



Technische Universität München
Ingenieurfaculty Bau Geo Umwelt
Lehrstuhl für Geoinformatik
Univ.-Prof. Dr. rer. nat. Thomas H. Kolbe

Straßenraummodellierung mittels Mobile Mapping in OpenDRIVE und CityGML sowie Entwicklung geeigneter Visualisierungsmethoden

Theresa Coduro, B.Sc.

Master's Thesis

Bearbeitung: 24.11.2017 - 24.05.2018
Studiengang: Geodäsie und Geoinformation (Master)
Betreuer: Univ.-Prof. Dr. rer. nat. Thomas H. Kolbe (TUM)
Christof Beil, M.Sc. (TUM)
Christiane Radies, M.Sc. (3D Mapping Solutions)

Kooperation:



2018

Selbstständigkeitserklärung

Erklärung gemäß §18 Absatz 9 APSO der Technischen Universität München (TUM):

„Ich versichere, dass ich diese Master’s Thesis selbstständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe“

München, den 24. Mai 2018

Theresa Coduro

Kurzfassung

In der vorliegenden Arbeit werden Methoden zur Überführung von Straßenraumdaten in ein semantisches 3D-Stadtmodell, welches vielfältige Analysen und Visualisierungsmöglichkeiten bietet, erarbeitet. Die Methoden werden anhand einer Umwandlung von Mobile Mapping Daten, die für das OpenDRIVE-Format aufbereitet wurden, in das CityGML-Format exemplarisch untersucht. Für die Durchführung von Experimenten wurden Datensätze durch das beteiligte Unternehmen 3D Mapping Solutions GmbH, das sich mit der kinematischen Aufnahme von Mobile Mapping Daten und der Auswertung zu hochgenauen Bestandsplänen im OpenDRIVE-Format befasst, zur Verfügung gestellt.

Neben verwandten Arbeiten wird insbesondere auch auf Daten eingegangen, die für die Straßenraummodellierung verwendet werden können. Die Beurteilung der aufgezeigten Datenquellen erfolgt anhand von neun Kriterien und legt die Kombination unterschiedlicher Datenquellen als bestmögliche Grundlage zur Erzeugung von Straßenraummodellen nahe. Zusätzlich wird eine Übersicht über unterschiedliche Modellierungsstandards aus dem Bereich der Straßenraummodellierung gegeben. Hauptsächlich wird vergleichend auf die Standards OpenDRIVE und CityGML eingegangen, wobei Ähnlichkeiten und Unterschiede der beiden Datenmodelle dargelegt werden. Der generelle Erfassungsprozess semantischer Straßenraumdaten mittels Mobile Mapping wird aufgezeigt, sowie auf das spezielle Vorgehen der Firma 3D Mapping Solutions eingegangen. Aufbauend darauf wird eine Strategie für die mögliche Visualisierung der kinematisch erfassten Vermessungsdaten des Unternehmens vorgestellt.

Im Zuge der Realisierung des entwickelten Ansatzes, konnte anhand konkreter Testgebiete gezeigt werden, dass die Umwandlung der für das OpenDRIVE-Format aufbereiteten Mobile Mapping Daten in das CityGML-Format eine dreidimensionale Visualisierung der kinematisch erfassten Straßenraumdaten ermöglicht. Das Ergebnis der praktischen Umsetzung ist ein semantisches Straßenraummodell im Level of Detail 3, welches neben einem texturierten Spurmodell zusätzlich verschiedene Straßenraumobjekte beinhaltet. Die abschließende Visualisierung erfolgte im, auf Cesium basierenden, 3D-Web-Map-Client der 3DCityDB.

Abstract

In this thesis, methods for transforming street space data to a 3D city model, which allows for detailed analysis and visualization, are developed. For the experiments, the proposed approach was exemplarily implemented for the conversion of Mobile Mapping data, pre-processed for publication in OpenDRIVE format, to CityGML. For the experiments, datasets were provided by the co-supervising company 3D Mapping Solutions GmbH, which utilizes mobile mapping systems to acquire high-resolution data of street spaces and derive highly accurate OpenDRIVE-compliant plans from it.

Besides related work, relevant sources of data for street space modeling are discussed in detail. Based on a catalog of nine criteria, the introduced sources of data are assessed and a well-suited combination of data sources is suggested. Moreover, an overview of standards for the modelling of street spaces, with a particular focus on the similarities and differences of OpenDRIVE and CityGML, is presented. The acquisition of semantic street space data using mobile mapping, in general, and the acquisition process implemented by 3D Mapping Solutions, in particular, is described. Based on this, an approach for the visualization of such kinematic measurements was developed.

In experiments, an implementation of this approach was successfully applied to study areas to show that a transformation of the pre-processed Mobile Mapping data to CityGML indeed enables visualization of the kinematic street space data in 3D. The result of this transformation is a semantic street space model in Level of Detail 3, containing a textured lane model as well as additional objects from the street space. Visualization of such models is possible using the Cesium-based 3D-Web-Map-Client, which is part of 3DCityDB.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Thematischer Hintergrund und Motivation	1
1.2	Problemstellung	3
1.3	Lösungsansatz	4
1.4	Aufbau und Inhalt der Arbeit	6
2	Literatur und verwandte Arbeiten	7
2.1	Anwendungsfelder und Anforderungen	7
2.2	Methoden der Informationsextraktion	8
2.3	Mögliche Datenquellen	10
2.3.1	Übersicht über Literatur möglicher Datenquellen	10
2.3.2	Kriterien zur Beurteilung genannter Datenquellen	11
3	Möglichkeiten zur Modellierung des Straßenraums	19
3.1	Überblick über Modellierungsstandards	19
3.2	OpenDRIVE	22
3.2.1	Wichtige Koordinatensysteme	23
3.2.2	Grundprinzip des Straßenverlaufs	24
3.2.3	Aufbau eines Straßenabschnitts	25
3.3	City Geography Markup Language	29
3.3.1	Kernmodell: CityGML Core	30
3.3.2	Relevante Erweiterungsmodelle	31
3.3.3	XLinks in CityGML	37
3.3.4	CityGML-Erweiterungen	38
4	Erfassung semantischer Straßenraummodelle mittels Mobile Mapping	39
4.1	Genereller Erfassungsprozess	39
4.1.1	Datenerfassung und Vorverarbeitung	40
4.1.2	Auswertung und Modellierung	41
4.2	Erfassungsprozess bei 3D Mapping Solutions	42
4.2.1	Aufbau des Messsystems und kinematische Vermessung	42
4.2.2	Auswertung und Modellierung	45

5	Abbildung des OpenDRIVE-Datenmodells auf das CityGML-Datenmodell	51
5.1	Inhalte und Anwendungsbereiche	51
5.2	Formaler Aufbau der Datenmodelle	52
5.3	Unterschiede der Datenmodelle bezüglich des Straßenraums	55
5.3.1	Geometrie	55
5.3.2	Topologie	61
5.3.3	Semantik	62
5.3.4	Visuelle Erscheinung	65
5.4	Semantische Zuordnung	66
6	Konzeptentwicklung zur Visualisierung von Mobile Mapping Daten	69
6.1	Mögliche Wege der Weiterverarbeitung	69
6.2	Tatsächliche Umsetzung	72
7	Erweiterung der Auswerteprozesse zur Straßenraummodellierung	75
7.1	Verwendete Software-Pakete	75
7.1.1	Feature Manipulation Engine	76
7.1.2	3D City Database	76
7.1.3	3DCityDB Web-Map-Client	77
7.2	Analyse der Grundlagendaten	77
7.3	Weiterverarbeitung in FME	78
7.3.1	Modellierung des Spurmodells	78
7.3.2	Modellierung der Straßenraumobjekte	82
7.3.3	Umwandlung nach CityGML	87
7.4	Darstellung im 3DCityDB Web-Map-Client	89
8	Realisierung anhand konkreter Testgebiete	91
8.1	Visualisierung und Evaluierung von Testgebiet 1	91
8.1.1	Aufbereitung der Grundlagendaten	91
8.1.2	Modellierung in FME	93
8.1.3	Darstellung der Ergebnisse im Webviewer	97
8.2	Übertragbarkeit auf Testgebiet 2	99
8.2.1	Aufbereitung der Grundlagendaten	99
8.2.2	Überprüfung der Übertragbarkeit	100
8.3	Kritische Diskussion der Ergebnisse	102
9	Fazit und Ausblick	105
	Abbildungsverzeichnis	117
	Tabellenverzeichnis	119
	Anhang	121

A	Generierte Attribute in Tabellen	123
B	Liste der modellierten Verkehrsschilder und Ampeln	127
C	Quellen von verwendeten 3D-Modelle und Texturen	129

Abkürzungsverzeichnis

3DCityDB 3D City Database

ADE Application Domain Extension

ALKIS Amtliches Liegenschaftskataster-Informationssystem

ALS Airborne Laser Scanning

ASB Anweisung Straßeninformationsbank

ATKIS Amtliches Topographisch-Kartographisches Informationssystem

BASt Bundesanstalt für Straßenwesen

BIM Building Information Modelling

CAD Computer-Aided Design

CityGML City Geography Markup Language

DETMETER Digital Electronic Mapping of European Territory

DGM Digitales Geländemodell

DGPS Differentielles Globales Positionierungssystem

DOP Digitales Orthophoto

ETL Extract, Transform, Load

FME Feature Manipulation Engine

GDF Geographic Data File

GIS Geographisches Informationssystem

GML Geography Markup Language

GNSS Globales Navigationssatellitensystem

IFC Industry Foundation Classes

IMU Inertial Measurement Unit

IT-KoA IT-Koordinationsausschuss Bund-Länder im Straßenwesen

ITS Intelligente Transportsysteme

KML Keyhole Markup Language

LandInfra Land and Infrastructure Conceptual Model Standard

LDBV Landesamt für Digitalisierung, Breitband und Vermessung

LoD Level of Detail

MLS Mobiles Laser Scanning

MMS Mobile Mapping System

OGC Open Geospatial Consortium

OKSTRA Objektkatalog für Straßen- und Verkehrswesen

OSM OpenStreetMap

TLS Terrestrisches Laser Scanning

TUM Technische Universität München

UAVs Unmanned Aerial Vehicles

UML Unified Modeling Language

W3C World Wide Web Consortium

XML Extensible Markup Language

1 Einleitung

1.1 Thematischer Hintergrund und Motivation

3D-Stadtmodelle dienen heutzutage als Informations- und Analysegrundlage für unterschiedliche Aufgaben. Der umfassende Journal-Artikel von Biljecki et al. [2015] der TU Delft mit dem Titel "Applications of 3D City Models: State of the Art Review" stellt eine umfangreiche Auflistung von möglichen Anwendungsfeldern für 3D-Stadtmodelle vor. Basierend auf einer umfangreichen Literaturstudie mit zahlreichen wissenschaftlichen Artikeln, technischen Reports und Onlinequellen werden insgesamt 29 Anwendungsfelder mit Beispielen vorgestellt. Als wesentlicher Verwendungszweck wird die einfache Visualisierung, z.B. für die Darstellung von Immobilien und geplanten Projekten, Webvisualisierungen oder Augmented Reality Anwendungen, angegeben. Zusätzlich können 3D-Stadtmodelle auch in Fällen, in denen die Satellitenpositionierung wenig zuverlässig ist, unterstützend für die Positionsbestimmung angewendet werden. In solchen Fällen kann mit Hilfe von 3D-Stadtmodellen die Sichtbarkeit von Satelliten der globalen Navigationssatellitensysteme (GNSS) analysiert und für eine verbesserte Navigation herangezogen werden. Anhand solcher Anwendungsfälle wird deutlich, dass eine Kombination aus Gebäude- und Straßenraummodellen wichtig ist. Stadtmodelle als Basis von Sichtbarkeitsanalysen können zudem die Verkehrssicherheit erhöhen. Neben der Visualisierung werden 3D-Stadtmodelle aber auch für andere Zwecke und in anderen Bereichen verwendet. Genannte Anwendungsbeispiele von Biljecki et al. [2015], die sich speziell auf Gebäudemodelle beziehen, lassen sich in vielen Fällen auch auf Straßenraummodelle übertragen. Die Bewertung der Sonneneinstrahlung beispielsweise ist nicht nur für die Beurteilung von Sonnenkollektoren auf Gebäudedächern interessant, sondern auch für die Bewertung der Straßenoberfläche und deren erwarteter Zustand nach extremer Sonneneinstrahlung. In Navigationsanwendungen helfen dreidimensionale Modelle mit Hinweispunkten bei der Orientierung und sind meist intuitiver und realistischer aufgebaut als zweidimensionale Karten. Letztere sind als Basis für die Distanzschätzung in Navigationsaufgaben jedoch besser geeignet. Für Routingaufgaben wie der Fußgängernavigation ist es sinnvoll dreidimensionale Straßenraummodelle zu verwenden, da Informationen über Rampen und Stufen im Zweidimensionalen nicht verfügbar sind.

Die große Bedeutung von Straßenraummodellen wird in momentan sehr aktuellen Themen deutlich. Fahrassistenzsysteme und das autonome Fahren benötigen Straßenraummodelle als Grundlage. Die für solche Anwendungsbereiche erforderte Positionsgenauigkeit von wenigen Zentimetern kann vor allem in urbanen Gebieten nicht allein durch die Lokalisierung mittels GNSS erreicht werden. Mit zusätzlich eingesetzten Sensoren wie Stereokameras oder Laserscansystemen können Grundlagen-

daten gewonnen werden durch die eine Positionierung in hochaufgelösten Karten möglich ist [Siegle et al., 1992; Schreiber et al., 2013; Levinson et al., 2007]. So verwenden Schreiber et al. [2013] und Wu und Ranganathan [2013] Straßenmarkierungen als kartenbasiertes Zusatzwissen, Qu et al. [2015] setzen Kartendaten mit georeferenzierten Verkehrszeichen zur Ortung in der Karte ein.

Zahlreiche wissenschaftliche Artikel und Forschungsarbeiten greifen die Straßendatenerfassung und Straßenraummodellierung auf. Dabei kann eine Unterteilung der Literatur zu diesem Thema in drei Hauptgruppen vorgenommen werden. Zur ersten Gruppe gehören Arbeiten, die sich mit bestimmten Anwendungsfeldern und den damit verbundenen Anforderungen beschäftigen. Die zweite Gruppe beinhaltet Forschungsarbeiten, die sich mit Methoden der Informationsextraktion auseinandersetzen. Zusätzlich geben zahlreiche Arbeiten Einblicke in für diesen Zweck potentiell nutzbare Datenquellen. Straßenraummodelle können aus Offenen Daten wie OpenStreetMap-Daten, aber auch aus amtlichen Geobasisdaten abgeleitet werden. Häufig werden auch photogrammetrische Methoden und Laserscanning-Verfahren für die Erfassung von Grundlagendaten verwendet. Meistens ist eine Bewertung in Bezug auf die Anwendung der Straßenraummodellierung mittels verschiedener Kriterien notwendig. Sowohl bei der Datenerfassung als auch bei der Informationsextraktion aus den gewonnenen Daten werden in den meisten Fällen kombinierte Methoden verwendet. Mobile Mapping Systeme (MMS) kombinieren oftmals Laserscanner und Kameras zur Umgebungserfassung, deren Daten durch Positionierungssensoren ergänzt und georeferenziert werden. Punktwolken mit hoher Punktdichte und hochaufgelöste Kameradaten ermöglichen die Extraktion verschiedenster Straßenraumobjekte. Durch die Befestigung der Sensoren auf einer mobilen Plattform, die eine schnelle Datenerfassung ermöglicht, werden MMS oft als bevorzugte Datenerfassungsmethode verwendet [Lemmens, 2017].

