdot \left(f''(\Theta_{map}) - \epsilon\right)\left(\Theta - \Theta_{map}\right)^2\right) \mathrm{d}\Theta \\ =& \exp\left(N \cdot f(\Theta_{map})\right) \cdot \frac{1}{\sqrt{N \cdot \left(\epsilon - f''(\Theta)\right)}} \underbrace{\int_{-\sqrt{N \cdot (\epsilon - f''(\Theta_{map}))}\delta}^{\sqrt{N \cdot (\epsilon - f''(\Theta_{map}))}\delta} \exp\left(-\frac{1}{2}u^2\right) \mathrm{d}u}_{\rightarrow \sqrt{2\pi} \text{ for } N \rightarrow \infty} \\ \overset{N \rightarrow \infty}{\rightarrow}& \exp\left(N \cdot f(\Theta_{map})\right) \cdot \sqrt{\frac{2\pi}{N \cdot \left(\epsilon - f''(\Theta_{map})\right)}}. \end{aligned} \tag{6.10}$$

Summing up, it was shown that in the limit $N \rightarrow \infty$

$$\int_{\Theta} Q\left(\mathcal{Z}^t,\, \Theta \mid \mathbf{L}^t\right) \mathrm{d}\Theta \geq \sqrt{\frac{2\pi}{\left|\left(\log Q\left(\mathcal{Z}^t,\, \Theta \mid \mathbf{L}^t\right)\right)''\right|}} \cdot Q\left(\mathcal{Z}^t,\, \Theta_{map} \mid \mathbf{L}^t\right) \tag{6.11}$$

holds since $\epsilon > 0$ was arbitrary.
Now it is shown that the opposite is true as well, that is

$$\int_{\Theta} Q\left(\mathcal{Z}^t,\, \Theta \mid \mathbf{L}^t\right) \mathrm{d}\Theta \leq \sqrt{\frac{2\pi}{\left|\left(\log Q\left(\mathcal{Z}^t,\, \Theta \mid \mathbf{L}^t\right)\right)''\right|}} \cdot Q\left(\mathcal{Z}^t,\, \Theta_{map} \mid \mathbf{L}^t\right), \tag{6.12}$$

accordingly both expressions must be identical for $N \to \infty$.
To see this, note that since $f''(\Theta)$ is assumed to be continuous, there is $\eta > 0$ such that for $|\Theta - \Theta_{map}| < \eta$

$$f''(\Theta) \leq f''(\Theta_{map}) + \epsilon. \tag{6.13}$$

Accordingly, again Taylor's theorem ensures that for $|\Theta - \Theta_{map}| < \eta$ there is $\varrho \in (\Theta, \Theta_{map})$ such that

$$\begin{aligned} f(\Theta) &= f(\Theta_{map}) + \frac{1}{2} f''(\varrho) \cdot (\Theta - \Theta_{map})^2 \\ &\overset{6.13}{\leq} f(\Theta_{map}) + \frac{1}{2}\left(f''(\Theta_{map}) + \epsilon\right) \cdot (\Theta - \Theta_{map})^2 \\ &=: \tilde{f}(\Theta). \end{aligned} \tag{6.14}$$

Secondly, since Θ_{map} is the global maximum, there is $\zeta > 0$, such that for $|\Theta - \Theta_{map}| > \gamma$,

$$f(\Theta) \leq f(\Theta_{map}) - \zeta. \tag{6.15}$$

Consequentially, the integral in 6.2 can be bounded upwards

$$\begin{aligned} &\int_{\Theta_{min}}^{\Theta_{max}} \exp\left(N \cdot f(\Theta)\right) \mathrm{d}\Theta \\ &= \int_{\Theta_{min}}^{\Theta_{map}-\gamma} \underbrace{\exp\left(N \cdot f(\Theta)\right)}_{\substack{6.15 \\ \leq \exp(N\cdot(f(\Theta_{map})-\zeta))}} \mathrm{d}\Theta + \int_{\Theta_{map}-\gamma}^{\Theta_{map}+\gamma} \underbrace{\exp\left(N \cdot f(\Theta)\right)}_{\substack{6.14 \\ \leq \exp\left(N\cdot\tilde{f}(\Theta)\right)}} \mathrm{d}\Theta + \\ &\int_{\Theta_{map}+\gamma}^{\Theta_{max}} \underbrace{\exp\left(N \cdot f(\Theta)\right)}_{\substack{6.15 \\ \leq \exp(N\cdot(f(\Theta_{map})-\zeta))}} \mathrm{d}\Theta \\ &\leq \exp\left(N \cdot \left(f(\Theta_{map}) - \zeta\right)\right) \cdot \underbrace{\int_{\Theta_{min}}^{\Theta_{map}-\gamma} \mathrm{d}\Theta}_{\Theta_{map}-\gamma-\Theta_{min}} + \int_{\Theta_{map}-\gamma}^{\Theta_{map}+\gamma} \exp\left(N \cdot \tilde{f}(\Theta)\right) \mathrm{d}\Theta + \\ &\exp\left(N \cdot \left(f(\Theta_{map}) - \zeta\right)\right) \cdot \underbrace{\int_{\Theta_{map}+\gamma}^{\Theta_{max}} \mathrm{d}\Theta}_{\Theta_{max}-\Theta_{map}-\gamma} \\ &\leq (\Theta_{max} - \Theta_{min} - 2\gamma) \cdot \exp\left(N \cdot \left(f(\Theta_{map}) - \zeta\right)\right) + \int_{-\infty}^{\infty} \exp\left(N \cdot \tilde{f}(\Theta)\right) \mathrm{d}\Theta. \end{aligned} \tag{6.16}$$

By substituting

$$\begin{aligned} v &:= \sqrt{N \cdot \left(-\epsilon - f''\left(\Theta_{map}\right)\right)} \cdot \left(\Theta - \Theta_{map}\right) \\ \rightarrow \frac{\mathrm{d}v}{\mathrm{d}\Theta} &= \sqrt{N \cdot \left(-\epsilon - f''\left(\Theta_{map}\right)\right)} \\ \rightarrow \mathrm{d}\Theta &= \frac{dv}{\sqrt{N \cdot \left(-\epsilon - f''\left(\Theta_{map}\right)\right)}} \end{aligned} \tag{6.17}$$

again the integral is transformed to a standard Gaussian integral

$$\begin{aligned} &\int_{\Theta_{min}}^{\Theta_{max}} \exp\left(N \cdot f\left(\Theta\right)\right) \mathrm{d}\Theta \\ &\overset{6.16}{\leq} \left(\Theta_{max} - \Theta_{min} - 2\gamma\right) \cdot \exp\left(N \cdot \left(f\left(\Theta_{map}\right) - \zeta\right)\right) + \\ &\quad \frac{\exp\left(N \cdot f\left(\Theta_{map}\right)\right)}{\sqrt{N \cdot \left(-\epsilon - f''\left(\Theta_{map}\right)\right)}} \cdot \underbrace{\int_{-\infty}^{\infty} \exp\left(-\frac{1}{2}v^2\right) \mathrm{d}v}_{\sqrt{2\pi}} \\ &= \left(\Theta_{max} - \Theta_{min} - 2\gamma\right) \cdot \exp\left(N \cdot \left(f\left(\Theta_{map}\right) - \zeta\right)\right) + \\ &\quad \exp\left(N \cdot f\left(\Theta_{map}\right)\right) \cdot \sqrt{\frac{2\pi}{N \cdot \left(-\epsilon - f''\left(\Theta_{map}\right)\right)}} \\ &= \exp\left(N \cdot f\left(\Theta_{map}\right)\right) \cdot \sqrt{\frac{2\pi}{N \cdot \left(-f''\left(\Theta_{map}\right)\right)}} \\ &\quad \cdot \left(\underbrace{\sqrt{\frac{N \cdot \left(-f''\left(\Theta_{map}\right)\right)}{N \cdot \left(-\epsilon - f''\left(\Theta_{map}\right)\right)}}}_{\rightarrow 1 \text{ for } \epsilon \rightarrow 0} + \right. \\ &\quad \left. \left(\Theta_{max} - \Theta_{min} - 2\gamma\right) \cdot \underbrace{\exp\left(-N \cdot \zeta\right) \cdot \sqrt{\frac{N \cdot \left(-\epsilon - f''\left(\Theta_{map}\right)\right)}{2\pi}}}_{\rightarrow 0 \text{ for } N \rightarrow \infty} \right). \end{aligned} \tag{6.18}$$