Aus vielfältigen Anwendungsfeldern haben sich auch unterschiedliche anwenderspezifische Modellierungsstandards entwickelt. In manchen Bereichen unterscheiden sich diese Standards, in anderen sind sie sich sehr ähnlich oder gehen sogar auseinander hervor. In den meisten Fällen sind die Modellierungsstandards aus ihren primären Anwendungsgebieten entstanden. Der Standard Geographic Data File (GDF) beispielsweise wurde vor allem zum standardisierten Aufbau und Austausch von digitalen Straßendaten entwickelt und wird hauptsächlich im Bereich der Fahrzeugnavigation und Verkehrstelematik eingesetzt [ISO 14825, 2011]. Das OpenDRIVE-Format der VIRES Simulationstechnologie GmbH dient als allgemeine Grundlage zur Beschreibung von spurbasierten Verkehrsnetzen und wird auch in Fahrassistenzsystemen, speziell in der Navigation, aber auch in Fahr- und Verkehrssimulatoren eingesetzt [Dupuis et al., 2010]. Die vom Open Geospatial Consortium (OGC) und der ISO TC211 definierte City Geography Markup Language (CityGML) ist ein offenes Datenmodell für die Repräsentation und den Austausch von virtuellen 3D-Stadtmodellen mit hohem visuellen Detaillierungsgrad. Mithilfe dieser Modellierungsstandards können umfassende 3D-Stadtmodelle mit Elementen wie Gebäuden, Straßen und Gewässernetzen dargestellt werden [Gröger et al., 2012].

Die Bearbeitung der Masterarbeit erfolgt am Lehrstuhl für Geoinformatik der Technische Universität München (TUM) in Kooperation mit der Firma 3D Mapping Solutions GmbH. Das beteiligte Unternehmen hat sich auf die Kalibrierung von Multisensorsystemen und die kinematischen Vermessung

von Verkehrsnetzen spezialisiert. 3D Mapping Solutions verfügt dazu über mehrere Messfahrzeuge bzw. Messaufbauten mit einem in Eigenentwicklung entstandenen Sensorsystems. Das Messsystem besteht zum einen aus Sensoren für die Aufnahme der Umgebung wie Kameras und Laserscanner und zum anderen aus Sensoren für die Geolokalisierung wie GNSS-Empfänger. Die generierten kinematischen Vermessungsdaten werden im Anschluss an die Befahrung firmenintern ausgewertet. Der Schwerpunkt liegt dabei auf der Bereitstellung von Referenzinformationen und Referenzkarten von Straßennetzen. Bereitgestellt werden diese vor allem im OpenDRIVE-Standard, der auf der Extensible Markup Language (XML) basiert. Obwohl auf Kundenwunsch auch andere Formate zur Verfügung gestellt werden können, hat sich OpenDRIVE als Ziel- und Austauschformat, insbesondere im Bereich der Fahrzeugsimulation, etabliert.

1.2 Problemstellung

Wie in Abschnitt 1.1 aufgezeigt, kann die Modellierung des Straßenraums aus verschiedenen Rohdaten und für verschiedene Anwendungen entstehen. Neben der Art der modellierten Informationen unterscheiden sich auch die Anforderungen an die Rohdaten und die Art der Aufbereitung je nach Anwendungsgebiet. Im Bereich der Fahrtrainingssimulationen ist die absolute Genauigkeit der modellierten Straßenraumelemente beispielsweise weniger relevant als der visuelle Detaillierungsgrad. In Anwendungsfällen wie dem autonomen Fahren weisen diese Anforderungen jedoch ein gegensätzliches Verhalten auf. Demnach ist die visuelle Erscheinung im Gegensatz zur absoluten Genauigkeit der Objekte zweitrangig.

Das beteiligte Unternehmen 3D Mapping Solutions hat sich auf die Bereitstellung von Referenzinformationen und Referenzkarten von Straßennetzen im OpenDRIVE-Standard spezialisiert. Der Auswertung und Modellierung der Rohdaten wird vereinfacht in Abb. 1.1 dargestellt. Aus den durch Mobile Mapping aufgenommenen Punktwolken und Kameradaten wird mit mehreren Zwischenschritten ein OpenDRIVE-Dokument erstellt. Weit verbreitet ist dieser Modellierungsstandard hauptsächlich in der Automobilindustrie bei der Entwicklung und zur Simulation im Bereich des autonomen Fahrens. Durch diesen Modellierungs- und Datenaustauschstandard werden der Verlauf und die Umgebung einer Fahrbahn parametrisch unter Verwendung der XML, einer internationalen standardisierten Auszeichnungssprache des World Wide Web Consortiums (W3C), definiert. Das visuelle Endprodukt von OpenDRIVE ist eine Referenzlinie mit der verschiedene Spur- und Objektinformationen verknüpft sind. Eine Visualisierung der Daten im OpenDRIVE-Format kann mithilfe des ODR-Viewer Scenario Editors vorgenommen werden. Das von der Firma VIRES Simulationstechnologie entwickelte Programm bietet jedoch nur eine sehr abstrakte Visualisierung der Daten. Alternativ können die Referenzlinien der befahrenen Strecken aus dem erzeugten OpenDRIVE-Dokument auch in das Keyhole Markup Language (KML)-Format konvertiert werden und dann beispielsweise in Google Earth visualisiert werden. Hierbei erfolgt die Darstellung allerdings ausschließlich linienhaft und beinhaltet außer dem Namen der Referenzlinie keinerlei semantische Information. Lediglich als Streckenbeschreibung ist eine solche Darstellung in der Praxis ausreichend.



Abbildung 1.1 – Überblick über den Datenauswertungs- und Modellierungsprozess bei 3D Mapping Solutions (Bildquellen: 3D Mapping Solutions)

Für die Präsentation der durch Mobile Mapping erfassten, hochgenauen Daten ist die Darstellung des OpenDRIVE Endprodukts mit diesen beiden Programmen unzureichend. Eine realitätsnahe Visualisierung, wie sie für Simulationsanwendungen verwendet wird, ist damit nicht erreichbar. Dies stellt insbesondere bei der Präsentation der Ergebnisse und bei der Kundenakquise einen Nachteil dar. Im Vergleich dazu sind die erzeugten Zwischenprodukte visuell deutlich weniger abstrakt. Aus den Rohdaten können nach der Befahrung ausgewählte Objekte des Straßenraums wie Fahrbahnränder, Straßenmarkierungen und Grünflächen, aber auch Laternen, Ampeln, Verkehrszeichen und Bäume extrahiert werden. Als Grundlage für die Spurberechnung in OpenDRIVE werden Plandaten in einem Computer-Aided Design (CAD) System erstellt. In diesem Verarbeitungsschritt liegen die Zwischenprodukte noch als Punkte, Linien und Flächen vor und wurden noch nicht mittels linearer Referenzierung mit der Referenzlinie verknüpft. Die Weiterverarbeitung von geometrischen Primitiven ist in den meisten Visualisierungsprogrammen deutlich leichter umzusetzen und erleichtert damit den Weg zur Visualisierung der Straßendaten.

1.3 Lösungsansatz

Mit der Darstellung von Straßenräumen beschäftigte sich auch Beil [2017] in seiner Masterarbeit mit dem Titel „Detaillierte Repräsentation des Straßenraums in 3D-Stadtmodellen“. Die Darstellung des Straßenraums erfolgt unter Verwendung des Standards CityGML. Es wird ein Konzept zur CityGML-konformen Repräsentation des Straßenraums in verschiedenen Detailstufen vorgestellt und beispielhaft für ein semantisches 3D-Stadtmodell von New York City implementiert. Das verwendete Stadtmodell wurde im Rahmen mehrerer Projekte und Masterarbeiten des Lehrstuhls für Geoinformatik aus frei zur Verfügung stehenden Daten generiert. Das Ergebnis der Masterarbeit umfasst ein über das

gesamte Stadtgebiet verlaufendes CityGML-konformes 3D-Straßenraummodell. Einige Teilbereiche wurden zusätzlich texturiert und im Level of Detail (LoD) 3 dargestellt [Beil, 2017]. Diese Darstellung des Straßenraums wurde unter anderem durch den 3D City Database (3DCityDB) Web-Map-Client, wie in Abb. 1.2 abgebildet, anschaulich präsentiert. Mithilfe des Web-Clients lassen sich Inhalte von 3D-Stadtmodellen, die im CityGML-Standard vorliegen, nach einer Umwandlung ins KML-Format in jedem beliebigen Webbrowser visualisieren. Außerdem ist eine Abfrage der thematischen Daten jedes Stadtobjekts möglich. Der Einsatz der genannten Softwarelösung ist damit auch ein vielversprechender Ansatz für die Visualisierung von OpenDRIVE-Daten. Diese Möglichkeit der Visualisierung soll daher auch in dieser Masterarbeit erprobt werden.



Abbildung 1.2 – Visualisierung des semantischen 3D-Straßenraummodells der Stadt New York [Beil, 2017]

Ein weiterer Vorteil in der Darstellung mit dem Client liegt in der möglichen Verbindung des Straßenraummodells mit 3D-Gebäudemodellen. Die Verknüpfung von detaillierten, hochgenauen Straßendaten und einem virtuellen 3D-Stadtmodell ermöglicht eine realitätsnähere Präsentation der aufgenommenen Daten, welche dadurch deutlich an Anschaulichkeit gewinnen. Über die Visualisierung hinausgehend sind mithilfe des Clients Abfragen und somit den Straßenraum betreffende Analysen, wie beispielsweise die Selektion von Straßen mit einem vorgegebenen Tempolimit, möglich.

Mit den bislang eingesetzten Methoden zur Darstellung werden die durch Mobile Mapping erfassten, hochgenauen Daten nicht in ihrer gesamten Nützlichkeit und Anwendbarkeit betrachtet. Die erweiterten Möglichkeiten zur Repräsentation und Bereitstellung der Daten durch die aufgezeigte Methode eröffnen neue Anwendungsperspektiven. Beil [2017] geht umfassend auf verschiedene Perspektiven ein und stellt insgesamt 16 potentielle Anwendungsfelder vor. Er nennt neben Simulationen des Verkehrs, der Fahrdynamik und des Fahrtrainings dabei auch zahlreiche, den Straßenraum betreffende, Analysen. Dazu gehören unter anderem die Berechnung der Hitzeeinwirkung auf große Betonflächen,

Modellierung der Feinstaubbelastung in Großstädten, Abflussanalysen bei auftretender Nässe und Sichtbarkeitsanalysen für Ampeln und Verkehrszeichen. Auch das städtische Flächenmanagement und kommunale Aufgaben werden als Anwendungsfelder aufgezählt. Zudem zeigt er die jeweiligen Vorteile auf, welche ein im Bereich des Straßennetzes verbessertes 3D-Stadtmodell mit sich bringen würde. Ein detailliertes, räumlich-semantisches Modell des Straßenraums eröffnet demnach zahlreiche weitere Anwendungen jenseits der Fahrsimulation und Navigation. Dies wiederum verspricht auch die potentielle Eröffnung neuer Kundenkreise für Anbieter wie 3D Mapping Solutions und wird deshalb in dieser Arbeit erprobt.

1.4 Aufbau und Inhalt der Arbeit

Die vorliegende Arbeit setzt sich zusammen aus einem theoretischen Teil mit einer Studie verwandter Arbeiten und der Erarbeitung eines Konzeptes für eine mögliche Visualisierung sowie einer praktischen Umsetzung des erarbeiteten Ansatzes mit abschließender Evaluierung. In Kapitel 2 werden relevante Literatur und Forschungsarbeiten vorgestellt, die sich allgemein mit der Modellierung von Straßenräumen beschäftigen. Der Übersicht halber werden diese dabei in die Bereiche Anwendungsgebiete, Methoden der Datenextraktion und Datenquellen gegliedert. Kapitel 3 gibt einen Überblick über Modellierungsstandards, die eingesetzt werden um den Straßenraum zu modellieren. Insbesondere wird dabei auf die Standards OpenDRIVE und CityGML eingegangen. In Kapitel 4 wird zunächst der generelle Erfassungsprozess semantischer Straßenraummodelle mittels Mobile Mapping aufgezeigt, sowie auf das spezielle Vorgehen der Firma 3D Mapping Solutions eingegangen. An dieser Stelle wird auch das Unternehmen und deren Tätigkeitsbereiche vorgestellt. Aufbauend auf Kapitel 3 werden in Kapitel 5 OpenDRIVE und CityGML vergleichend gegenübergestellt sowie Ähnlichkeiten und Unterschiede der beiden Datenmodelle ausgearbeitet. In Kapitel 6 wird anhand der aus dem Erfassungsprozess abgeleiteten Daten von 3D Mapping Solutions eine Strategie für eine mögliche Visualisierung der MMS-Daten vorgestellt, welche in Kapitel 7 als erweiterter Auswerteprozess umgesetzt wird. Das Ergebnis wird in Kapitel 8 anhand konkreter Testgebiete realisiert und validiert. Abschließend wird in Kapitel 9 ein zusammenfassendes Fazit und ein Ausblick auf Möglichkeiten zur Weiterführung der Visualisierung der MMS-Daten gezogen.

2 Literatur und verwandte Arbeiten

Zahlreiche wissenschaftliche Artikel greifen das Thema der Straßendatenerfassung und Straßenraummodellierung vor allem im Hinblick auf die Aktualität des autonomen Fahrens und der damit verbundenen Forschung auf. Diese Arbeit gibt einen Überblick über die umfangreiche Literatur und nimmt eine Unterteilung in drei Gruppen vor. Zur ersten Gruppe gehört Literatur, die sich mit bestimmten Anwendungsfeldern und den damit verbundenen Anforderungen beschäftigt. Die zweite Gruppe beinhaltet Forschungsarbeiten, die sich mit Methoden der Informationsextraktion auseinandersetzen. Zusätzlich geben zahlreiche Arbeiten Einblicke über mögliche Datenquellen. Neben der Auflistung werden abschließend Kriterien für die Bewertung dieser Daten in Bezug auf die Straßenraummodellierung gefunden und vorgestellt.

2.1 Anwendungsfelder und Anforderungen

Beil [2017] differenziert insgesamt 16 potentielle Anwendungsfelder für die Informationen über den Verkehrsraum von Relevanz sind. Er nennt neben Simulationen des Verkehrs, der Fahrdynamik und des Fahrtrainings dabei auch zahlreiche, den Straßenraum betreffende Analysen. Dazu gehören unter anderem die Berechnung der Hitzeeinwirkung auf große Betonflächen, die Modellierung der Feinstaubbelastung in Großstädten, Abflussanalysen bei auftretender Nässe und Sichtbarkeitsanalysen für Ampeln und Verkehrszeichen. Auch das Städtische Flächenmanagement und kommunale Aufgaben werden als Anwendungsfelder aufgezählt. Zudem zeigt er die jeweiligen Vorteile auf, welche ein im Bereich des Straßennetzes verbessertes 3D-Stadtmodell mit sich bringen würde. Ein detailliertes, räumlich-semantisches Modell des Straßenraums eröffnet demnach zahlreiche weitere Anwendungen jenseits der Fahrsimulation und Navigation. Neben der Art der Informationen unterscheiden sich auch die Anforderungen an die Daten je nach Anwendungsgebiet. Im Bereich der Fahrtrainingssimulationen ist die absolute Genauigkeit weniger relevant, während diese in Anwendungsfällen wie dem Autonomen Fahren von besonderer Wichtigkeit ist. Die Bedeutung der Visualisierung verhält sich in den beiden genannten Bereichen hingegen genau gegensätzlich. Beil und Kolbe [2017] nennen folgende sieben generelle Anforderungen an die Art der Modellierung: Lagegenauigkeit, geometrische Auflösung, thematische Genauigkeit, Aktualität, Netzwerk- und Flächentopologie, zeitliche Komponente und Visualisierung.

Während Burger [2015] das Straßenraummodell als Vervollständigung der ebenfalls generierten Gebäudemodelle einführt, gehen die meisten Arbeiten zu diesem Thema jeweils auf einen bestimmten Anwendungsfall ein. Ruhdorfer [2017] beispielsweise vollzieht auf Modellebene eine Kopplung von

Verkehrssimulation und 3D-Stadtmodell. Für Verkehrs- und Fahrsimulationen notwendige Verkehrsrauminformationen und bestehende Standards und Modelle vor allem aus dem Bereich Simulation werden diskutiert und analysiert. Auch wird eine Strukturanalyse nach Art und Detaillierungsgrad der benötigten Information abhängig von der jeweiligen Simulationsanwendung erarbeitet. Offer [2016] generiert ein Straßenraummodell mit Trambahnschienen, welches als Simulationstrainer für Straßenbahnfahrer verwendet werden kann. Straßenraummodelle und hochaufgelöste Straßendaten sind auch in sehr aktuellen Themen wie dem autonomen Fahren von großer Bedeutung. In Fahrassistenzsystemen erhöhen zusätzlich eingesetzte Kartendaten des Straßennetzes die Genauigkeit der Positionsbestimmung [Siegle et al., 1992; Schreiber et al., 2013; Levinson et al., 2007]. Straßenmarkierungen [Wu und Ranganathan, 2013], Verkehrszeichen [Qu et al., 2015] und andere Straßenobjekte erleichtern die Lokalisierung des Fahrzeuges.

Neben der einfachen Visualisierung, darauf aufbauender Analysen und speziellen Forschungsanwendungen wie der Simulation werden Straßendaten auch als Grundlage für Navigationsanwendungen benötigt. Dreidimensionale Szenen stellen die abgebildete Umgebung realistischer dar und helfen bei der Orientierung. Heutzutage gehören fest integrierte Navigationsgeräte oftmals zur Grundausstattung von Fahrzeugen. Hinzu kommen Routinglösungen in Smartphones, die auf GNSS basieren und für jeden Smartphonebesitzer frei zugänglich sind. Dadurch hat sich der alltägliche Gebrauch solcher Lösungen erhöht. Hochaufgelöste Straßendaten helfen auch Fußgängern bei der eigenen Lokalisation und Navigation. Dreidimensionale Straßenraumdaten mit Kenntnissen über abgesenkte Bordsteine sind beispielsweise für Abfragen der Barrierefreiheit von Vorteil. Ein Straßenraummodell findet demnach in verschiedenen Anwendungsgebieten Verwendung.

2.2 Methoden der Informationsextraktion

Die Methoden der Informationsextraktion aus den Ausgangsdaten unterscheiden sich je nach Anwendungsfall der Straßenraummodellierung. Chaplier et al. [2010] klassifizieren die notwendigen Informationen zur Beschreibung des Straßenraums in einen topologischen, einen logischen, einen physikalischen und einen visuellen Layer. Ruhdorfer [2017] analysiert und ordnet diesen Aufbau bestehend aus vier Layern weiter ein. Einen Überblick über die Elemente der vier Ebenen gibt Abbildung 2.1. Die **topologische** Darstellung entspricht demnach einem gerichteten Graphen, der die Position und die Beziehungen zu den restlichen Elementen des Straßennetzes beschreibt. Mit **semantischen** Attributen können logische Informationen und die Bedeutung bestimmter Straßenraumobjekte definiert werden. Neben der Topologie und der Logik/Semantik ist die **Geometrie** der Fahrspuren ein wichtiger Bestandteil bei der Beschreibung des Straßenraums. Die Modellierung der Umgebung in 3D Modellen und deren Texturierung für ein realistischeres Aussehen der Objekte fällt unter den **visuellen** Layer. Die Art der Informationsextraktion hängt vom Fachgebiet und der resultierenden Anwendung ab. Die Literatur, die sich mit Methoden zur Extraktion von Straßenrauminformationen beschäftigt, kann ebenfalls annähernd in die genannten vier Ebenen eingeordnet werden.

Bei Hatger und Brenner [2003] und Hatger [2005] beispielsweise steht der Straßenkörper an sich bei



Abbildung 2.1 – Aufbau eines Straßenraummodells bestehend aus vier Ebenen [Ruhdorfer, 2017]

der Extraktion im Vordergrund. Aus den Ausgangsdaten werden neben den Mittelachsen Straßenränder extrahiert. Dadurch erhält man mit Höhe und Lage auch die Krümmung, Breite und Querneigung der Straße. Zhang [2004] und Guan et al. [2014] gehen einen Schritt weiter und erhalten durch Kantenextraktion auch Straßenmarkierungen. Die Detektion von Straßenmarkierungen und deren geometrischen Informationen aus digitalen Bildern und Videos wird jedoch durch verschiedene umgebungsbedingter Einflüsse erschwert. McCall und Trivedi [2006] nennen vier solcher Faktoren. Zum einen ist die Detektion abhängig von der Form und dem Markierungstyp, zum Beispiel Linien, Pfeile und Symbole, aber auch das Straßenoberflächenmaterial ist von Bedeutung. Die größte Auswirkung auf die Sichtbarkeit der Straßenoberfläche hat das Wetter und die Tageszeit. Zudem beeinflussen Abschattungen durch Vegetation und Fahrzeuge die Detektion der Geometrieobjekte. Diese umgebungsbedingten Einflüsse erschweren nicht nur die Detektion von Straßenmarkierungen, sondern auch die Extraktion jeglicher Objekte des Straßenraums.

Auch Straßenmöblierungsobjekte wie Ampeln und Verkehrsschilder sind für die Straßensicherheit und Straßenplanung von großer Bedeutung. Mit der Detektion von pfostenähnlichen Objekten befassen sich zum Beispiel Yokoyama et al. [2013] und Yadav et al. [2015]. Unterschieden wird teilweise auch zwischen Onground- und Offground-Objekten bzw. Ground- und Non-Ground [Pu et al., 2011; Yang et al., 2017]. Punktwolken aus Laserscanverfahren beinhalten Höheninformationen mit deren Hilfe die Straßenoberfläche von anderen Objekten abgegrenzt werden kann. Durch diese semantische Einstufung der Objekte können Gebäude, Geländer, Straßenlampen, Verkehrszeichen, Bäume und Autos von der Oberfläche differenziert werden.

2.3 Mögliche Datenquellen

Verschiedene Daten werden je nach Anwendungsgebiet und Anforderungen an die Daten als Ausgangsdaten verwendet. Zahlreiche Arbeiten geben Einblicke über mögliche Datenquellen, welche in den folgenden Unterabschnitt 2.3.1 und Unterabschnitt 2.3.2 aufgezeigt und bewertet werden.

2.3.1 Übersicht über Literatur möglicher Datenquellen

Airborne Laser Scanning (ALS) ist eine häufig verwendete Methode zur Datenaufnahme. Narwade und Musande [2014] legen eine ausführliche Dokumentation von fast 50 Referenzen zu diesem Thema dar. Punkte einer Lidarpunktwolke beinhalten Höhen- und Intensitätsinformationen. Intensitätswerte können zur Kantendetektion wie Straßenränder und Markierungen verwendet werden [Hatger, 2005], Höheninformationen hingegen sind für die Klassifizierung zwischen Straße und Straßenobjekten geeignet [Clode et al., 2004; Yadav et al., 2015].

In den meisten Fällen werden jedoch verschiedene Möglichkeiten der Datenaufnahme kombiniert, um aus den Vorteilen der jeweiligen Aufnahmemethode zu profitieren und den Informationsgehalt des Straßenmodells zu erhöhen. Die Verbindung von zweidimensionalen Vektordaten und Laserscandaten ermöglicht eine genaue Repräsentation der Mittelachsen und Straßenränder mit exaktem Digitalem Oberflächenmodell der Straße [Hatger und Brenner, 2003; Chen et al., 2006; Chen und Lo, 2009; Segerström, 2015]. Während Geodaten dabei die Lageinformationen der Straße liefern, ermöglichen Laserscandaten die Informationen über die Höhe und Intensität der Straße. Geodaten können zum einen von den Vermessungsverwaltungen bereitgestellte amtliche Geobasisdaten sein, zum anderen frei verfügbare Offene Daten. Burger [2015] und Beil [2017] verwenden Offene Daten der Stadt New York und generieren daraus ein über das gesamte Stadtgebiet verlaufendes 3D-Straßenraummodell mit texturierten, CityGML-konformen LoD 3 Straßenraumteilbereichen. Das Straßenraummodell mit Trambahnschienen von Offer [2016] wurde aus OpenStreetMap (OSM)-Daten erzeugt.

Hu et al. [2004] verwenden zu den Laserscandaten zusätzlich hochaufgelöste Multispektralluftbilder. Durch die spektralen Informationen der Luftbilder können Punkte, die nicht dem Straßenkörper zugewiesen werden können, herausgefiltert werden. Das Problem der Abschattung durch Vegetation und fahrenden Fahrzeugen kann so reduziert werden. Photogrammetrische Methoden der Kantenextraktion zur Detektion des Straßenkörpers sowie der Detektion von Markierungen und Gebäuden werden auch in Stereoluftbildern angewendet [Suveg und Vosselman, 2004; Doucette et al., 2004]. Neben dem ALS kommen auch terrestrische Mobiles Laser Scanning (MLS) als Datenquelle zum Einsatz [Pu et al., 2011; Yang et al., 2013; Kumar et al., 2015]. Die Genauigkeit ist dabei höher und Abschattungen durch Gebäude, Fahrzeuge und Vegetation können vermindert oder sogar vermieden werden. MMS setzen neben Laserscannern auch Kameras als Aufnahmemodelle zur Datenerfassung ein [Tao, 2000; Guan et al., 2014].

Tao und Li [2007] geben einen Überblick über verschiedene Forschungsarbeiten zu diesem Thema. Dabei erfolgt eine Einteilung in die Themenbereiche Terrestrisches Laser Scanning (TLS) und luftgestützte Mappingsysteme, Integrationsmöglichkeiten von Multisensorsystemen, Bildverarbeitung und

Objektextraktion und mobile Geoinformationssysteme. Dabei wird auch auf Schwarz und El-Sheimy [2007] verwiesen, die sich mit digitalen, mobilen Mappingsystemen beschäftigen. Unter digitalen, mobilen Mappingsystemen werden Systeme eingeordnet, die Kamerasensoren und Sensoren für die Georeferenzierung beinhalten.

2.3.2 Kriterien zur Beurteilung genannter Datenquellen

Zusammenfassend lassen sich aus der Übersicht in Kapitel 2.3.2 folgende Datenquellen ableiten:

- Offene Daten (OSM)
- Amtliche Geobasisdaten
- Terrestrische Bilddaten
- Hochaufgelöste Luftbilder und Satellitenbilder
- Punktwolken aus Scanning-Verfahren (TLS, ALS, MLS)
- Punktwolken aus photogrammetrischen Verfahren (Basis: Terrestrische Bilddaten)
- Punktwolken aus photogrammetrischen Verfahren (Basis: Luft- und Satellitenbilder)

Die Open Knowledge Foundation Deutschland definiert **Offene Daten** als Daten, die von jedermann frei verwendet, genutzt und verbreitet werden können. Die Nennung der Quelle und Lizenzangabe sind verpflichtend, wobei personenbezogene Daten davon ausgenommen sind. Es werden eine Vielzahl von unterschiedlichen Datenarten gesammelt. Dazu gehören unter anderem Lageinformationen von Straßen und Gebäuden, Kulturdaten über kulturelle Werke, aber auch Wetterdaten und andere Daten innerhalb einer wissenschaftlichen Forschung [OKFD, 2016]. Im Folgenden werden unter diesem Begriff OSM-Daten verstanden. OSM ist ein Projekt für die Bereitstellung von frei nutzbaren Geodaten zur Erstellung von Karten, insbesondere Straßenraumkarten. Darunter fallen Informationen zur Lage der Straße und den Objekten entlang und auf einer Straße, sowie Topographien und Grenzen [Open-StreetMap, 2018a]. **Geobasisdaten** sind amtliche Geodaten, die von den Vermessungsverwaltungen bereitgestellt werden. Eine weitere Unterteilung in zum Beispiel topographische Daten, Katasterdaten, digitale Höhenmodelle und Digitale Orthophotos (DOP) ist möglich. Interessant für die Beurteilung von Geobasisdaten sind für die Straßenraummodellierung vor allem die Datenmodelle Amtliches Topographisch-Kartographisches Informationssystem (ATKIS) und Amtliches Liegenschaftskataster-Informationssystem (ALKIS) [LDBV, 2018a]. Unter **Terrestrischen Bilddaten** werden für die folgende Bewertung statisch und mobil erfasste terrestrische Daten verstanden. Kamerasysteme von MMS werden dadurch zum Beispiel miteinbezogen. Albertz [2001] bezeichnet **Luftbilder** als photographische Bilder, die von Luftfahrzeugen, in der Regel Flugzeuge, aufgenommen werden. **Satellitenbilder** sind dagegen Bilder, die vom All aus aufgenommen werden, entweder von Satelliten oder bemannten Plattformen (z.B. Space Shuttle). Diese Definition wird im Folgenden übernommen. **Laserscanning-Verfahren** unterscheiden sich in der Art der Punktwolkenaufnahme, welche terrestrisch, luftgestützt

oder mobil erfolgen kann. Die Unterscheidung der Laserscanning-Verfahren in TLS, ALS und MLS erleichtert die Beurteilung der Punktwolken nach den im Folgenden aufgezeigten Kriterien, wobei unter TLS die stationäre, terrestrische Aufnahme verstanden wird. **Punktwolken aus photogrammetrischen Verfahren** müssen nach zugrundeliegendem Ausgangsbild bewertet werden. Als Ausgangsbild können wiederum terrestrisch aufgenommene Bilddaten oder Luft- und Satellitenbilder verwendet werden.