To sum up, Equation 6.12 was proven and together with Equation 6.11,

$$\int_{\Theta} Q\left(\mathcal{Z}^t,\, \Theta \mid \mathbf{L}^t\right) \mathrm{d}\Theta = \sqrt{\frac{2\pi}{\left|\left(\log Q\left(\mathcal{Z}^t,\, \Theta \mid \mathbf{L}^t\right)\right)''\right|}} \cdot Q\left(\mathcal{Z}^t,\, \Theta_{map} \mid \mathbf{L}^t\right) \tag{6.19}$$

for $N \rightarrow \infty$. □

List of Figures

List of Tables

Bibliography

[Achanta et al., 2010] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. (2010). Slic superpixels. *École Polytechnique Fédéral de Lausssanne (EPFL), Tech. Rep*, 149300.

[Achanta et al., 2012] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Susstrunk, S. (2012). Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*.

[ADAC, 2013] ADAC (2013). Zahlen, Fakten, Wissen. Aktuelles aus dem Verkehr. `http://www.adac.de/_mmm/pdf/statistik_zahlen_fakten_wissen_0413_46600.pdf`.

[Alessandretti et al., 2007] Alessandretti, G., Broggi, A., and Cerri, P. (2007). Vehicle and Guard Rail Detection using Radar and Vision Data Fusion. *Intelligent Transportation Systems, IEEE Transactions on*, 8(1):95–105.

[Aloimonos and Swain, 1988] Aloimonos, J. and Swain, M. (1988). Shape from patterns: Regularization. *International journal of computer vision*, 2(2):171–187.

[Ashkin and Teller, 1943] Ashkin, J. and Teller, E. (1943). Statistics of two-dimensional lattices with four components. *Physical Review*, 64(5-6):178.

[Aster et al., 2013] Aster, R. C., Borchers, B., and Thurber, C. H. (2013). *Parameter estimation and inverse problems*. Academic Press.

[Bachmann, 2009] Bachmann, A. (2009). Applying recursive EM to scene segmentation. In *Pattern Recognition*, pages 512–521. Springer.

[Bachmann, 2010] Bachmann, A. (2010). *Dichte Objektsegmentierung in Stereobildfolgen*. KIT Scientific Publishing.

[Badino, 2007] Badino, H. (2007). A robust approach for ego-motion estimation using a mobile stereo platform. *Proceedings of the 1st International Conference on Complex Motion*, pages 198–208.

[Badino et al., 2009] Badino, H., Franke, U., and Pfeiffer, D. (2009). The stixel world-a compact medium level representation of the 3d-world. *Pattern Recognition*, pages 51–60.

[Barth, 2010] Barth, A. (2010). *Vehicle Tracking and Motion Estimation based on Stereo Image Sequences*. PhD thesis, University of Bonn, Bonn, Germany.

[Barth and Franke, 2008] Barth, A. and Franke, U. (2008). Where will the oncoming vehicle be the next second? In *Intelligent Vehicles Symposium, 2008 IEEE*, pages 1068–1073. IEEE.

[Barth and Franke, 2009] Barth, A. and Franke, U. (2009). Estimating the driving state of oncoming vehicles from a moving platform using stereo vision. *Intelligent Transportation Systems, IEEE Transactions on*, 10(4):560–571.

[Barth et al., 2010] Barth, A., Siegemund, J., Meißner, A., Franke, U., and Förstner, W. (2010). Probabilistic multi-class scene flow segmentation for traffic scenes. In *Proceedings of the 32nd DAGM conference on Pattern recognition*, pages 503–512. Springer-Verlag.

[Bellman, 1954] Bellman, R. (1954). The theory of dynamic programming. *Bulletin (New Series) of the American Mathematical Society*, 60(6):503–515.

[Benenson et al., 2012a] Benenson, R., Mathias, M., Timofte, R., and Van Gool, L. (2012a). Fast stixel computation for fast pedestrian detection. In *Computer Vision–ECCV 2012. Workshops and Demonstrations*, pages 11–20. Springer.

[Benenson et al., 2012b] Benenson, R., Mathias, M., Timofte, R., and Van Gool, L. (2012b). Pedestrian detection at 100 frames per second. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2903–2910. IEEE.

[Benenson et al., 2011] Benenson, R., Timofte, R., and Gool, L. V. (2011). Stixels estimation without depth map computation. *ICCV*.

[Bertele and Brioschi, 1972] Bertele, U. and Brioschi, F. (1972). *Nonserial dynamic programming*. Academic Press, Inc.

[Bertha Benz Memorial Club e.V., 2013] Bertha Benz Memorial Club e.V. (2013). Bertha Benz Memorial Route. `http://www.bertha-benz.de/`.

[Besag, 1986] Besag, J. (1986). On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 259–302.

[Beutelspacher et al., 2006] Beutelspacher, A., Schwenk, J., and Wolfenstetter, K.-D. (2006). *Moderne Verfahren der Kryptographie: von RSA zu Zero-Knowledge*. Springer DE.

[Bishop, 2007] Bishop, C. M. (2007). *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., New York, NY, USA.

[Blackman, 2004] Blackman, S. S. (2004). Multiple hypothesis tracking for multiple target tracking. *Aerospace and Electronic Systems Magazine, IEEE*, 19(1):5–18.

[Blake et al., 2011] Blake, A., Kohli, P., and Rother, C. (2011). *Markov random fields for vision and image processing*. MIT Press.

[Blake and Zisserman, 2012] Blake, A. and Zisserman, A. (2012). *Visual reconstruction*. MIT Press.

[Blakemore et al., 1990] Blakemore, C., Barlow, H. B., Weston-Smith, M., and Funds, R. P. (1990). *Images and understanding: thoughts about images, ideas about understanding*. Cambridge University Press.

[Bleyer and Gelautz, 2008] Bleyer, M. and Gelautz, M. (2008). Simple but effective tree structures for dynamic programming-based stereo matching. In *VISAPP (2)*, pages 415–422.

[Borgefors, 1986] Borgefors, G. (1986). Distance transformations in digital images. *Computer vision, graphics, and image processing*, 34(3):344–371.

[Boros and Hammer, 2002] Boros, E. and Hammer, P. L. (2002). Pseudo-boolean optimization. *Discrete applied mathematics*, 123(1):155–225.

[Boyd et al., 2007] Boyd, S., Xiao, L., Mutapcic, A., and Mattingley, J. (2007). Notes on decomposition methods. *Notes for EE364B, Stanford University*.

[Boykov and Kolmogorov, 2004] Boykov, Y. and Kolmogorov, V. (2004). An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(9):1124–1137.

[Boykov et al., 2001] Boykov, Y., Veksler, O., and Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239.

[Broggi, 1999] Broggi, A. (1999). *Automatic Vehicle Guidance: the Experience of the ARGO Autonomous Vehicle*. World Scientific Publishing Company Incorporated.

[Broggi et al., 2009] Broggi, A., Cattani, S., Medici, P., and Zani, P. (2009). Applications of Computer Vision to Vehicles: An Extreme Test. *Machine Learning for Computer Vision*, pages 215–250.

[Brostow et al., 2008] Brostow, G. J., Shotton, J., Fauqueur, J., and Cipolla, R. (2008). Segmentation and recognition using structure from motion point clouds. *ECCV*.