Jede der aufgezeigten Datenquellen kann für unterschiedliche Arten der Informationsextraktion verwendet werden und hat dementsprechend Vor- und Nachteile. An dieser Stelle sollen Kriterien für die Bewertung der genannten Quellen in Bezug auf die Straßenraummodellierung gefunden werden. Insgesamt werden folgende neun Kriterien für die bessere Beurteilung der Datenquellen vorgestellt:

1. Verfügbarkeit
2. Aktualität
3. Kosten
4. Mess- und Extraktionsmethode
5. Semantischer Informationsgehalt
6. Verarbeitungs- und Interpretationsaufwand
7. Korrektheit (Zuverlässigkeit)
8. Vollständigkeit und räumliche Abdeckung
9. Absolute und relative Genauigkeit

Alle dazugehörigen Erkenntnisse werden zusammengefasst in Tabelle 2.1 und Tabelle 2.2 dargestellt und im Folgenden erklärt. Mit * versehene Angaben beruhen auf Erfahrung oder der Diskussion mit Fachleuten.

1. Verfügbarkeit

Ein Hauptkriterium von Offenen Daten ist die freie Verfügbarkeit [OKFD, 2016]. Zahlreiche Benutzer vervollständigen den Bestand der OSM-Daten durch GNSS-Aufnahmen oder durch die Angabe von Informationen zu bestimmten Orten. Auch Firmen und Institutionen kooperieren mit OSM und geben Karteninformationen und Straßendaten frei. Satellitenbilder von Yahoo und Microsoft (Bing Maps) beispielsweise dürfen in OSM genutzt werden. Das Landesvermessungsamt Bayern hat für ein Pilotprojekt flächendeckend qualitativ hochwertige detaillierte DOP freigegeben [OpenStreetMap, 2018b]. Die Bayerische Staatsregierung bietet mit dem Open Data Bayern Portal den Nutzern die Möglichkeit, gezielt nach frei zugänglichen Daten der staatlichen Behörden in Bayern zu suchen. Die bayerische Vermessungsverwaltung stellt unterschiedliche Geobasisdaten und Luftbildaufnahmen auf dem Geoportal online zur Verfügung. Das Geoportal beinhaltet Daten aus dem Liegenschaftskataster wie digitale Flugkarten, topographische Karten mit Höhendaten und Luftbildprodukte wie DOP [Bayerische Vermessungsverwaltung, 2018]. Luft- und Satellitenbilder sind grundsätzlich für wenige bestimmte Zeitpunkte verfügbar, werden aber in den meisten Fällen extra für bestimmte Anwendungsfälle erzeugt. Alle anderen Quelldaten stehen im Allgemeinen nicht grundsätzlich zur Verfügung und werden explizit projektbezogen aufgenommen. Der Erzeugungsaufwand für die Neuaufnahme kann dabei stark variieren und hängt von der Aufnahmedauer und dem gewünschten Aufnahmegebiet ab. Zur Erzeugung von terrestrischen Kameradaten werden lediglich ein Aufnahmegerät, eine Einzelperson und möglicherweise

eine mobile Plattform benötigt. Der Aufwand von terrestrischen Laserscanaufnahmen kann minimal großer eingeschätzt werden, da die Standpunktwahl und Planung auch zeitintensiver ist. Der Erzeugungsaufwand von Luft- und Satellitenbildern, ALS und MLS kann als sehr hoch eingestuft werden, da in diesen Fällen meist eine umfangreiche Kampagnenplanung, teilweise große Messaufbauten und Einsatz von teuren Geräten und Spezialisten mit inbegriffen ist. Dafür können im Vergleich zur Aufnahme von terrestrischen Bilddaten und dem TLS großflächige Gebiete in kurzer Zeit aufgenommen werden. Der Erzeugungsaufwand von Punktwolken aus photogrammetrischen Verfahren ist hauptsächlich von den zugrundeliegenden Bilddaten abhängig.

2. Aktualität

Eng mit der Verfügbarkeit ist die Aktualität der Ausgangsdaten verbunden. Alle Daten, die in der Regel explizit für ein bestimmtes Projekt aufgenommen werden, sind demnach auch projektbezogen sehr aktuell bzw. tagesaktuell.

Die Aktualität der Offenen Daten ist stark abhängig von Ort und Datenart und somit schwer einschätzbar. Grundsätzlich können neue Einträge und Änderungen aber in wenigen Minuten auf der Karte erscheinen [OpenStreetMap, 2018a]. Die Aktualität der Geobasisdaten ist auch abhängig von der Art der Daten. Das amtliche Liegenschaftskatasterbuch dient als flächendeckender Nachweis aller Liegenschaften in Bayern und wird ständig aktualisiert [Bayerische Vermessungsverwaltung, 2018]. Die bayerische Vermessungsverwaltung befliegt in einem Turnus von zwei Jahren abwechselnd Nord- und Südbayern. Aus den daraus gewonnenen Luftbildern werden DOP abgeleitet, deren Aktualität demnach auch auf zwei Jahre geschätzt werden kann. Die Informationen aus ALKIS-Daten wie Flurkarten, Gebäude und Eigentümer werden laut Bayerischer Vermessungsverwaltung laufend aktualisiert und sind tagesaktuell. Einige Informationen wie Bodenschätzwertdaten werden halbjährlich aktualisiert. Bestimmte Objekte der ATKIS-Daten werden mindestens alle drei Monate erneuert.

3. Kosten

Offene Daten sind für den Nutzer frei zugänglich und können meist ohne Einschränkungen weiterverwendet werden [OpenStreetMap, 2018a]. Geobasisdaten werden von der Bayerischen Vermessungsverwaltung entweder flächenbezogen oder objektbezogen abgerechnet. Eine ALKIS-Komplettabgabe mit Flurstücken, Gebäuden, Tatsächlicher Nutzung, Bodenschätzung und Eigentümer liegt bei 3,90 € pro Flurstück. Dreidimensionale Gebäudemodelle im LoD 2 liegen bei 0,65 € pro Objekt. Vektordaten des Amtlichen Topographischen Karteninformationssystem ATKIS können pro Quadratkilometer gekauft werden. Luftbilder aus digitaler Befliegung werden mit 18 € pro km² berechnet [LDBV, 2018b].

Die Kosten von Terrestrischen Bilddaten belaufen sich vor allem auf den einmaligen Anschaffungspreis der Geräte und möglichen Kalibrierungskosten. Die Aufnahme ist eher kostengünstig, kann aber abhängig von der Größe des Gebiets sehr zeitintensiv sein. Luft- und Satellitenbilder können als Offene Daten kostenlos zur Verfügung stehen. Satellitenbilder von Yahoo und Microsoft (Bing Maps) oder freigegebene Luftbildaufnahmen der Bayerischen Vermessungsverwaltung werden in OSM bereitgestellt [OpenStreetMap, 2018b]. Die Kosten von nicht frei zur Verfügung stehenden Luft-

und Satellitenbilder werden hoch eingeschätzt. Die Erfassung von Straßendaten mittels ALS- und MLS-Systemen ist zeitaufwändiger und planungsintensiver. Hohe Kosten treten bei ALS durch die Befliegung, die üblicherweise von einem Unterauftragnehmer durchgeführt werden, auf. Die Bayerische Vermessungsverwaltung gibt als Preis für Laserscanpunkt klassifiziert in mehreren Klassen 100 % des Preises des genauesten in dem Gebiet verfügbaren Digitalen Geländemodells (DGM) an (80 € pro km²) an [LDBV, 2018b]. Die Kosten von ALS und MLS sind generell schwer einzuschätzen, da die meisten kommerziellen Anbieter keine Angaben über genaue Preise pro Quadratkilometer oder Kilometer geben. Außerdem werden die Daten selten in Rohdatenform weiterverkauft, sondern erst vorverarbeitet und georeferenziert. Trotz allem wird zusätzlich zu einem hochauflösenden Laserscanner eine Befestigung und Plattform benötigt. Meistens werden zusätzliche Sensoren zur Positionsbestimmung installiert. Diese Faktoren und eine zeitaufwändigere Planung erhöhen die Kosten im Vergleich zu TLS-Verfahren. Im Vergleich dazu werden solche Systeme aber häufig als sehr kosteneffizient gesehen, da in relativ kurzer Zeit große Gebiete aufgenommen werden können. Die genannten Faktoren gelten auch für die Aufnahme von Luft- und Satellitenbilder. Auch die Kosten für Punktwolken aus photogrammetrischen Verfahren sind abhängig von den Grundlagendaten.

4. Mess- und Extraktionsmethode

Vorliegende Quelldaten unterscheiden sich auch durch die Mess- und Extraktionsmethode. Direkt gemessen bedeutet dabei, dass die gewünschte Klasse bereits klassifiziert gemessen wurde und direkt aus den Messdaten entnommen werden kann. Bei einer indirekten Messmethode sind noch zusätzliche Methoden für die spätere Segmentierung notwendig. Offene Daten und Geobasisdaten werden direkt und punktuell gemessen. Informationen können demnach auch direkt extrahiert werden. Alle anderen Quelldaten werden indirekt aufgenommen. Für die Weiterverarbeitung der Daten sind zusätzliche Segmentierungs- und Extraktionsalgorithmen notwendig. Während Luft- und Satellitenbilder und Punktwolken aus ALS die Umgebung dabei flächendeckend wiedergeben, decken Terrestrische Bilddaten und Punktwolken aus TLS die Standpunktumgebung ab. MLS nehmen die Umgebung um die Trajektorie flächendeckend auf.

5. Semantischer Informationsgehalt

Dieses Kriterium beschreibt wie viel Semantik explizit in den Daten repräsentiert wird. Die Semantik von Offenen Daten kann regional stark variieren und wird deshalb mit mittelmäßig bis hoch angegeben. In urbanen Gebieten beispielsweise sind teilweise neben Straßendaten, auch Straßennamen, Gebäudeinformationen wie Hausnummern und interessante Orte eingetragen. Dieser semantische Informationsgehalt kann in ländlichen Gebieten meist nicht oder noch nicht angeboten werden [OpenStreetMap, 2018b]. Der semantische Informationsgehalt von Geobasisdaten ist bundesweit flächendeckend hoch. Die Liegenschaftskatasterdaten beinhalten unter anderem Angaben zu Flurstücken, 3D Gebäudedaten und Bodenschätzwerten [LDBV, 2018b]. In allen anderen Datenquellen kann der Wert als gering angenommen werden.

6. Verarbeitungsaufwand und Interpretationsaufwand

Außerdem muss zwischen dem Verarbeitungs- und Interpretationsaufwand separiert werden. Offene Daten und Geobasisdaten bestehen oftmals aus mehreren Klassen und Attributen, deren Verar-

beitungsaufwand wesentlich größer erscheint als der von einzelnen Punktkoordinaten oder Bilddateien. Der Interpretationsaufwand und die damit verbundene Extraktion der gewünschten Daten verhält sich jedoch genau gegensätzlich. Mit relativ einfachen Abfragen können bestimmte Daten aus Offenen Daten und Geobasisdaten segmentiert und gefiltert werden. Im Vergleich dazu werden zur Extraktion von Gebäudedaten aus Kamerabildern und Punktwolken mathematische Algorithmen, spezielles Wissen und besondere Software benötigt.

7. Korrektheit und Zuverlässigkeit

Die Korrektheit und Zuverlässigkeit der Offenen Daten ist ortsabhängig und datenabhängig unterschiedlich. Während die von Verwaltungen bereitgestellten frei zugänglichen Daten als zuverlässig angenommen werden können, können von Privatpersonen aufgenommene Daten fehlerhaft sein. Zusätzlich ist die Zuverlässigkeit auch mit der Aktualität der Daten verbunden, die vor allem bei Offenen Daten ortsabhängig variiert. Aus diesen Unsicherheiten lässt sich eine geringe bis mittelmäßige Zuverlässigkeit der Daten festhalten. Im Vergleich dazu ist die Zuverlässigkeit von Geobasisdaten sehr hoch und wird höchstens von der Aktualität der Daten eingeschränkt.

Punktwolken aus Scanning-Verfahren können grundsätzlich als korrekt angenommen werden. Dennoch wird die Zuverlässigkeit durch Spiegelung an bestimmten Oberflächen vermindert. Oberflächenpunkte können zum Beispiel durch erfasste Pfützen auf der Fahrbahn oder durch die Aufnahme von Fensterfronten falsche Koordinatenwerte enthalten. Die Korrektheit von TLS-Daten wird minimal höher eingeschätzt als die von ALS und MLS, da die Aufnahme von bestimmten Flächen leichter vermieden und die Zuverlässigkeit somit erhöht werden kann. Die genannten Einflüsse gelten auch für terrestrische und luftgestützte Bildaufnahmen. Die Korrektheit von Punktwolken aus bildbasierenden Verfahren ist abhängig von den Texturen der aufgenommenen Objekte. Sehr glatte Oberflächen ohne Texturen können zu Fehlern in den Bilddaten und damit fehlerbehafteten Punktwolken führen. Im Vergleich zu Offenen Daten deren Korrektheit schwer abschätzbar ist sind Laserscanning-Verfahren und bildbasierte Verfahren dennoch zuverlässiger.

8. Vollständigkeit und räumliche Abdeckung

Eng mit der Zuverlässigkeit ist die Vollständigkeit und räumliche Abdeckung verbunden. In den OSM-Daten sind für Deutschland auch im ländlichen Bereich zumindest die wichtigsten Straßen flächendeckend vorhanden [OpenStreetMap, 2018a]. Alle größeren Städte sind nahezu komplett erfasst. Grundsätzlich ist die räumliche Abdeckung allerdings regional schwankend. Der Begriff „vollständig“ in diesem Zusammenhang ist schwer eingrenzbar, da auch eine Karte eines räumlich gut abgedeckten Straßennetzes immer weiter verfeinert werden kann. Außerdem ist die räumliche Abdeckung durch bereitgestellte Luft- und Satellitenbilder deutlich höher als die Abdeckung von durch GNSS gemessene Straßenstücke. Die Vollständigkeit ist demnach zum einen abhängig vom Detaillierungsgrad und der Aktualität der Daten, aber auch von der Aufnahmeart der Daten. Alle verfügbaren Geobasisdaten sind räumlich komplett abgedeckt und sind höchstens aus Aktualitätsgründen in manchen Teilen unvollständig. Die ALKIS- und ATKIS-Daten stehen für Bayern flächendeckend zur Verfügung [LDBV, 2018b].