[Brostow et al., 2009] Brostow, G. J., Shotton, J., Fauqueur, J., and Cipolla, R. (2009). Combining appearance and structure from motion features for road scene understanding. *BMVC*.

[Brown et al., 2003] Brown, M., Burschka, D., and Hager, G. (2003). Advances in computational stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(8):993–1008.

[Buehler et al., 2009] Buehler, M., Iagnemma, K., and Singh, S. (2009). *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*, volume 56. Springer.

[Bundesamt für Statistik, 2009] Bundesamt für Statistik (2009). Unfallgeschehen im Straßenverkehr 2007, aktualisiert am 5.8. 2008. German road traffic accident statistics.

[Buzug, 2012] Buzug, T. (2012). Computed tomography. *Springer Handbook of Medical Technology*, pages 311–342.

[Černỳ, 1985] Černỳ, V. (1985). Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of optimization theory and applications*, 45(1):41–51.

[Chow and Liu, 1968] Chow, C. and Liu, C. (1968). Approximating discrete probability distributions with dependence trees. *Information Theory, IEEE Transactions on*, 14(3):462–467.

[Clifford, 1990] Clifford, P. (1990). Markov random fields in statistics. *Disorder in physical systems*, pages 19–32.

[Continental Automotive Industrial Sensors, 2011] Continental Automotive Industrial Sensors (July, 2011). ARS 300 Long Range Radar Sensor 77 GHz. `http://http://www.conti-online.com/www/industrial_sensors_de_en/themes/ars_300_en.html`.

[Cordts et al., 2014] Cordts, M., Schneider, L., Enzweiler, M., Franke, U., and Roth, S. (2014). Object-level priors for stixel generation. In *Pattern Recognition*, pages 172–183. Springer.

[Cormen et al., 2001] Cormen, T., Leiserson, C., Rivest, R., and Stein, C. (2001). *Introduction to algorithms*. MIT press.

[Crandall et al., 2012] Crandall, D. J., Fox, G. C., and Paden, J. D. (2012). Layer-finding in radar echograms using probabilistic graphical models. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 1530–1533. IEEE.

[Cremers et al., 2008] Cremers, D., Schmidt, F., and Barthel, F. (2008). Shape priors in variational image segmentation: Convexity, lipschitz continuity and globally optimal solutions. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–6.

[Cualain et al., 2007] Cualain, D., Glavin, M., Jones, E., and Denny, P. (2007). Distance detection systems for the automotive environment: a review. In *Irish Signals and Systems Conf.* Citeseer.

[Daimler AG, 2012] Daimler AG (2012). The Road to Accident-free Driving.

[Dal Mutto et al., 2012] Dal Mutto, C., Zanuttigh, P., Mattoccia, S., and Cortelazzo, G. (2012). Locally consistent tof and stereo data fusion. In *Computer Vision–ECCV 2012. Workshops and Demonstrations*, pages 598–607. Springer.

[Dang et al., 2015] Dang, T., Lauer, M., Bender, P., Schreiber, M., Ziegler, J., Franke, U., Fritz, H., Strauß, T., Lategahn, H., Keller, C. G., Erbs, F., et al. (2015). Autonomes fahren auf der historischen bertha-benz-route. *tm-Technisches Messen*, 82(5):280–297.

[Dantzig and Wolfe, 1960] Dantzig, G. B. and Wolfe, P. (1960). Decomposition principle for linear programs. *Operations research*, 8(1):101–111.

[Darbon, 2008] Darbon, J. (2008). Global optimization for first order markov random fields with submodular priors. *Combinatorial Image Analysis*, pages 229–237.

[Dawid et al., 2007] Dawid, P., Lauritzen, S. L., and Spiegelhalter, D. J. (2007). *Probabilistic networks and expert systems: Exact computational methods for Bayesian networks*. Springer.

[Delong et al., 2012] Delong, A., Osokin, A., Isack, H., and Boykov, Y. (2012). Fast approximate energy minimization with label costs. *International journal of computer vision*, 96(1):1–27.

[Dempster et al., 1977] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, B39(1)*, pages 1–38.

[Deng and Lin, 2006] Deng, Y. and Lin, X. (2006). A fast line segment based dense stereo algorithm using tree dynamic programming. In *Computer Vision–ECCV 2006*, pages 201–212. Springer.

[Dickmanns et al., 1994] Dickmanns, E. D., Behringer, R., Dickmanns, D., Hildebrandt, T., Maurer, M., Thomanek, F., and Schiehlen, J. (1994). The seeing passenger car 'VaMoRs-P'. In *Intelligent Vehicles' 94 Symposium, Proceedings of the*, pages 68–73. IEEE.

[Dinic, 1970] Dinic, E. A. (1970). Algorithm for solution of a problem of maximum flow in networks with power estimation. In *Soviet Math. Dokl*, volume 11, pages 1277–1280.

[E-Mags Media GmbH, 2013] E-Mags Media GmbH (September 10, 2013). Carl & Bertha Benz: Zwei Leben für einen großen Traum. `http://www.mercedes-fans.de/galerie/id=1127`.

[Elfes, 1987] Elfes, A. E. (1987). Sonar-based real-world mapping and navigation. *Journal of Robotics and Automation*, 3(3):249–265.

[ELROB, 2006] ELROB (2006). European Land-Robot Trial. `http://www.elrob.org/`.

[English, 2012] English, A. (20 November 2012). New car tech: 2014 mercedes-benz s-class. *Road & Track*.

[Enzweiler et al., 2012] Enzweiler, M., Hummel, M., Pfeiffer, D., and Franke, U. (2012). Efficient stixel-based object recognition. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 1066–1071. IEEE.

[Erbs et al., 2012a] Erbs, F., Schwarz, B., and Franke, U. (2012a). From stixels to objects - a conditional random field based approach. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 401–407. IEEE.

[Erbs et al., 2012b] Erbs, F., Schwarz, B., and Franke, U. (2012b). Stixmentation - probabilistic stixel based traffic scene labeling. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 71.1–71.12.

[Erbs et al., 2014] Erbs, F., Witte, A., Scharwaechter, T., Mester, R., and Franke, U. (2014). Spider-based stixel object segmentation. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 906–911. IEEE.

[Erico, 2013] Erico, G. (2013). How google's self-driving car works. *IEEE Spectrum*, 18.

[Ess et al., 2009] Ess, A., Müller, T., Grabner, H., and Van Gool, L. (2009). Segmentation-based urban Traffic Scene Understanding. In *Proceedings 20th British machine vision conference-BMVC 2009*.

[Everingham et al., 2010] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338.

[Ewing, 2013] Ewing, J. (16 May 2013). A benz with a virtual chauffeur. *The New York Times*.

[Fach and Ockel, 2009] Fach, M. and Ockel, D. (2009). Evaluation Methods for the Effectiveness of Active Safety Systems with respect to Real World Accident Analysis. In *International Technical Conference on the Enhanced Safety of Vehicles, Stuttgart*.

[Faerber, 2004] Faerber, G. (2004). Automobile Zukunft. *Seminarband, Technical University of Munich*.

[Felzenszwalb and Huttenlocher, 2006] Felzenszwalb, P. and Huttenlocher, D. (2006). Efficient belief propagation for early vision. *International journal of computer vision*, 70(1):41–54.

[Felzenszwalb and Huttenlocher, 2004] Felzenszwalb, P. F. and Huttenlocher, D. P. (2004). Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181.

[Felzenszwalb and Huttenlocher, 2012] Felzenszwalb, P. F. and Huttenlocher, D. P. (2012). Distance transforms of sampled functions. *Theory OF Computing*, 8:415–428.

[Felzenszwalb and Veksler, 2010] Felzenszwalb, P. F. and Veksler, O. (2010). Tiered scene labeling with dynamic programming. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3097–3104. IEEE.

[Felzenszwalb and Zabih, 2011] Felzenszwalb, P. F. and Zabih, R. (2011). Dynamic programming and graph algorithms in computer vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(4):721–740.