Die Vollständigkeit von Kameradaten und TLS-Daten ist vor allem kampagnenabhängig. Grundsätzlich

sind vollständige Daten und eine räumliche Abdeckung durch eine gut gewählte Standpunktwahl bei der Aufnahme möglich. Auch für Luft- und Satellitenbilder sowie Punktwolken aus Scanning-Verfahren ist die Vollständigkeit zum einen kampagnenabhängig, kann aber zum anderen aufgrund von Abschattungen wie Vegetation oder Fahrzeugen Lücken aufweisen. Im Vergleich zu Luftbildern kann die Vollständigkeit von Satellitenbildern durch Abschattung aufgrund von Wolken beschränkt sein. Flugzeuge zur Luftbildaufnahme können bei geeignetem Wetter und unter der Wolkendecke fliegen. Die Vollständigkeit von MLS-Daten sind abhängig von der gefahrenen Trajektorie. Innenhöfe und unzugängliche Gebiete werden meist nicht aufgenommen. Die Abschattung von Bäumen entfällt zwar meistens bei der mobilen Erfassung mit dem PKW, dennoch bleibt die Abschattung aufgrund parkender Autos.

9. Geometrische Genauigkeit

Bei der Geometrischen Genauigkeit muss zwischen absoluter und relativer Genauigkeit unterschieden werden. Während die absolute Genauigkeit von der Georeferenzierung abhängig ist, wird die relative Genauigkeit an der Präzision des Messgeräts gemessen. Die absolute Genauigkeit ist abhängig von der verwendeten Georeferenzierungsmethode, die beispielsweise durch die Kombination verschiedener Verfahren wie GNSS und Passpunkte erhöht wird. Statische Laserscanmethoden besitzen im Vergleich zu mobilen Verfahren eine höhere relative Genauigkeit, wodurch auch eine bessere absolute Genauigkeit erreicht werden kann.

OSM-Daten werden größtenteils mittels GNSS-Geräten aufgenommen. GNSS-Ungenauigkeiten im Straßennetz können durch das Gerät selbst, aber auch beispielsweise durch das Wechseln der Straßenseite bei der Aufnahme auftreten. Die Lagegenauigkeit der GNSS-Daten wird mit etwa 10 m angegeben, was für 99% der Messungen zutrifft [OpenStreetMap, 2018b]. Neis et al. [2010] geben einen Überblick über verschiedene Untersuchungen zur Genauigkeit von OSM-Daten. Demnach werden von Jahr zu Jahr immer mehr und immer bessere Daten bereitgestellt. Genauigkeitswerte liegen je nach Straßentyp zwischen 3-8 m.

Die Genauigkeiten der amtlichen Geobasisdaten werden mit absoluten Werten angegeben. Für die ATKIS-Daten wird eine Genauigkeit von +/- 3 Meter für Straßen und Schienen und eine maximale Genauigkeit von +/- 15 Meter bei untergeordneten Objekten angegeben. Die Lagegenauigkeit der dreidimensionalen Gebäudemodelle entspricht der Genauigkeit der zugrundeliegenden Gebäudegrundrisse aus dem ALKIS-Datensatz und wird mit ca. 5 m im LoD1 und ca. 1 m für das LoD2 angegeben. Amtliche ALS-Daten erreichen im ebenen, freien Gelände eine Höhengenaugigkeit von 0,1 m und eine Lagegenauigkeit von 0,50 m [LDBV, 2018b].

Für die Beurteilung der genannten Datenquellen sind verschiedene Kriterien notwendig. Grundsätzlich müssen vor der Verwendung bestimmter Daten Vor- und Nachteile für den bestimmten Anwendungsfall überprüft werden. Eine Verknüpfung der Datenquellen ermöglicht beispielsweise die Verwendung semantisch umfangreicherer Daten mit höherer räumlichen Abdeckung. Darin lässt sich auch der Vorteil von MMS ableiten, die durch eine Kombination verschiedener Verfahren Umgebungs- und Positionsdaten aufnehmen.

Tabelle 2.1 – Mögliche Kriterien zur Beurteilung genannter Datenquellen (Teil 1)

Kriterien	Offene Daten (OSM)	Amtiliche Geobasisdaten	Terrestrische Bilddaten	Hochaufgelöste Luft- und Satellitenbilder
Verfügbarkeit	Ja	Ja	Nein, nur projekt-bezogene Aufnahme	Ja für wenige Zeitpunkte
Aktualität	Mittel	Hoch bis sehr hoch	Erzeugungsaufwand gering	Erzeugungsaufwand sehr hoch
Kosten	-	Mittel	Sehr hoch	Gering oder Sehr hoch
Mess- und Extraktionsmethode *	Direkt (punktuell)	Direkt (punktuell)	Indirekt (deckt Stand-punktsicht ab)	Indirekt (flächendeckend)
Semantischer Informationsgehalt	Mittel bis Hoch (ortsabhängig)	Hoch (flächendeckend)	Gering	Gering
Verarbeitungs- und Interpretationsaufwand	Hoch, gering	Hoch, gering	Gering, hoch	Gering, hoch
Korrektheit und Zuverlässigkeit *	Gering bis Mittel (ortsabhängig und datenabhängig)	Sehr hoch	Hoch	Mittel (abhängig von Texturen, Spiegelung,...)
Vollständigkeit und räumliche Abdeckung	Mittel (ortsabhängig und daten-abhängig)	Sehr hoch	Hoch	Mittel bis Hoch (lückenlos, aber Abschattung)
Geometrische Genauigkeit (absolut)	Mittel bis hoch	Hoch (datenabhängig)	Mittel bis hoch	Mittel
Geometrische Genauigkeit (relativ)	Mittel bis hoch	-	Hoch	Mittel bis hoch

* dieser Wert beruht auf Erfahrung oder Diskussion mit Fachleuten

Tabelle 2.2 – Mögliche Kriterien zur Beurteilung genannter Datenquellen (Teil 2)

Kriterien	Punktwolken aus Scanning-Verfahren:			Punktwolken aus photogrammetrischen Verfahren:	
	TLS	ALS	MLS	Basis: Terrestrische Bilder	Basis: Luft- und Satelliten- bilder
Verfügbarkeit	Nein, nur projekt- bezogene Aufnahme	Nein, nur projekt- bezogene Aufnahme	Nein, nur projekt- bezogene Aufnahme	Nein, nur projekt- bezogene Aufnahme	Nein, nur projekt- bezogene Aufnahme
Aktualität	Erzeugungsaufwand mittel	Erzeugungsaufwand sehr hoch	Erzeugungsaufwand sehr hoch	Erzeugungsaufwand mittel bis hoch	Erzeugungsaufwand mittel bis hoch
Kosten	Sehr hoch	Sehr hoch	Sehr hoch	Sehr hoch	Sehr hoch
Mess- und Extraktionsmethode *	Hoch	Indirekt (flächendeckend)	Indirekt (deckt Trajektorie ab)	Gering	Hoch
Semantischer Informationsgehalt	Indirekt (deckt Stand- punktsicht ab)	Indirekt (flächendeckend)	Indirekt (deckt Trajektorie ab)	Indirekt (deckt Stand- punktsicht ab)	Indirekt (flächendeckend)
Verarbeitungs- und Inter- pretationsaufwand *	Gering *	Gering *	Gering *	Gering *	Gering *
Korrektheit und Zuverlässigkeit *	Gering, sehr hoch	Gering, sehr hoch	Gering, sehr hoch	Gering, sehr hoch	Gering, sehr hoch
Vollständigkeit und räumliche Abdeckung	Hoch (abhängig von Spiegelung,...)	Mittel (abhängig von Spiegelung,...)	Mittel (abhängig von Spiegelung,...)	Hoch (abhängig von Texturen, Spiegelung,...)	Mittel (abhängig von Texturen, Spiegelung,...)
Geometrische Genauigkeit (absolut)	Mittel (abhängig von Standpunktzahl)	Mittel bis Hoch (lückenlos, aber Abschätzung)	Mittel (abhängig von Trajektorie)	Mittel bis hoch	Mittel siehe Luftbilder und Satellitenbilder
Geometrische Genauigkeit (relativ)	Hoch	Mittel bis hoch	Mittel bis hoch	Mittel bis hoch	Mittel

* dieser Wert beruht auf Erfahrung oder Diskussion mit Fachleuten

3 Möglichkeiten zur Modellierung des Straßenraums

Aus vielfältigen Anwendungsbereichen – von einfachen Infrastrukturaufgaben bis hin zu komplexen Simulationen – haben sich unterschiedliche, anwenderspezifische Standards für die Straßenraummodellierung entwickelt. In manchen Bereichen unterscheiden sich diese Standards, in anderen sind sie sich sehr ähnlich oder gehen sogar auseinander hervor. Während sich das vorhergehende Kapitel mit wissenschaftlichen Artikeln aus dem Bereich der Straßenraummodellierung beschäftigt, soll an dieser Stelle ein Überblick über bestehende Modellierungsstandards und deren primären Anwendungsbereiche geschaffen werden.

Bewusst beinhaltet dieses Kapitel ausschließlich Modellierungsstandards und verzichtet aus zwei Gründen auf die zusätzliche Vorstellung von Visualisierungsstandards. Die Kooperation mit 3D Mapping Solutions schließt einerseits die genaue Auseinandersetzung mit dem parametrischen Standard OpenDRIVE ein, der sich als Zielformat von Kunden der Automobilindustrie etabliert hat. Andererseits ist mit Modellierungsstandards wie CityGML auch eine detailreiche Visualisierung möglich. Durch vorhandene Open Source Werkzeuge wie dem 3D Web-Map-Client der 3DCityDB sind die Visualisierungsmöglichkeiten von CityGML sehr hoch. Zum Vergleich sollen deshalb an dieser Stelle Modellierungsstandards genannt werden, die sich mit dem Straßenraums auseinandersetzen. Auf die für diese Arbeit besonders relevanten Teile der beiden Standards CityGML und OpenDRIVE wird detailliert in separaten Kapiteln eingegangen. Die genaue Betrachtung dieser beiden Standards wird als Grundlage für das folgende Kapitel 5 dienen.

3.1 Überblick über Modellierungsstandards

Der folgende Abschnitt stellt eine Auswahl an unterschiedlichen, anwenderspezifischen Modellierungsstandards zur Repräsentation des Straßenraums vor. Zumeist sind diese Standards aus ihren primären Anwendungsgebieten wie der Navigation, Simulation, Oberflächenbeschreibung und Stadtmodellierung entstanden. Die Standards OpenDrive und CityGML werden detaillierter in anschließenden Unterabschnitten beschrieben.

OpenCRG

OpenCRG ist als ein komplementäres Projekt zu OpenDRIVE zu verstehen und wurde ebenfalls von der VIRES Simulationstechnologie GmbH als Open-Source-Projekt entwickelt. Ziel des Projekts

ist die Bereitstellung einer Reihe von offenen Datenformaten und Open-Source Werkzeugen für die detaillierte Beschreibung, Erzeugung und Auswertung von Straßenoberflächen. Während sich der OpenDRIVE-Standard auf die Beschreibung von Straßennetzwerken in Bezug auf die Logik des Straßenverlaufs konzentriert, werden durch den OpenCRG-Standard ausgewählte Straßenoberflächen beschrieben. Diese Oberflächendaten finden in der Fahrsimulation, sowie bei Reifen- und Vibrationssimulationen Anwendung und erweitern in den meisten Fällen Teile der OpenDRIVE-Straßendaten mit detaillierten Oberflächeninformationen wie Bodenwellen und Schlaglöcher [Dupuis et al., 2010].

IFC-Alignment

Der Industry Foundation Classes-Standard (IFC-Standard) ist auf die Bauprojektentwicklung ausgerichtet. Alle Informationen eines Projektzyklus werden in einem einheitlichen Informationsmodell (IFC-Produktmodell) abgebildet [Ji et al., 2008]. Das IFC-Alignment-Produktdatenmodell als Erweiterung von IFC4 dient der Beschreibung von Straßen- bzw. Bahntrassierungen. Es wurde in Zusammenarbeit von buildingSMART und des OGC umgesetzt und dient als Basis für weitere Projekte wie IFC-Bridge und IFC-Road. Der Standard wird während des gesamten Lebenszyklus eines Infrastrukturbauwerks – von der Planung bis hin zur Wartung – eingesetzt und findet somit vor allem in Building Information Modelling (BIM) Systemen Anwendung [Amann et al., 2015].

LandXML

LandXML ist ein XML-basiertes Datenformat zum Austausch von Straßenentwurfsdaten. Die erste Version wurde von der Firma Autodesk im Jahr 2001 veröffentlicht und wurde bis 2014 weiterentwickelt. Seit 2014 wird die Weiterentwicklung von LandXML von Carson Software gesponsert. Mit dem vor allem im Bereich des Hochbaus und der Vermessung verbreiteten Standard können hauptsächlich Vermessungs- und Infrastrukturdaten übergeben werden [Ji, 2014]. InfraGML und der Land and Infrastructure Conceptual Model Standard (LandInfra), die beide im Folgenden genauer erklärt werden, sind eine Art Neuformulierung und Erweiterung des LandXML-Standards unter Verwendung des OGC-Standards und der ISO191xx.

LandInfra

LandInfra wurde 2016 vom OGC veröffentlicht. Mit LandInfra wurde ein zu LandXML alternativer Standard entwickelt, der der Struktur eines OGC-Standarddokuments folgt. Unter anderem ist dadurch das Dokument für Standard- und Software-Entwickler leichter lesbar. LandInfra basiert auf Teilen des LandXML-Standards, wird aber durch ein UML-Konzeptmodell unterstützt. Der Standard eignet sich für die gesamte Lebenszyklusanalyse eines Infrastrukturbauwerks und schließt damit alle Bereiche von der Planung, über das Design und den Bau, bis hin zu Instandhaltung und Rückbau ein. Zielgruppen und Anwendungsgebiete finden sich im Ingenieurbau, in der Vermessung, sowie im Gebäudemanagement und der Vermögensverwaltung [Scarponcini, 2016]. LandInfra und CityGML überlappen sich in einigen Bereichen. Viele der in LandInfra definierten Features, wie zum Beispiel Gebäude und Straßen, kommen als Unterklassen in der CityGML Klasse CityObject vor. Das Alignment-Konzeptmodell von LandInfra wurde gemeinsam mit dem IFC-Alignment-Konzeptmodell entwickelt [Gruler et al., 2016].