[Feng et al., 2010] Feng, W., Jia, J., and Liu, Z.-Q. (2010). Self-validated labeling of markov random fields for image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(10):1871–1887.

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

[Floros and Leibe, 2012] Floros, G. and Leibe, B. (2012). Joint 2D-3D temporally consistent Semantic Segmentation of Street Scenes. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2823–2830. IEEE.

[Ford and Fulkerson, 1962] Ford, L. R. and Fulkerson, D. R. (1962). Flows in networks. *Princeton University Press, Princeton, NJ*.

[Forsyth and Ponce, 2002] Forsyth, D. and Ponce, J. (2002). *Computer vision: a modern approach*. Prentice Hall Professional Technical Reference.

[Franke et al., 2014] Franke, U., Pfeiffer, D., Rabe, C., Knoeppel, C., Enzweiler, M., Stein, F., and Herrtwich, R. G. (2014). Making Bertha See. In *First International Workshop on Computer Vision for Autonomous Driving (CVAD)*. IEEE.

[Franke et al., 2005] Franke, U., Rabe, C., Badino, H., and Gehrig, S. (2005). 6d vision - fusion of motion and stereo for robust environment perception. *DAGM Symposium*.

[Freedman and Drineas, 2005] Freedman, D. and Drineas, P. (2005). Energy minimization via graph cuts: Settling what is possible. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 939–946. IEEE.

[Funke and Frensch, 2006] Funke, J. and Frensch, P. A. (2006). Handbuch der allgemeinen psychologie—. *Kognition. Göttingen ua*.

[Galliani et al., 2012] Galliani, S., Breuss, M., and Ju, Y. C. (2012). Fast and robust surface normal integration by a discrete eikonal equation. In *Proceedings of the British Machine Vision Conference*, pages 106.1–106.11. BMVA Press.

[Gallup et al., 2010] Gallup, D., Pollefeys, M., and Frahm, J. (2010). 3d reconstruction using an n-layer heightmap. *Pattern Recognition*, pages 1–10.

[Garey and Johnson, 1979] Garey, M. R. and Johnson, D. S. (1979). *Computers and intractability*, volume 174. Freeman New York.

[Gargano et al., 2002] Gargano, L., Hell, P., Stacho, L., and Vaccaro, U. (2002). Spanning trees with bounded number of branch vertices. In *Automata, Languages and Programming*, pages 355–365. Springer.

[Gavrila and Philomin, 1999] Gavrila, D. M. and Philomin, V. (1999). Real-time object detection for "smart" vehicles. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 87–93. IEEE.

[Gehrig et al., 2012] Gehrig, S., Barth, A., Schneider, N., and Siegemund, J. (2012). A multi-cue approach for stereo-based object confidence estimation. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 3055–3060. IEEE.

[Gehrig et al., 2009] Gehrig, S., Eberli, F., and Meyer, T. (2009). A real-time low-power stereo vision engine using semi-global matching. *ICCV*.

[Gehrig and Franke, 2007] Gehrig, S. and Franke, U. (2007). Improving stereo sub-pixel accuracy for long range stereo. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–7. IEEE.

[Gehrig and Rabe, 2010] Gehrig, S. and Rabe, C. (2010). Real-time semi-global matching on the cpu. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 85–92. IEEE.

[Gehrig and Scharwachter, 2011] Gehrig, S. K. and Scharwachter, T. (2011). A real-time multi-cue framework for determining optical flow confidence. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 1978–1985. IEEE.

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

[Geiger et al., 2013] Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*.

[Gharavi and Mills, 1990] Gharavi, H. and Mills, M. (1990). Blockmatching motion estimation algorithms-new results. *Circuits and Systems, IEEE Transactions on*, 37(5):649–651.

[Giachetti, 2000] Giachetti, A. (2000). Matching techniques to compute image motion. *Image and Vision Computing*, 18(3):247–260.

[Goldberg and Tarjan, 1988] Goldberg, A. V. and Tarjan, R. E. (1988). A new approach to the maximum-flow problem. *Journal of the ACM (JACM)*, 35(4):921–940.

[Goldberg and Holland, 1988] Goldberg, D. E. and Holland, J. H. (1988). Genetic algorithms and machine learning. *Machine learning*, 3(2):95–99.

[Goldstein, 2010] Goldstein, E. (2010). *Sensation and perception*. Wadsworth Publishing Company.

[Gould et al., 2009] Gould, S., Gao, T., and Koller, D. (2009). Region-based segmentation and object detection. NIPS.

[Gray, 1990] Gray, R. M. (1990). *Entropy and information theory*. Springer-Verlag New York, Inc., New York, NY, USA.

[Green, 1995] Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732.

[Gu et al., 2009] Gu, C., Lim, J. J., Arbeláez, P., and Malik, J. (2009). Recognition using regions. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1030–1037. IEEE.

[Guevara et al., 2012] Guevara, A., Conrad, C., and Mester, R. (2012). Curvature oriented clustering of sparse motion vector fields. In *Image Analysis and Interpretation (SSIAI), 2012 IEEE Southwest Symposium on*, pages 161–164. IEEE.

[Günyel et al., 2012] Günyel, B., Benenson, R., Timofte, R., and Gool, L. (2012). Stixels motion estimation without optical flow computation. 7577:528–539.

[Halin, 1976] Halin, R. (1976). S-functions for graphs. *Journal of Geometry*, 8(1-2):171–186.

[Haller and Nedevschi, 2010] Haller, I. and Nedevschi, S. (2010). Gpu optimization of the sgm stereo algorithm. In *Intelligent Computer Communication and Processing (ICCP), 2010 IEEE International Conference on*, pages 197–202. IEEE.

[Häming and Peters, 2010] Häming, K. and Peters, G. (2010). The structure-from-motion reconstruction pipeline–a survey with focus on short image sequences. *Kybernetika*, 46(5):926–937.

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

[Heath, 1998] Heath, M. (1998). Scientific computing. an introductory survey.

[Heinrich, 2005] Heinrich, S. (2005). Sensor-Fusion von Radar und Kamera mittels Kalmanfilter. In *CTI-conference, Stuttgart, Germany*.

[Heracles et al., 2010] Heracles, M., Martinelli, F., and Fritsch, J. (2010). Vision-based behavior prediction in urban traffic environments by scene categorization.

[Hermann and Klette, 2012] Hermann, S. and Klette, R. (2012). Iterative semi-global matching for robust driver assistance systems. In *ACCV*.

[Hillenbrand, 2011] Hillenbrand, M. (2011). *Funktionale Sicherheit nach ISO 26262 in der Konzeptphase der Entwicklung von Elektrik/Elektronik Architekturen von Fahrzeugen*, volume 4. KIT Scientific Publishing.

[Hirschmuller, 2005] Hirschmuller, H. (2005). Accurate and efficient stereo processing by semi-global matching and mutual information. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 807–814. IEEE.

[Hirschmuller and Gehrig, 2009] Hirschmuller, H. and Gehrig, S. (2009). Stereo matching in the presence of sub-pixel calibration errors. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 437–444. IEEE.

[Hirschmuller and Scharstein, 2009] Hirschmuller, H. and Scharstein, D. (2009). Evaluation of stereo matching costs on images with radiometric differences. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(9):1582–1599.

[Holland, 1975] Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence.* U Michigan Press.

[Homm et al., 2010] Homm, F., Kaempchen, N., Ota, J., and Burschka, D. (2010). Efficient occupancy grid computation on the gpu with lidar and radar for road boundary detection. *IEEE Intelligent Vehicles Symposium (IV)*, pages 1006–1013.

[Hong and Chen, 2004] Hong, L. and Chen, G. (2004). Segment-based stereo matching using graph cuts. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–74. IEEE.

[Horn and Brooks, 1986] Horn, B. and Brooks, M. (1986). The variational approach to shape from shading. *Computer Vision, Graphics, and Image Processing*, 33(2):174–208.

[Horn and Schunck, 1981] Horn, B. and Schunck, B. (1981). Determining optical flow. *Artificial intelligence*, 17(1):185–203.

[Hu and Mordohai, 2010] Hu, X. and Mordohai, P. (2010). Evaluation of stereo confidence indoors and outdoors. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1466–1473. IEEE.

[Huguet and Devernay, 2007] Huguet, F. and Devernay, F. (2007). A variational method for scene flow estimation from stereo sequences. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–7. IEEE.

[Hussmann et al., 2008] Hussmann, S., Ringbeck, T., and Hagebeuker, B. (2008). A performance review of 3d tof vision systems in comparison to stereo vision systems. *Stereo Vision*, pages 103–120.

[Ikeuchi, 1984] Ikeuchi, K. (1984). Shape from regular patterns. *Artificial Intelligence*, 22(1):49–75.

[Ingraham, 2013] Ingraham, N. (18 May 2013). Mercedes-benz shows off self-driving car technology in its new $ 100,000 s-class. *The Verge*.

[Institute of Control Engineering, 2007] Institute of Control Engineering, Institute of Flight Guidance, T. U. o. B. (2007). Stadtpilot. *http://stadtpilot.tu-bs.de/en/stadtpilot/*.

[Ishikawa, 2009] Ishikawa, H. (2009). Higher-order clique reduction in binary graph cut. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2993–3000. IEEE.

[Iu, 1995] Iu, S. (1995). Robust estimation of motion vector fields with discontinuity and occlusion using local outliers rejection. *Journal of Visual Communication and Image Representation*, 6(2):132–141.

[Ivănescu, 1965] Ivănescu, P. L. (1965). Some network flow problems solved with pseudo-boolean programming. *Operations Research*, 13(3):388–399.

[Jähne, 2005] Jähne, B. (2005). *Digitale Bildverarbeitung*. Springer.

[Jaynes, 1957] Jaynes, E. (1957). Information theory and statistical mechanics. *Physical review*, 106(4):620.

[Jensen and Nielsen, 2007] Jensen, F. V. and Nielsen, T. D. (2007). *Bayesian networks and decision graphs*. Springer.

[Kalender, 2011] Kalender, W. (2011). *Computed tomography*. Wiley-VCH.

[Kämpchen et al., 2012] Kämpchen, I. N., Aeberhard, M., Ardelt, M., and Rauch, S. (2012). Technologies for highly automated driving on highways. *ATZ worldwide*, 114(6):34–38.

[Kämpchen, 2007] Kämpchen, N. (2007). *Feature Level Fusion of Laser Scanner and Video Data for Advanced Driver Assistance Systems*. PhD thesis, University of Ulm.

[Khan and Shah, 2001] Khan, S. and Shah, M. (2001). Object based segmentation of video using color, motion and spatial information. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–746. IEEE.

[Kirkpatrick et al., 1983] Kirkpatrick, S., Jr., D. G., and Vecchi, M. P. (1983). Optimization by simmulated annealing. *science*, 220(4598):671–680.

[KIT, 2006] KIT (2006). AnnieWAY. `http://www.mrt.kit.edu/annieway/`. Department of Measurement and Control, Karlsruhe Institute of Technology.

[Kitt et al., 2010] Kitt, B., Moosmann, F., and Stiller, C. (2010). Moving on to dynamic environments: Visual odometry using feature classification. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 5551–5556. IEEE.

[Kittler and Föglein, 1984] Kittler, J. and Föglein, J. (1984). Contextual classification of multispectral pixel data. *Image and Vision Computing*, 2(1):13–29.

[Kohli and Kumar, 2010] Kohli, P. and Kumar, M. (2010). Energy minimization for linear envelope mrfs. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1863–1870. IEEE.

[Kohli et al., 2009] Kohli, P., Ladickỳ, L., and Torr, P. (2009). Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision*, 82(3):302–324.

[Kohli and Torr, 2006] Kohli, P. and Torr, P. H. (2006). Measuring uncertainty in graph cut solutions–efficiently computing min-marginal energies using dynamic graph cuts. In *Computer Vision–ECCV 2006*, pages 30–43. Springer.

[Koller et al., 2007] Koller, D., Friedman, N., Getoor, L., and Taskar, B. (2007). Graphical models in a nutshell. *Introduction to statistical relational learning*, page 13.

[Kolmogorov and Rother, 2007] Kolmogorov, V. and Rother, C. (2007). Minimizing nonsubmodular functions with graph cuts-a review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(7):1274–1279.

[Kolmogorov and Zabin, 2004] Kolmogorov, V. and Zabin, R. (2004). What energy functions can be minimized via graph cuts? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(2):147–159.

[Konrad and Dubois, 1991] Konrad, J. and Dubois, E. (1991). Comparison of stochastic and deterministic solution methods in bayesian estimation of 2d motion. *Image and Vision Computing*, 9(4):215–228.

[Kooij et al., 2012] Kooij, J. F., Englebienne, G., and Gavrila, D. M. (2012). A non-parametric hierarchical model to discover behavior dynamics from tracks. In *Computer Vision–ECCV 2012*, pages 270–283. Springer.

[Korb and Nicholson, 2003] Korb, K. B. and Nicholson, A. E. (2003). *Bayesian artificial intelligence*. CRC press.

[Kullback and Leibler, 1951] Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

[Kumar and Torr, 2008] Kumar, M. P. and Torr, P. H. (2008). Efficiently solving convex relaxations for map estimation. In *Proceedings of the 25th international conference on Machine learning*, pages 680–687. ACM.

[Ladicky et al., 2009] Ladicky, L., Russell, C., Kohli, P., and Torr, P. H. (2009). Associative hierarchical crfs for object class image segmentation. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 739–746. IEEE.

[Ladickỳ et al., 2010] Ladickỳ, L., Sturgess, P., Alahari, K., Russell, C., and Torr, P. (2010). What, where and how many? Combining Object Detectors and Crfs. *Computer Vision–ECCV 2010*, pages 424–437.

[Ladickỳ et al., 2012] Ladickỳ, L., Sturgess, P., Russell, C., Sengupta, S., Bastanlar, Y., Clocksin, W., and Torr, P. H. (2012). Joint optimization for object class segmentation and dense stereo reconstruction. *International Journal of Computer Vision*, pages 1–12.

[Lafferty et al., 2001] Lafferty, J., McCallum, A., and Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Machine Learning*, 951(2001):282–289.

[Lempitsky et al., 2012] Lempitsky, V., Blake, A., and Rother, C. (2012). Branch-and-mincut: global optimization for image segmentation with high-level priors. *Journal of Mathematical Imaging and Vision*, 44(3):315–329.

[Lempitsky et al., 2010] Lempitsky, V., Rother, C., Roth, S., and Blake, A. (2010). Fusion moves for markov random field optimization. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(8):1392–1405.

[Lenz et al., 2011] Lenz, P., Ziegler, J., Geiger, A., and Roser, M. (2011). Sparse scene flow segmentation for moving object detection in urban environments. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 926–932. IEEE.

[Lerro and Bar-Shalom, 1993] Lerro, D. and Bar-Shalom, Y. (1993). Tracking with debiased consistent converted measurements versus ekf. *Aerospace and Electronic Systems, IEEE Transactions on*, 29(3):1015–1022.

[Levinshtein et al., 2009] Levinshtein, A., Stere, A., Kutulakos, K. N., Fleet, D. J., Dickinson, S. J., and Siddiqi, K. (2009). Turbopixels: Fast superpixels using geometric flows. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(12):2290–2297.

[Li, 2009] Li, S. (2009). *Markov random field modeling in image analysis*. Springer.

[Lie et al., 2004] Lie, A., Tingvall, C., Krafft, M., and Kullgren, A. (2004). The effectiveness of esp (electronic stability program) in reducing real life accidents. *Traffic Injury Prevention*, 5(1):37–41.