InfraGML

InfraGML ist ein vom OGC in Kooperation mit buildingSMART entwickeltes, auf der Geography Markup Language (GML) basierendes, Produktdatenmodell. InfraGML basiert auf LandInfra-Konzepten und setzt diese als GML-Anwendungsschema um. Dadurch basiert auch InfraGML auf Teilen des LandXML-Standards. Im Gegensatz zu LandXML ist der Standard anwendungsorientiert, mit dem OGC-Standard vereinbar und wird durch ein Unified Modeling Language (UML) Konzeptmodell unterstützt [Gruler et al., 2016]. Als Grundlage des Standards im Bereich der Straßenmodellierung dient IFC-Alignment. Allerdings werden neben der reinen Trassierung auch Anwendungsfälle wie Landvermessung oder Liegenschaftsverwaltung unterstützt. Im Unterschied zu IFC-Alignment wird InfraGML häufig für die Abbildung des Bestands in Geographischen Informationssystemen (GIS) eingesetzt. Durch die beiden Standards soll eine Verbindung zwischen der BIM- und GIS-Welt geschaffen werden [Amann, 2015].

RoadXML

Das RoadXML-Format ist ein XML-basiertes Dateiformat, welches durch ein internationales Gremium bestehend aus INRETS, Oktal, PSA, Renault und Thales veröffentlicht wurde. Es resultiert aus der Zusammensetzung verschiedener etablierter Formate und wurde auf die Bedürfnisse von Fahrsimulatoren abgestimmt. RoadXML ist flexibel und erweiterbar um Nutzern die Möglichkeit zu geben, die Straßennetzbeschreibung mit anwenderspezifischen und firmeneigenen Daten anzureichern. Im RoadXML-Format ist ein Straßennetz eine Zusammensetzung aus Teilnetzen, die jeweils eine Sammlung von Kreuzungen und Pfaden darstellt. Eingesetzt wird RoadXML in der akademischen und industriellen Forschung und Entwicklung hauptsächlich im Bereich der Fahrsimulation und Fahrtdynamik. Dazu gehört unter anderem die Erzeugung von dreidimensionalen Straßennetzwerken und Verkehrssimulationen [Ducloux et al., 2009].

OKSTRA

Das Bundesministerium für Verkehr, Bau- und Wohnungswesen führte 2000 den Objektkatalog für Straßen- und Verkehrswesen (OKSTRA) ein. In einem allgemeinen Rundschreiben von 2010 wird die Zielsetzung: "Zur einheitlichen Aufnahme, Bearbeitung und Vorhaltung der Merkmale von Straßen bedarf es für die weitere Bearbeitung eines Ordnungskriteriums und eines standardisierten Katalogs von Objekten des Straßen- und Verkehrswesens. Dies gewährleisten die Anweisung Straßeninformationsbank (ASB) mit der fachlichen und der OKSTRA mit der datentechnischen Beschreibung aufzunehmender Objekte und deren Strukturen der verschiedenen Fachbereiche des Straßen- und Verkehrswesens" definiert [BAST, 2016]. Das Ziel des OKSTRA ist demnach „die Gewährleistung einer einheitlichen Objektdarstellung und eines vereinheitlichten Datenaustausches von grafischen/geometrischen Daten im Straßen- und Verkehrswesen“ [Erstling & Portele 1996]. OKSTRA ist ein standardisiertes, konzeptionelles Datenmodell, das für verschiedenste Bereiche des Straßen- und Verkehrswesens eingesetzt wird [Hettwer, 2008]. Der IT-Koordinationsausschuss Bund-Länder im Straßenwesen (IT-KoA), in dem der Bund und die Bundesländer vertreten sind, finanziert die Pflege von OKSTRA und fungiert als Steuerungs- und Kontrollgremium bei grundlegenden Entscheidungen. Die Aufgaben der, bei der interactive instruments GmbH in Bonn ein-

gerichteten, OKSTRA-Pflegestelle liegen in den Bereichen Pflege und Weiterentwicklung. Die Weiterentwicklung des Datenmodells liegt im Wesentlichen bei den OKSTRA-Anwendern, die mittels eines Antragsverfahren neue Änderungs- oder Erweiterungswünsche einbringen können [Hettwer, 2008].

Geographic Data Files

Der Standard GDF wurde zum wirtschaftlichen und standardisierten Aufbau und Austausch von digitalen Karten, insbesondere Straßenkarten, entwickelt. Bereits Ende der 1980er Jahre wurden sich Hersteller und Nutzer digitaler Straßenkarten der Notwendigkeit einer standardisierten Datenaustauschmöglichkeit bewusst. Vor der Entwicklung intelligenter Transportsysteme (ITS) wurden Datenaustausch-Standards nur regional entwickelt, aber nicht an spezielle Straßentransportanforderungen angepasst. Die ISO 14825:2011-Norm wurde durch das Technische Komitee ISO/TC 204 „Intelligent Transport Systems“ in Zusammenarbeit mit dem Technischen Komitee CEN/TC 278 „Telematik für den Straßenverkehr und –transport“ erarbeitet [ISO 14825, 2011]. Im Zuge des Projekts Digital Electronic Mapping of European Territory (DETMETER) wurde 1988 der erste Standard (GDF 1.0) vorgestellt [May, 2002]. Nach zahlreichen Weiterentwicklungen wurde im Jahr 2011 der neueste Standard GDF 5.0 als Resultat intensiver Normungsaktivitäten und dem Bemühen einer Harmonisierung mit ISO/TC211 veröffentlicht [ISO 14825, 2011].

GDF ist ein konzeptuelles, logisches Datenmodell mit Definition eines Standard-Datenaustauschformats für vektorisierte Kartendaten und ist daher auch für diese Arbeit von Relevanz. Hauptanwendungsfeld dieses Austauschformats liegt im Bereich der Fahrzeugnavigation und Verkehrstelematik. Heutzutage wird dieser Standard auch in GIS, im Geomarketing, in der räumlichen Planung und Logistik, in Einsatzleitzentralen sowie zur Herstellung von digitalen und analogen Stadtplänen und Straßenkarten eingesetzt [VICKUS 1996].

3.2 OpenDRIVE

Das OpenDRIVE-Format wurde von der VIRES Simulationstechnologie GmbH in enger Zusammenarbeit mit Daimler Driving Simulator entwickelt. Das Format bietet eine XML-basierte, allgemeine Grundlage zur Beschreibung von spurbasierten Verkehrsnetzen. Die in einem OpenDRIVE-Dokument gespeicherten Daten beschreiben sowohl die parametrische Geometrie von Straßen als auch von Objekten entlang einer Straße, die die Logik des Straßenverlaufs beeinflussen, wie zum Beispiel Fahrspuren, Verkehrszeichen und Signale. Die repräsentierten Daten sind gegliedert in Knoten, welche mit benutzerdefinierten Daten erweitert werden können. Dadurch ist ein hoher Spezialisierungsgrad für individuelle Anwendungen unter Beibehaltung der Gemeinsamkeit, die für den Austausch von Daten zwischen verschiedenen Anwendungen erforderlich ist, möglich [Dupuis et al., 2015].

Der OpenDRIVE-Standard wird in der Automobilindustrie und Forschung zur Beschreibung von Straßen und Erzeugung virtueller Landschaften auf Grundlage realer Daten verwendet. Er wird in Fahr- und Verkehrssimulatoren und in Fahrassistenzsystemen, speziell in der Navigation, eingesetzt. Darüber hinaus findet er als Eingabeformat für Generierungswerkzeuge von Straßendatenbanken und

als Teil von weiterführenden, kommerziellen Produkten Anwendung [Dupuis et al., 2010].

Im Folgenden wird die aktuelle Version 1.4 der 2015 von VIRES Simulationstechnologie GmbH veröffentlichten OpenDRIVE Formatspezifikation vorgestellt.

3.2.1 Wichtige Koordinatensysteme

Zunächst wird ein Überblick über wichtige Koordinatensysteme, die Einsatz im vorgestellten Modellierungsstandards finden, gegeben. In OpenDRIVE werden in erster Linie zwei Koordinatensysteme verwendet. Diese werden in Abb. 3.1 dargestellt.

- Das **Inertial-Koordinatensystem** ist gemäß der ISO 8855 ein Rechtssystem. Die Achsen x , y und z beschreiben im geographischen Kontext die Freiheitsgrade Ost, Nord und Höhe. Die Rotation um die z -Achse wird *heading*, um die y -Achse *pitch* und um die x -Achse *roll* genannt.
- Das **Track-Koordinatensystem** liegt entlang der Referenzlinie der Straße und ist ebenfalls ein Rechtssystem. Es hat die Freiheitsgrade s , t und h . Die Position entlang der Referenzlinie, gemessen in Meter von Beginn der Strecke an, wird im xy -Plan berechnet und wird als Freiheitsgrad s angegeben. Der Freiheitsgrad t zeigt positiv nach links, h beschreibt die Höhe.

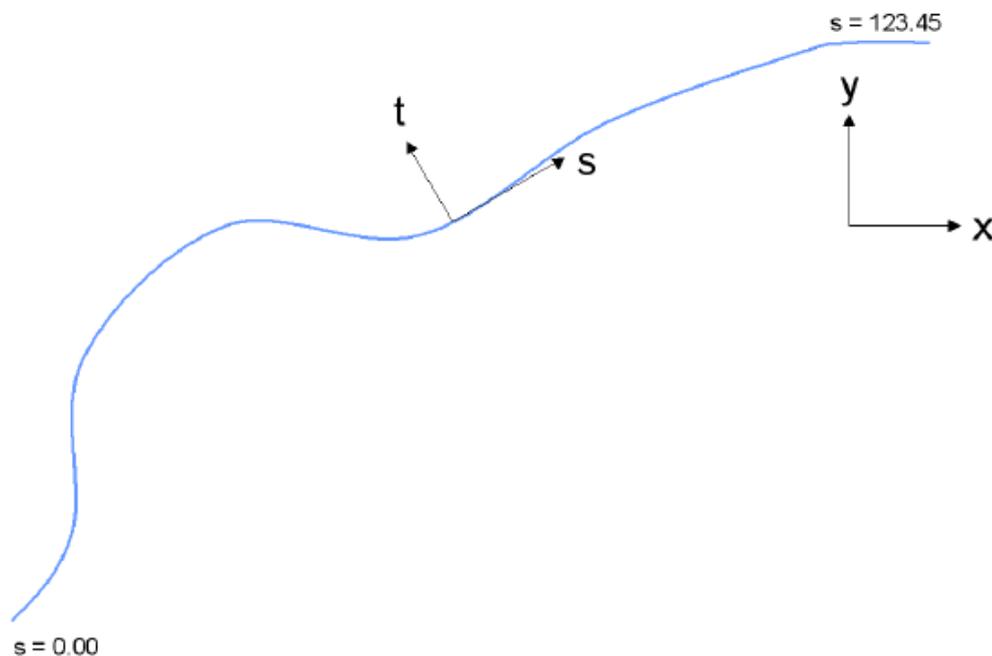


Abbildung 3.1 – Track- und Inertial-Koordinatensystem in OpenDRIVE [Dupuis et al., 2015, S.13]

Neben diesen beiden Koordinatensystemen wird ein ebenfalls rechtsdrehendes **Lokales Koordinatensystem** gemäß der ISO 8855 definiert. Die Achsen u : forward, v : left und z : up besitzen die mit dem Inertialsystem vergleichbaren Freiheitsgrade heading, pitch und roll. Das lokale System kann nur

durch die Bereitstellung der vollständigen Spurkoordinaten und der Orientierung seines Ursprungs im Spurraum positioniert werden.

3.2.2 Grundprinzip des Straßenverlaufs

Das Grundprinzip des Straßenverlaufes ergibt sich durch eine Referenzlinie, Fahrspuren und Objekte und wird in Abb. 3.2 anschaulich dargestellt.

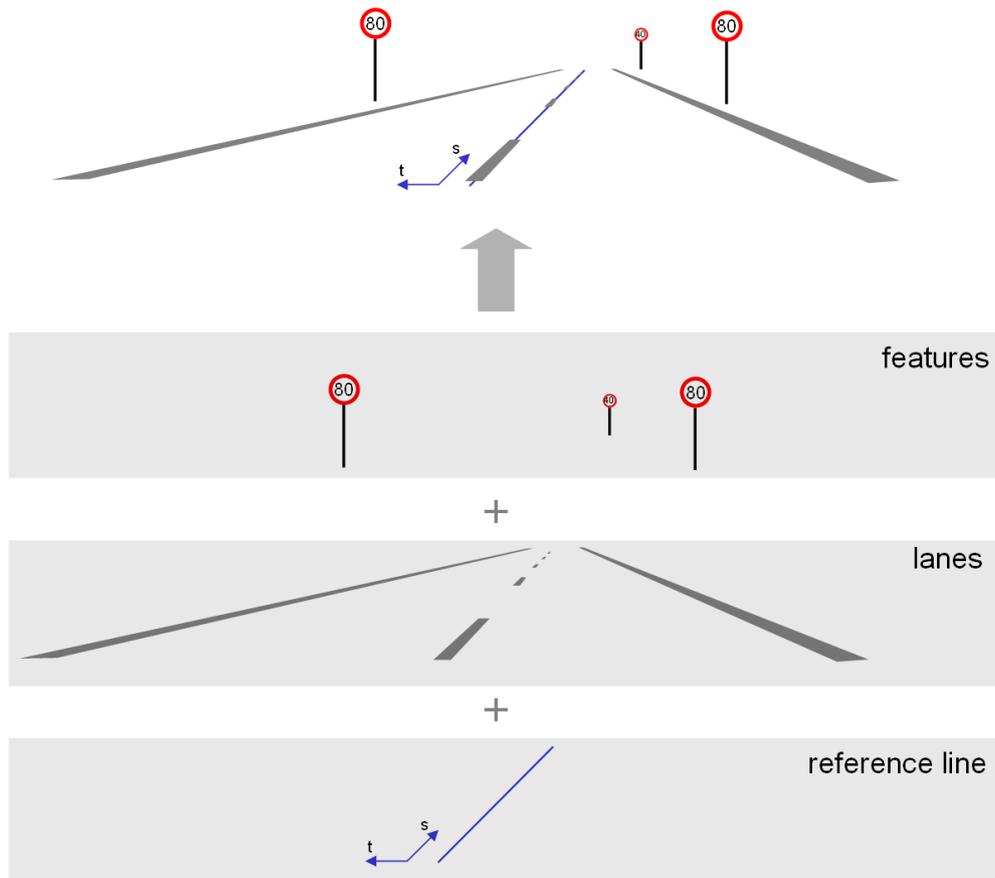


Abbildung 3.2 – Grundprinzip des Straßenverlaufs in OpenDRIVE [Dupuis et al., 2015, S.18]

Alle Straßen bestehen aus einer *Referenzlinie* durch die die grundlegende Geometrie wie Achsen und Linien festlegt wird. Die Geometrie der Referenzlinie wird als Sequenz von Primitiven unterschiedlicher Typen beschrieben. Zu den verfügbaren Primitiven gehören die Gerade mit einer konstanten Null-Krümmung, die Klothoide mit einem linearen Krümmungswechsel und die Kurve als konstante Nicht-Null-Krümmung entlang der Lauflänge. Zusätzlich können kubische Polynome und parametrische kubische Kurven verwendet werden.