[Liu and Zaccarin, 1993] Liu, B. and Zaccarin, A. (1993). New fast algorithms for the estimation of block motion vectors. *Circuits and Systems for Video Technology, IEEE Transactions on*, 3(2):148–157.

[Liu et al., 2011] Liu, C., Yuen, J., and Torralba, A. (2011). Nonparametric scene parsing via label transfer. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(12):2368–2382.

[Liu et al., 2008] Liu, F., Sparbert, J., and Stiller, C. (2008). IMMPDA Vehicle Tracking System using Asynchronous Sensor Fusion of Radar and Vision. In *Intelligent Vehicles Symposium, 2008 IEEE*, pages 168–173. IEEE.

[Lorenz, 1985] Lorenz, D. (1985). Das stereobild in wissenschaft und technik. *Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt, Köln, Oberpfaffenhofen*.

[Lucas et al., 1981] Lucas, B., Kanade, T., et al. (1981). An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th international joint conference on Artificial intelligence*.

[Meltzer et al., 2005] Meltzer, T., Yanover, C., and Weiss, Y. (2005). Globally optimal solutions for energy minimization in stereo vision using reweighted belief propagation. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 428–435. IEEE.

[Merrell et al., 2007] Merrell, P., Akbarzadeh, A., Wang, L., Mordohai, P., Frahm, J.-M., Yang, R., Nistér, D., and Pollefeys, M. (2007). Real-time visibility-based fusion of depth maps. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE.

[Mester, 2012] Mester, R. (2012). A bayesian view on matching and motion estimation. In *Image Analysis and Interpretation (SSIAI), 2012 IEEE Southwest Symposium on*, pages 197–200. IEEE.

[Micusik and Kosecka, 2009] Micusik, B. and Kosecka, J. (2009). Semantic Segmentation of street scenes by superpixel co-occurrence and 3d geometry. In *Computer*

Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on, pages 625–632. IEEE.

[Milella and Siegwart, 2006] Milella, A. and Siegwart, R. (2006). Stereo-based ego-motion estimation using pixel tracking and iterative closest point. In *Computer Vision Systems, 2006 ICVS'06. IEEE International Conference on*, pages 21–21. IEEE.

[Mobus and Kolbe, 2004] Mobus, R. and Kolbe, U. (2004). Multi-Target Multi-Object Tracking, Sensor Fusion of Radar and Infrared. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 732–737. IEEE.

[Montemerlo et al., 2008] Montemerlo, M., Becker, J., Bhat, S., Dahlkamp, H., Dolgov, D., Ettinger, S., Haehnel, D., Hilden, T., Hoffmann, G., Huhnke, B., et al. (2008). Junior: The Stanford Entry in the Urban Challenge. *Journal of Field Robotics*, 25(9):569–597.

[Moravec and Elfes, 1985] Moravec, H. P. and Elfes, A. E. (1985). High resolution maps from wide angle sonar. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 116–121.

[Moussouris, 1974] Moussouris, J. (1974). Gibbs and markov random systems with constraints. *Journal of statistical physics*, 10(1):11–33.

[Muffert et al., 2012] Muffert, M., Milbich, T., Pfeiffer, D., and Franke, U. (2012). May i enter the roundabout? a time-to-contact computation based on stereo-vision. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 565–570. IEEE.

[Mühlmann et al., 2002] Mühlmann, K., Maier, D., Hesser, J., and Männer, R. (2002). Calculating dense disparity maps from color stereo images, an efficient implementation. *International Journal of Computer Vision*, 47(1-3):79–88.

[Müller et al., 2011] Müller, T., Rabe, C., Rannacher, J., Franke, U., and Mester, R. (2011). Illumination-robust dense optical flow using census signatures. *Pattern Recognition*, pages 236–245.

[Munkres, 1957] Munkres, J. (1957). Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial & Applied Mathematics*, 5(1):32–38.

[Munz and Dietmayer, 2011] Munz, M. and Dietmayer, K. (2011). Using dempster-shafer-based modeling of object existence evidence in sensor fusion systems for advanced driver assistance systems. *IEEE Intelligent Vehicles Symposium (IV)*, pages 776–781.

[Ohta and Kanade, 1985] Ohta, Y. and Kanade, T. (1985). Stereo by intra-and inter-scanline search using dynamic programming. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 7(2):139–154.

[Orlin, 2009] Orlin, J. B. (2009). A faster strongly polynomial time algorithm for submodular function minimization. *Mathematical Programming*, 118(2):237–251.

[Papoulis, 1984] Papoulis, A. (1984). Probability, random variables, and stochastic processes.

[Park et al., 2011] Park, J., Kim, H., Tai, Y., Brown, M., and Kweon, I. (2011). High quality depth map upsampling for 3d-tof cameras. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1623–1630. IEEE.

[Pearl, 1984] Pearl, J. (1984). Heuristics: Intelligent search strategies for computer problem solving.

[Pearl, 1985] Pearl, J. (1985). Bayesian networks: A model of self-activated memory for evidential reasoning. *Proceedings of the 7th Conference of the Cognitive Science Society*, pages 329–334.

[Pearl, 1988] Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann.

[Peterson and Söderberg, 1989] Peterson, C. and Söderberg, B. (1989). A new method for mapping optimization problems onto neural networks. *International Journal of Neural Systems*, 1(01):3–22.

[Pfeiffer, 2012] Pfeiffer, D. (2012). *The Stixel World: A Compact Medium-level Representation for Efficiently Modeling Dynamic Three-dimensional Environments*. PhD thesis, Humboldt-University of Berlin, Berlin, Germany.

[Pfeiffer et al., 2012] Pfeiffer, D., Erbs, F., and Franke, U. (2012). Pixels, stixels, and objects. In *Computer Vision–ECCV 2012. Workshops and Demonstrations*, pages 1–10. Springer.

[Pfeiffer and Franke, 2010] Pfeiffer, D. and Franke, U. (2010). Efficient representation of traffic scenes by means of dynamic stixels. In *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pages 217–224. IEEE.

[Pfeiffer and Franke, 2011] Pfeiffer, D. and Franke, U. (2011). Towards a global optimal multi-layer stixel representation of dense 3d data. In *British Machine Vision Conference (BMVC), Dundee, Scotland*.

[Pfeiffer et al., 2010] Pfeiffer, D., Morales, S., Barth, A., and Franke, U. (2010). Ground truth evaluation of the stixel representation using laser scanners. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 1091–1097. IEEE.

[Pfeiffer et al., 2013] Pfeiffer, D., Schneider, N., and Gehrig, S. (2013). Exploiting the power of stereo confidences. In *Computer Vision and Pattern Recognition, 2013. CVPR 2013. Proceedings of the 2013 IEEE Computer Society Conference on*. IEEE.

[Pollefeys et al., 2008] Pollefeys, M., Nistér, D., Frahm, J.-M., Akbarzadeh, A., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S.-J., Merrell, P., et al. (2008). Detailed real-time urban 3d reconstruction from video. *International Journal of Computer Vision*, 78(2-3):143–167.

[Pons et al., 2007] Pons, J., Keriven, R., and Faugeras, O. (2007). Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. *International Journal of Computer Vision*, 72(2):179–193.

[Potts, 1952] Potts, R. B. (1952). Some generalized order-disorder transformations. In *Proceedings of the Cambridge Philosophical Society*, volume 48, pages 106–109. Cambridge Univ Press.

[Prătorius, 1993] Prătorius, G. (1993). Das PROMETHEUS-Projekt.

[Premebida et al., 2009] Premebida, C., Ludwig, O., and Nunes, U. (2009). LIDAR and Vision-based Pedestrian Detection System. *Journal of Field Robotics*, 26(9):696–711.

[Prim, 1957] Prim, R. C. (1957). Shortest connection networks and some generalizations. *Bell system technical journal*, 36(6):1389–1401.

[Rabe, 2011] Rabe, C. (2011). *Detection of Moving Objects by Spatio-Temporal Motion Analysis*. PhD thesis, University of Kiel, Kiel, Germany.