Entlang dieser Referenzlinie können verschiedene Bestandteile der Straße wie Höhenprofile, Fahrspuren und Verkehrszeichen definiert werden. Alle diese Bestandteile können nach den in dieser Spezifikation festgelegten Richtlinien und gegebenenfalls durch benutzerdefinierten Daten parametrisiert werden.

Bestandteile des gleichen Typs, die entlang einer einzelnen Referenzlinie definiert sind, müssen in aufsteigender Reihenfolge aufgelistet werden.

3.2.3 Aufbau eines Straßenabschnitts

In den folgenden Kapiteln wird der Aufbau eines Straßenabschnitts in OpenDRIVE und die damit verbundene Modellierung der Fahrspuren, Querschnitte und möglichen Straßenverbindungen detaillierter beschrieben. Alle Elemente einer Straße werden mit einer Referenzlinie verknüpft und mittels linearer Referenzierung gespeichert.

3.2.3.1 Fahrspuren

Fahrspuren können durch folgende Eigenschaften charakterisiert werden:

- Pro Spurabschnitt treten Fahrspuren der Reihe nach ohne Lücken einzeln auf.
- Beginnend bei 0 auf der Referenzlinie werden die Spuren aufsteigend nach links, d.h. in positiver t -Richtung, und absteigend nach rechts, d.h. in negativer t -Richtung, nummeriert.
- Die Gesamtanzahl der Spuren ist nicht begrenzt.
- Die Referenzlinie selbst ist als Nullspur definiert und darf keinen Breitereintrag besitzen, d.h. sie erhält immer die Breite 0.0.

In bestimmten Fällen, z.B. wenn eine Linksabbiegespur als innerste Spur eingefügt wird, ist es sinnvoll das Spurprofil anstelle der Referenzlinie zu verschieben und somit einen *Spurenoffset* einzufügen. Für diesen Fall kann die Spur 0 mit einem kubischen Polynom versetzt werden (vergleichbar Abb. 3.3).

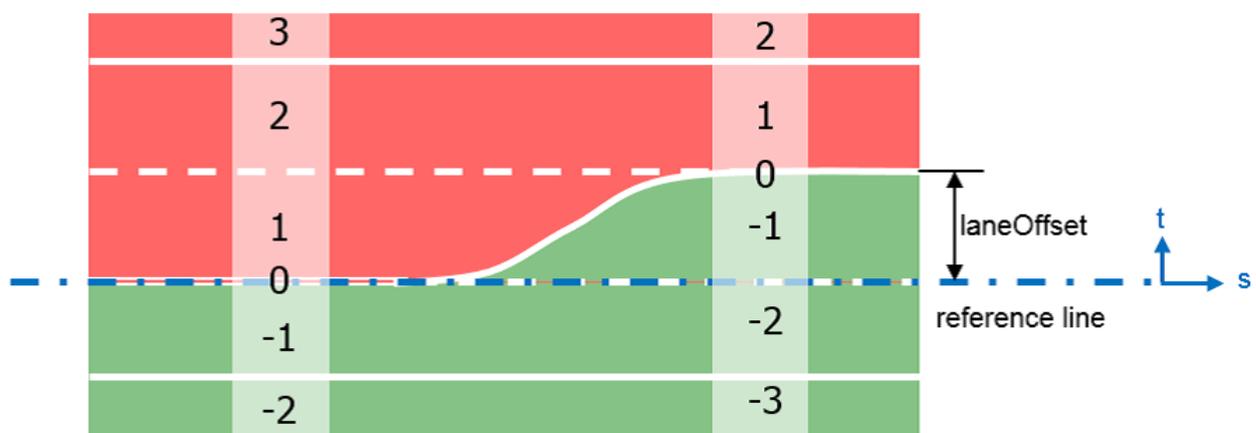


Abbildung 3.3 – Fahrspuren in OpenDRIVE [Dupuis et al., 2015, S.20]

Die Spuren, die in einem gegebenen Querprofil entlang einer Straße auftreten, sind als sogenannte *Straßenabschnitte* definiert. Mehrere Straßenabschnitte können in aufsteigender Reihenfolge entlang einer Referenzlinie definiert werden. Jeder Abschnitt ist gültig bis der nächste Abschnitt definiert wird. Folglich besteht jede Straße aus mindestens einem Straßenabschnitt beginnend bei $s=0.0$ m. Dieser Ansatz wird in Abb. 3.4 dargestellt:

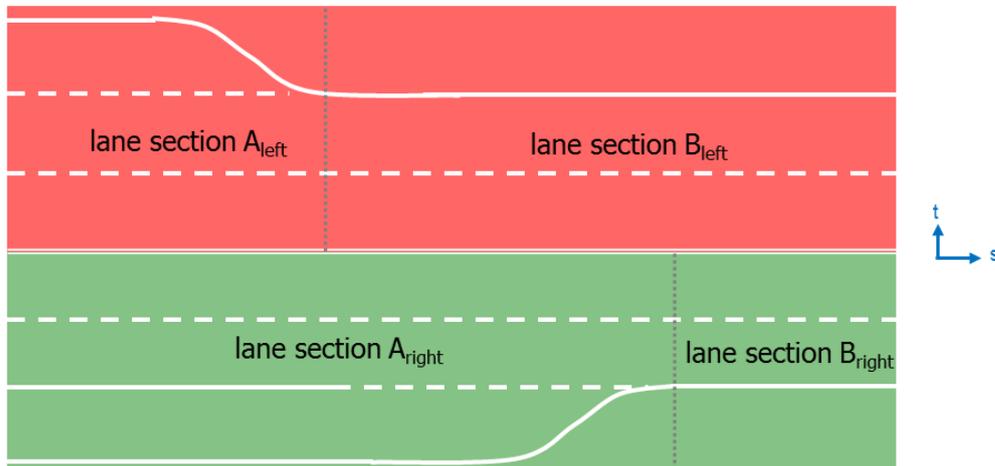


Abbildung 3.4 – Straßenabschnitte in OpenDRIVE [Dupuis et al., 2015, S.21]

Für jeden Straßenabschnitt ist die Anzahl der Spuren konstant. Bestandteile eines Abschnitts wie z.B. Straßenmarkierungen können sich trotzdem ändern. Beginnend mit der Version OpenDRIVE 1.3 müssen diese Abschnitte nur noch für eine Straßenseite gültig sein. Eine Änderung der Spuranzahl auf einer Seite der Straße ist dadurch ohne Einschränkungen und ohne erhöhtem Aufwand in der Definition des Straßenverlaufs möglich.

Spureigenschaften werden relativ zum Start des entsprechenden Spurprofils definiert. Offsets beginnen bei $s=0.0$ und nehmen entsprechend der Straßenkoordinate s zu. Die Spureigenschaften sind gültig bis eine neue Eigenschaft desselben Typs definiert wird oder der Abschnitt endet. Eigenschaften identischer Typen müssen in aufsteigender Reihenfolge definiert werden. Dies können Punkt- oder Bereichseigenschaften sein, d.h. sie sind lediglich durch einen gegebenen Punkt oder innerhalb eines gegebenen Bereichs gültig.

3.2.3.2 Überhöhung, Querneigung und Querprofil eines Straßenquerschnitts

In vielen Fällen liegt ein Straßenquerschnitt nicht parallel zum zugrundeliegenden Terrain. Stattdessen neigt sich der Querschnitt entweder zu einer Seite, was vor allem in Kurven auftritt, oder der Querschnitt wird für die Entwässerung zur Achsmitte erhöht. Diese Fälle können im OpenDRIVE-Format als Überhöhung (Superelevation) und Querneigung (Crossfall) beschrieben werden. Während die Überhöhung über den kompletten Straßenquerschnitt definiert wird, muss die Querneigung pro Straßenseite festgelegt werden. Einzelne Spuren können von der Verwendung der Überhöhungs- und

Querneigungseigenschaften ausgeschlossen werden. Fußgängerwege beispielsweise belaufen sich immer auf niveaugleicher Ebene (vergleichbar Abb. 3.5 Überhöhung links, Querneigung rechts).



Abbildung 3.5 – Querschnitte mit Überhöhung (links) und Querneigung (rechts) in OpenDRIVE [Dupuis et al., 2015, S.23]

Die Definition von Überhöhungswerten ändert nicht die tatsächliche Breite einer Fahrspur, sondern ihre projizierte Breite. Die Querneigung weist dagegen ein umgekehrtes Verhalten auf.

Manche Straßenquerschnitte sind für die bloße Beschreibung durch Überhöhung und Querneigung zu komplex. Eine Möglichkeit zur Beschreibung solcher komplexen Oberflächen ist die Verwendung von expliziten Oberflächendaten. Eine analytische Methode zur Beschreibung von Querprofilen wird der Formatspezifikation mit OpenDRIVE 1.4 hinzugefügt. Das erlaubt die Beschreibung von kurvigen Straßenoberflächen wie sie in Hochgeschwindigkeitsteststrecken vorkommen. Mit der Einführung von Querprofilen muss die Querneigung als Spezialfall behandelt werden. Die Kombination aus Querneigung und Querprofil ist nicht möglich. Die Überhöhung agiert auf der Referenzebene der Straße und nicht direkt auf dem Querprofil selbst (siehe Abb. 3.6).



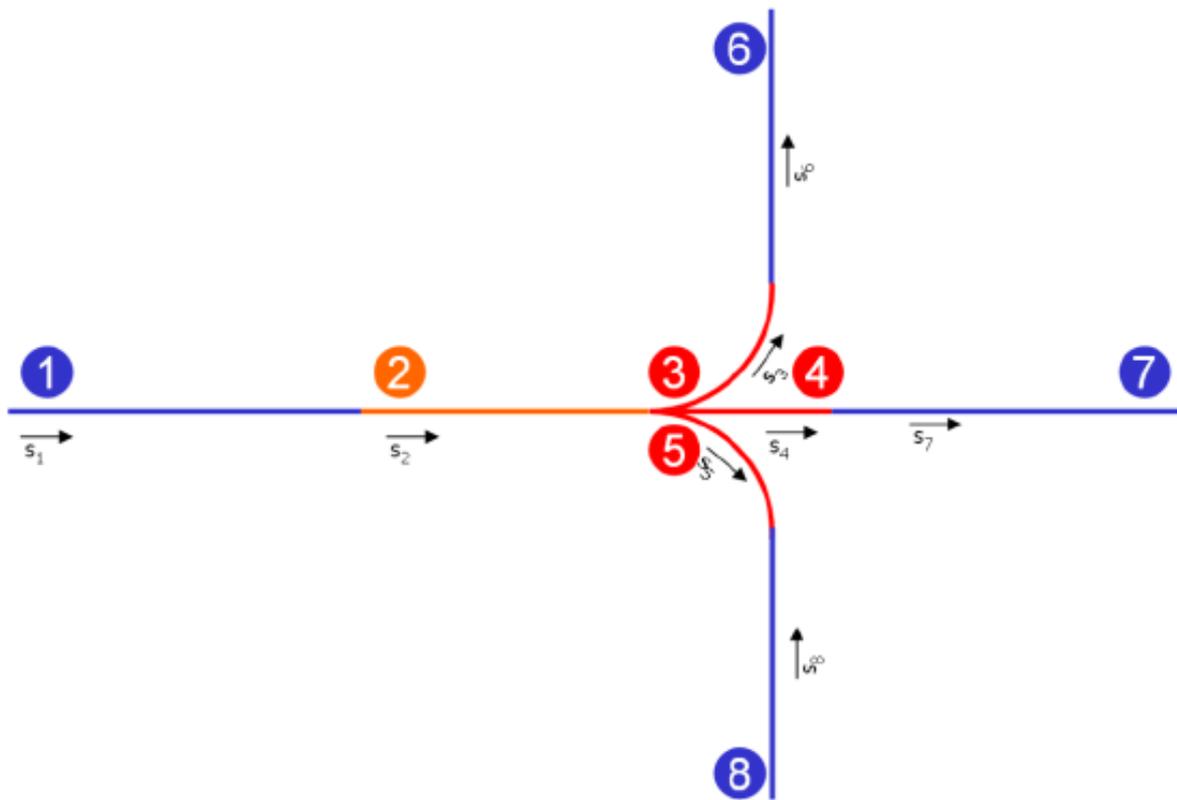
Abbildung 3.6 – Querprofile in OpenDRIVE [Dupuis et al., 2015, S.24]

3.2.3.3 Verbindungen zwischen einzelnen Straßenabschnitten

Zur Navigation durch ein Straßennetzwerk sind Kenntnisse über die Verbindung zwischen Straßenabschnitten notwendig. Im OpenDRIVE-Standard sind zwei Straßenverbindungstypen möglich: *Nachfolger-Vorgänger-Verbindung* und *Kreuzungen (Junctions)*. Außerdem können *Nachbarschaftsbeziehungen* zwischen Fahrspuren abgebildet werden.

Die Standardverbindung mit Informationen über Vorgänger- und Nachfolgerabschnitte ist ausreichend, wenn zwei gegebene Straßen direkt miteinander verbunden werden können, d.h. falls es nur eine mögliche Verbindung zwischen ihnen gibt. Eine Kreuzung zur Verknüpfung der Straßenabschnitte wird

benötigt, wenn die Nachfolger-Vorgänger-Beziehung mehrdeutig ist und es mehr als eine mögliche Verbindung eines Straßenabschnitts zu anderen Abschnitten gibt. Innerhalb einer Kreuzung sind mehrere Verbindungen möglich. Zur Verdeutlichung kann folgende Abbildung mit zugehöriger Tabelle herangezogen werden.



Road	Predecessor	Successor
1	-	2
2	1	ambiguous
3	2	6
4	2	7
5	2	8
6	3	-
7	4	-
8	-	5

Abbildung 3.7 – Straßenverbindungsmöglichkeiten in OpenDRIVE [Dupuis et al., 2015, S.25]

Die Straßenabschnitte 3, 4 und 5 werden „connecting roads“ genannt. Für diese Abschnitte agiert Straße 2 als sogenannte „incoming road“. Die Straßen 6, 7 und 8 sind „outgoing roads“, welche auch als „incoming roads“ für andere Verbindungen, die nicht in dieser Abbildung dargestellt sind, agieren können. Auf Fahrplanniveau können zusätzliche Verbindungsinformationen zur besseren Navigation durch das Straßennetzwerk bereitgestellt werden.