[Rabe et al., 2010] Rabe, C., Müller, T., Wedel, A., and Franke, U. (2010). Dense, Robust, and Accurate Motion Field Estimation from Stereo Image Sequences in Real-Time. In Daniilidis, K., Maragos, P., and Paragios, N., editors, *Proceedings of the 11th European Conference on Computer Vision*, volume 6314 of *Lecture Notes in Computer Science*, pages 582–595. Springer.

[Rabiner, 1989] Rabiner, L. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

[Ranftl et al., 2012] Ranftl, R., Gehrig, S., Pock, T., and Bischof, H. (2012). Pushing the limits of stereo using variational stereo estimation. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 401–407. IEEE.

[Rother et al., 2009] Rother, C., Kohli, P., Feng, W., and Jia, J. (2009). Minimizing sparse higher order energy functions of discrete variables. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1382–1389. IEEE.

[Rother et al., 2004] Rother, C., Kolmogorov, V., and Blake, A. (2004). Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 309–314. ACM.

[Rother et al., 2007] Rother, C., Kolmogorov, V., Lempitsky, V., and Szummer, M. (2007). Optimizing binary mrfs via extended roof duality. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE.

[Sarawagi and Cohen, 2004] Sarawagi, S. and Cohen, W. (2004). Semi-markov conditional random fields for information extraction. *Advances in Neural Information Processing Systems*, 17:1185–1192.

[Scharstein and Szeliski, 1998] Scharstein, D. and Szeliski, R. (1998). Stereo matching with nonlinear diffusion. *International Journal of Computer Vision*, 28(2):155–174.

[Scharstein and Szeliski, 2002] Scharstein, D. and Szeliski, R. (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International journal of computer vision*, 47(1):7–42.

[Scharwächter et al., 2013] Scharwächter, T., Enzweiler, M., Franke, U., and Roth, S. (2013). Efficient multi-cue scene segmentation. In *Pattern Recognition*, pages 435–445. Springer.

[Scharwächter et al., 2014] Scharwächter, T., Enzweiler, M., Franke, U., and Roth, S. (2014). Stixmantics: A medium-level model for real-time semantic scene understanding. In *Computer Vision–ECCV 2014*, pages 533–548. Springer.

[Scharwaechter, 2012] Scharwaechter, T. (2012). Stereo scene analysis under adverse conditions. Master's thesis, RWTH Aachen.

[Schlesinger and Flach, 2006] Schlesinger, D. and Flach, B. (2006). *Transforming an arbitrary minsum problem into a binary one.* TU, Fak. Informatik.

[Schmid et al., 2010] Schmid, M., Maehlisch, M., Dickmann, J., and Wuensche, H.-J. (2010). Dynamic level of detail 3d occupancy grids for automotive use. *IEEE Intelligent Vehicles Symposium (IV)*, pages 269–274.

[Schneider, 2005] Schneider, M. (2005). Automotive Radar – Status and Trends. *German Microwave Conference (GeMIC).*

[Schneider et al., 2012] Schneider, N., Gehrig, S., Pfeiffer, D., and Banitsas, K. (2012). An evaluation framework for stereo-based driver assistance. In *Outdoor and Large-Scale Real-World Scene Analysis*, pages 27–51. Springer.

[Schopper et al., 2013] Schopper, M., Henle, L., and Wohland, T. (2013). Intelligent drive vernetzte intelligenz für mehr sicherheit. *ATZextra*, 18(5):106–114.

[Schuhmacher et al., 2008] Schuhmacher, D., Vo, B.-T., and Vo, B.-N. (2008). A consistent metric for performance evaluation of multi-object filters. *Signal Processing, IEEE Transactions on*, 56(8):3447–3457.

[Schulter et al., 2013] Schulter, S., Leistner, C., Roth, P. M., and Bischof, H. (2013). Unsupervised object discovery and segmentation in videos. In *British machine vision conference (BMVC).*

[Schütz et al., 2012] Schütz, M., Wiyogo, Y., Schmid, M., and Dickmann, J. (2012). Laser-based hierarchical grid mapping for detection and tracking of moving objects. *Advanced Microsystems for Automotive Applications 2012*, pages 167–176.

[Schwarz, 1978] Schwarz, G. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464.

[Sheasby et al., 2012] Sheasby, G., Warrell, J., Zhang, Y., Crook, N., and Torr, P. H. (2012). Simultaneous human segmentation, depth and pose estimation via dual decomposition. *Proceedings of the workshop of British Machine Vision Conference (BMVC).*

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

[Sibley et al., 2007] Sibley, G., Matthies, L., and Sukhatme, G. (2007). Bias reduction and filter convergence for long range stereo. In *Robotics Research*, pages 285–294. Springer.

[Sinha et al., 2006] Sinha, S. N., Frahm, J.-M., Pollefeys, M., and Genc, Y. (2006). Gpu-based video feature tracking and matching. In *EDGE, Workshop on Edge Computing Using New Commodity Architectures*, volume 278, page 4321.

[Sivia, 1996] Sivia, D. S. (1996). *Data Analysis: A Bayesian Tutorial*. Oxford University Press, USA.

[Skolnik, 1962] Skolnik, M. I. (1962). Introduction to radar. *Radar Handbook*, page 1990.

[Spelke, 1990] Spelke, E. S. (1990). Principles of object perception. *Cognitive science*, 14(1):29–56.

[Spies and Spies, 2006] Spies, M. and Spies, H. (2006). Automobile lidar sensorik: Stand, trends und zukuenftige herausforderungen. *Advances in Radio Science*, 4:99–104.

[Stein, 2004] Stein, F. (2004). Efficient computation of optical flow using the census transform. *Pattern Recognition*, pages 79–86.

[Steinemann et al., 2012] Steinemann, P., Klappstein, J., Dickann, J., Wuensche, H.-J., and v. Hundelshausen, F. (2012). 3D outline contours of vehicles in 3D-LIDAR-measurements for tracking extended targets. *IEEE Intelligent Vehicles Symposium 2012 (IV)*, pages 432–437.

[Steinhaus, 1956] Steinhaus, H. (1956). Sur la division des corp materiels en parties. *Bull. Acad. Polon. Sci*, 1:801–804.

[Strekalovskiy and Cremers, 2011] Strekalovskiy, E. and Cremers, D. (2011). Generalized ordering constraints for multilabel optimization. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2619–2626. IEEE.

[Sturm, 1999] Sturm, J. F. (1999). Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization methods and software*, 11(1-4):625–653.

[Sun et al., 2012] Sun, D., Sudderth, E. B., and Black, M. J. (2012). Layered segmentation and optical flow estimation over time. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1768–1775. IEEE.

[Sun et al., 2003] Sun, J., Zheng, N., and Shum, H. (2003). Stereo matching using belief propagation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(7):787–800.

[Szeliski et al., 2008] Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., and Rother, C. (2008). A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(6):1068–1080.

[Tappen and Freeman, 2003] Tappen, M. and Freeman, W. (2003). Comparison of graph cuts with belief propagation for stereo, using identical mrf parameters. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 900–906. IEEE.

[Thrun, 2010] Thrun, S. (2010). What we're driving at. *http://googleblog.blogspot.de/2010/10/what-were-driving-at.html*.

[Thrun et al., 2005] Thrun, S., Burgard, W., Fox, D., et al. (2005). *Probabilistic robotics*, volume 1. MIT press Cambridge, MA.

[Thrun et al., 2006] Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., et al. (2006). Stanley: The robot that won the DARPA Grand Challenge. *Journal of field Robotics*, 23(9):661–692.

[Tomasi and Kanade, 1991] Tomasi, C. and Kanade, T. (1991). *Detection and tracking of point features*. School of Computer Science, Carnegie Mellon Univ.

[Trobin et al., 2008] Trobin, W., Pock, T., Cremers, D., and Bischof, H. (2008). Continuous energy minimization via repeated binary fusion. In *Computer Vision–ECCV 2008*, pages 677–690. Springer.

[Unger et al., 2012] Unger, M., Werlberger, M., Pock, T., and Bischof, H. (2012). Joint Motion Estimation and Segmentation of Complex Scenes with Label Costs and Occlusion Modeling. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1878–1885. IEEE.

[Urmson et al., 2007] Urmson, C., Anhalt, J., Bagnell, D., Baker, C., Bittner, R., Dolan, J., Duggins, D., Ferguson, D., Galatali, T., Geyer, C., et al. (2007). Tartan Racing: A Multi-Modal Approach to the DARPA Urban Challenge. *Defense Advanced Res. Projects Agency, Arlington, VA, DARPA Tech. Rep.*

[Vedula et al., 1999] Vedula, S., Baker, S., Rander, P., Collins, R., and Kanade, T. (1999). Three-dimensional scene flow. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 722–729. IEEE.

[Vedula et al., 2005] Vedula, S., Rander, P., Collins, R., and Kanade, T. (2005). Three-dimensional scene flow. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(3):475–480.

[Veksler, 2005] Veksler, O. (2005). Stereo correspondence by dynamic programming on a tree. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 384–390. IEEE.

[Veksler, 2012] Veksler, O. (2012). Dynamic programming for approximate expansion algorithm. pages 850–863.

[Veksler et al., 2010] Veksler, O., Boykov, Y., and Mehrani, P. (2010). Superpixels and supervoxels in an energy optimization framework. In *Computer Vision–ECCV 2010*, pages 211–224. Springer.

[Viterbi, 1967] Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, IEEE Transactions on*, 13(2):260–269.

[Vo and Ma, 2006] Vo, B.-N. and Ma, W.-K. (2006). The gaussian mixture probability hypothesis density filter. *Signal Processing, IEEE Transactions on*, 54(11):4091–4104.

[Vogel et al., 2015] Vogel, C., Schindler, K., and Roth, S. (2015). 3d scene flow estimation with a piecewise rigid scene model. pages 1–28.

[Wainwright et al., 2005] Wainwright, M. J., Jaakkola, T. S., and Willsky, A. S. (2005). Map estimation via agreement on trees: message-passing and linear programming. *Information Theory, IEEE Transactions on*, 51(11):3697–3717.

[Wedel et al., 2011] Wedel, A., Brox, T., Vaudrey, T., Rabe, C., Franke, U., and Cremers, D. (2011). Stereoscopic scene flow computation for 3d motion understanding. *International journal of computer vision*, 95(1):29–51.

[Wedel et al., 2009] Wedel, A., Meißner, A., Rabe, C., Franke, U., and Cremers, D. (2009). Detection and segmentation of independently moving objects from dense scene flow. In *Energy minimization methods in computer vision and pattern recognition*, pages 14–27. Springer.

[Wedel et al., 2008] Wedel, A., Rabe, C., Vaudrey, T., Brox, T., Franke, U., and Cremers, D. (2008). Efficient dense scene flow from sparse or dense stereo data. *Computer Vision–ECCV 2008*, pages 739–751.

[Weiss et al., 2004] Weiss, K., Kaempchen, N., and Kirchner, A. (2004). Multiple-Model Tracking for the Detection of Lane Change Maneuvers. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 937–942. IEEE.

[Williams, 1988] Williams, M. (1988). PROMETHEUS-the European research programme for optimising the road transport system in Europe. In *Driver Information, IEE Colloquium on*, pages 1–1. IET.

[Winn and Shotton, 2006] Winn, J. and Shotton, J. (2006). The layout consistent random field for recognizing and segmenting partially occluded objects. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 37–44. IEEE.

[Winner et al., 2012] Winner, H., Danner, B., and Steinle, J. (2012). Adaptive cruise control. *Handbuch Fahrerassistenzsysteme*, pages 478–521.

[Witkin, 1981] Witkin, A. (1981). Recovering surface shape and orientation from texture. *Artificial Intelligence*, 17(1):17–45.

[Witte, 2013] Witte, A. (2013). Stixel-Based image segmentation for Object Detection in Traffic Scenes. Master's thesis, RWTH Aachen, Aachen, Germany.

[Wojek et al., 2010] Wojek, C., Roth, S., Schindler, K., and Schiele, B. (2010). Monocular 3d scene modeling and inference: Understanding multi-object traffic scenes. *Computer Vision–ECCV 2010*, pages 467–481.

[Wojek and Schiele, 2008] Wojek, C. and Schiele, B. (2008). A dynamic conditional random field model for joint labeling of object and scene classes. *Computer Vision–ECCV 2008*, pages 733–747.

[Xiao and Quan, 2009] Xiao, J. and Quan, L. (2009). Multiple view semantic segmentation for street view images. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 686–693. IEEE.

[Yang et al., 2010] Yang, Q., Wang, L., and Ahuja, N. (2010). A constant-space belief propagation algorithm for stereo matching. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1458–1465. IEEE.

[Yedidia et al., 2003] Yedidia, J., Freeman, W., and Weiss, Y. (2003). Understanding belief propagation and its generalizations. *Exploring artificial intelligence in the new millennium*, 8:236–239.

[Yu et al., 2007] Yu, T., Lin, R., Super, B., and Tang, B. (2007). Efficient message representations for belief propagation. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE.

[Yuan and Boykov, 2010] Yuan, J. and Boykov, Y. (2010). Tv-based multi-label image segmentation with label cost prior. In *British machine vision conference (BMVC)*.

[Zach et al., 2007] Zach, C., Pock, T., and Bischof, H. (2007). A duality based approach for realtime TV-L1 optical flow. *Pattern Recognition*, pages 214–223.

[Zappella et al., 2008] Zappella, L., Lladó, X., and Salvi, J. (2008). Motion segmentation: A review. In *Proceedings of the 2008 conference on Artificial Intelligence Research and Development: Proceedings of the 11th International Conference of the Catalan Association for Artificial Intelligence*, pages 398–407. IOS Press.

[Zhang et al., 2010] Zhang, C., Wang, L., and Yang, R. (2010). Semantic segmentation of urban scenes using dense depth maps. *Computer Vision–ECCV 2010*, pages 708–721.

[Zhang and Kambhamettu, 2001] Zhang, Y. and Kambhamettu, C. (2001). On 3d scene flow and structure estimation. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–778. IEEE.

[Zheng et al., 2012] Zheng, Y., Gu, S., and Tomasi, C. (2012). Fast tiered labeling with topological priors. In *Computer Vision – ECCV 2012*, volume 7575 of *Lecture Notes in Computer Science*, pages 587–601. Springer Berlin Heidelberg.

[Zhu and Yuille, 1996] Zhu, S. C. and Yuille, A. (1996). Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(9):884–900.

[Ziegler et al., 2014] Ziegler, J., Bender, P., Schreiber, M., Lategahn, H., Strauss, T., Stiller, C., Dang, T., Franke, U., Appenrodt, N., Keller, C., Kaus, E., Herrtwich, R., Rabe, C., Pfeiffer, D., Lindner, F., Stein, F., Erbs, F., Enzweiler, M., Knoeppel, C., Hipp, J., Haueis, M., Trepte, M., Brenk, C., Tamke, A., Ghanaat, M., Braun, M., Joos, A., Fritz, H., Mock, H., Hein, M., and Zeeb, E. (2014). Making bertha drive-an autonomous journey on a historic route. *Intelligent Transportation Systems Magazine, IEEE*, 6(2):8–20.

[Zomet et al., 2001] Zomet, A., Rav-Acha, A., and Peleg, S. (2001). Robust super-resolution. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages 645–650. IEEE.

[Zomotor, 1987] Zomotor, A. (1987). *Fahrwerktechnik: Fahrverhalten: Kraefte am Fahrzeug, Bremsverhalten, Lenkverhalten, Testverfahren, Messtechnik, Bewertungsmethoden, Versuchseinrichtungen, aktive Sicherheit, Unfallverhuetung*. Vogel.