Das Grundprinzip der Kreuzungen besagt demnach, dass Kreuzungen „incoming roads“ durch „connecting roads“ zu „outgoing roads“ verbinden. Auch die „Connecting Roads“ werden als Straßenabschnitte gemäß den hier beschriebenen Richtlinien bestehend aus Referenzlinien und Spuren modelliert.

3.3 City Geography Markup Language

CityGML ist ein offenes Datenmodell und XML-basiertes Format für die Repräsentation, Speicherung und den Austausch von virtuellen 3D-Stadtmodellen. Sie ist ein Anwendungsschema für die GML, welche ihrerseits den erweiterbaren internationalen Standard für räumliche Austauschdaten darstellt und vom OGC und der ISO TC211 definiert wird. Das Ziel der Entwicklung von CityGML ist die Schaffung einer allgemeinen Definition für die wesentlichen Instanzen, Attribute und Beziehungen eines 3D-Stadtmodells. Dies ist hinsichtlich der kostengünstigen, nachhaltigen Verwaltung von 3D-Stadtmodellen und der damit verbundenen Wiederverwendung der gleichen Daten in verschiedenen Anwendungsfeldern besonders wichtig. CityGML differenziert fünf konsekutive Detaillierungsgrade, sogenannte LoD. Mit steigendem LoD werden die Objekte sowohl in Hinblick auf ihre Geometrie, aber auch bezüglich ihrer thematischen Differenzierung detailreicher abgebildet. Modelle im CityGML-Format können gleichzeitig mehrere Repräsentationen (und Geometrien) für jedes Objekt in verschiedenen LoD beinhalten. Generalisierungsbeziehungen erlauben die eindeutige Repräsentation von Objekten über verschiedene Skalen [Gröger et al., 2012].

Der umfassende Journal-Artikel von Biljecki et al. [2015] der TU Delft mit dem Titel „Applications of 3D City Models: State of the Art Review“ stellt eine umfangreiche Auflistung von möglichen Anwendungsfeldern für 3D-Stadtmodelle bereit. Basierend auf einer umfangreichen Literaturstudie mit zahlreichen wissenschaftlichen Artikeln, technischen Reports und Onlinequellen werden insgesamt 29 Anwendungsfelder mit Beispielen vorgestellt [Biljecki et al., 2015]. Typische Anwendungen speziell von CityGML sind unter anderem die Stadt- und Landschaftsplanung, 3D-Kataster, die Fahrrad- und Fußgängeravigation sowie Trainingssimulatoren. Auch Umweltsimulationen und das Katastrophenmanagement können durchaus vom Einsatz CityGML-basierter 3D-Stadtmodelle profitieren [Gröger et al., 2012].

Im Folgenden werden alle für diese Arbeit besonders relevanten Teile des 2012 veröffentlichten Encoding Standards für CityGML 2.0 vorgestellt. Der CityGML Standard wird zur übersichtlicheren Darstellung einzelner Klassen und deren Zusammenhänge (Assoziation, Aggregation und Vererbung) untereinander in der Klassendiagrammdarstellung der objektorientierten Datenmodellierungssprache Unified Modeling Language (UML) dargestellt. Auf eine ausführliche Erläuterung des UML-Konzepts wird an dieser Stelle verzichtet, jedoch auf Booch et al. [1991] verwiesen. Um die Lesbarkeit von CityGML UML-Diagrammen zu erhöhen, werden Klassen, die zu unterschiedlichen UML-Paketen gehören, in verschiedenen Farben dargestellt: Gelb dargestellte Klassen gehören zum UML-Paket, welches gerade im jeweiligen Abschnitt behandelt wird. Blau dargestellte Klassen gehören zu den UML-Paketen, die

sich von den gelb ge-kennzeichneten Paketen unterscheiden. Grün dargestellte Klassen sind bereits in GML3 definiert und ihrem Klassennamen wird der Prefix *gml* vorangestellt.

3.3.1 Kernmodell: CityGML Core

Das CityGML-Datenmodell ist thematisch in ein Kernmodell (Core module) und in mehrere Erweiterungsmodulen (Extension modules) unterteilt. Das Kernmodell, welches in Abb. 3.8 als UML-Diagramm dargestellt wird, umfasst die grundlegenden Konzepte und zentralen Komponenten des CityGML-Datenmodells. Es stellt hauptsächlich abstrakte Grundklassen bereit, aus denen weitere Klassen innerhalb der Erweiterungsmodulen abgeleitet werden können. Die Basisklasse aller thematischen Klassen innerhalb des CityGML-Datenmodells ist die abstrakte Klasse *CityObject*, welche eine Unterklasse der GML-Klasse *Feature* darstellt. *CityObject* beinhaltet Attribute zur Erzeugung und Zerstörung von Objekten, sowie zur Festlegung der genauen Lage eines Objekts relativ zur Gelände- oder Wasseroberfläche [Gröger et al., 2012].

Basierend auf dem Kernmodell deckt jede Erweiterung einen spezifischen thematischen Bereich virtueller 3D-Stadtmodellen ab. CityGML differenziert die folgenden dreizehn thematischen Erweiterungsmodulen: Appearance, Bridge, Building, CityFurniture, CityObjectGroup, Generics, LandUse, Relief, Transportation, Tunnel, Vegetation, WaterBody und TexturedSurface [Gröger et al., 2012].

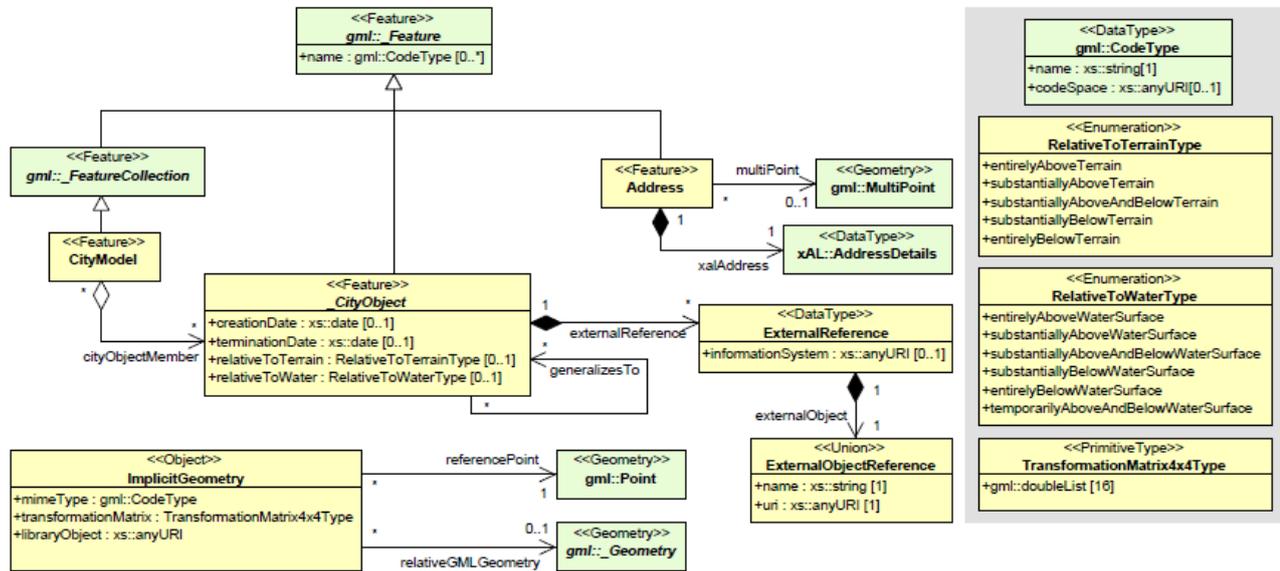


Abbildung 3.8 – UML-Diagramm des CityGML-Kernmodell [Gröger et al., 2012, S.50]

CityGML differenziert fünf konsekutive Detaillierungsgrade, sogenannte LoD, welche in Abb. 3.9 graphisch dargestellt werden. Mit steigendem LoD werden die Objekte sowohl in Hinblick auf ihre Geometrie, aber auch bezüglich ihrer thematischen Differenzierung detailreicher abgebildet [Gröger et al., 2012]. Auch im Bereich des Straßenraums werden fünf Detaillierungsgrade, die in Abschnitt 3.3.2.1

vorgestellt werden, differenziert.

Die niedrigste Detaillierungsstufe LoD 0 beinhaltet ein 2,5-dimensionales digitales Geländemodell auf das ein Luftbild oder eine Karte gelegt werden kann. Gebäude können im LoD 0 als Grundriss oder Dachkantenpolygone dargestellt werden. Die nächst höhere Stufe LoD 1, auch als Klötzchenmodell bekannt, besteht aus quaderförmigen Gebäuden mit flachen Dachstrukturen. Im Gegensatz dazu hat ein Gebäude im LoD 2 thematisch differenzierbare Dachstrukturen und Begrenzungsflächen. Im LoD 3 werden Architekturmodelle mit detaillierten Wand- und Dachstrukturen und eventuell enthaltenen Türen und Fenstern dargestellt. Die höchste Stufe LoD 4 vervollständigt das LoD 3-Modell durch Ergänzung innerer Strukturen wie Räume, Innentüren, Treppen und Mobiliar. In allen Detaillierungsstufen können zusätzliche Darstellungsinformationen wie zum Beispiel hochaufgelöste Texturen auf die Strukturen abgebildet werden [Gröger et al., 2012].

Modelle im CityGML-Format können gleichzeitig mehrere Repräsentationen (und Geometrien) für jedes Objekt in verschiedenen LoD beinhalten. Generalisierungsbeziehungen erlauben die eindeutige Repräsentation von Objekten über verschiedene Skalen [Gröger et al., 2012].

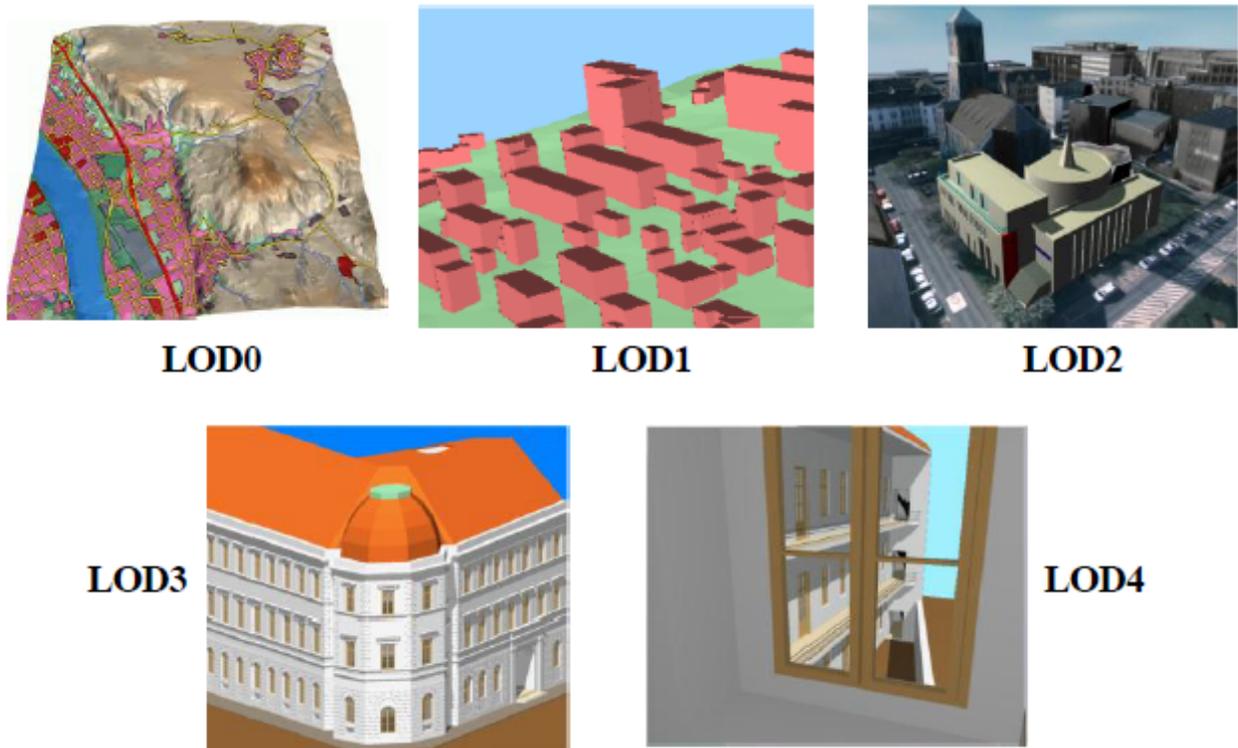


Abbildung 3.9 – Die fünf LoD in CityGML (Graphik: IGG Uni Bonn) [Gröger et al., 2012, S.11]

3.3.2 Relevante Erweiterungsmodelle

3.3.2.1 Transportmodell: Transportation Object

Das Transportmodell von CityGML ist ein multifunktionales, mehrfachskalierbares Modell, das sich sowohl auf thematische und funktionelle, als auch auf geometrisch topologische Aspekte fokussiert.

Die *Transportation features* werden im LoD 0 als linienhaftes Netzwerk repräsentiert. Startend bei LoD 1 werden alle *Transportation features* geometrisch als dreidimensionale Oberflächen beschrieben. Die flächenhafte Modellierung der *Transportation features* erlaubt die Anwendung von geometrischen Straßenplanungsalgorithmen. Dies kann zur Festlegung von benötigten Begrenzungen entlang eines Verkehrsweges nützlich sein. Diese Information kann auch für die Trajektorienplanung mobiler Roboter in der realen Welt oder für die automatische Platzierung virtueller Personen und Fahrzeugmodellen in 3D-Visualisierungen und Trainingssimulatoren verwendet werden. Das Transportmodell von CityGML wird durch das thematische Erweiterungsmodul *Transportation* bereitgestellt.

Die Hauptklasse *TransportationComplex* ist eine Unterklasse der Klasse *_TransportationObject*, welche wiederum eine Wurzelklasse von *_CityObject* darstellt. *TransportationComplex* kann in die vier thematisch unterschiedlichen Klassen *Road*, *Track*, *Railway* und *Square* unterteilt werden (siehe Abb. 3.10). Die im ersten Bild abgebildete Klasse *Road* beinhaltet sämtliche Objekte, die einer Straße zugeordnet werden können. Dazu gehören neben der Verkehrsfläche auch angrenzende Gehwege, Randsteine oder Fahrbahnmarkierungen. Im Gegensatz dazu beschreibt die Klasse *Track*, dargestellt im zweiten Bild, in erster Linie Pfade und Gehwege in Parks. Mithilfe der Klasse *Railway* werden Eisenbahnschienen repräsentiert. Öffentliche Plätze und großflächig versiegelte Bereiche werden durch die im vierten Bild dargestellten Klasse *Square* beschrieben. Der Standard liefert keine exakte Definition, ob auch Kreuzungsbereiche zu dieser Klasse zuzuordnen sind.

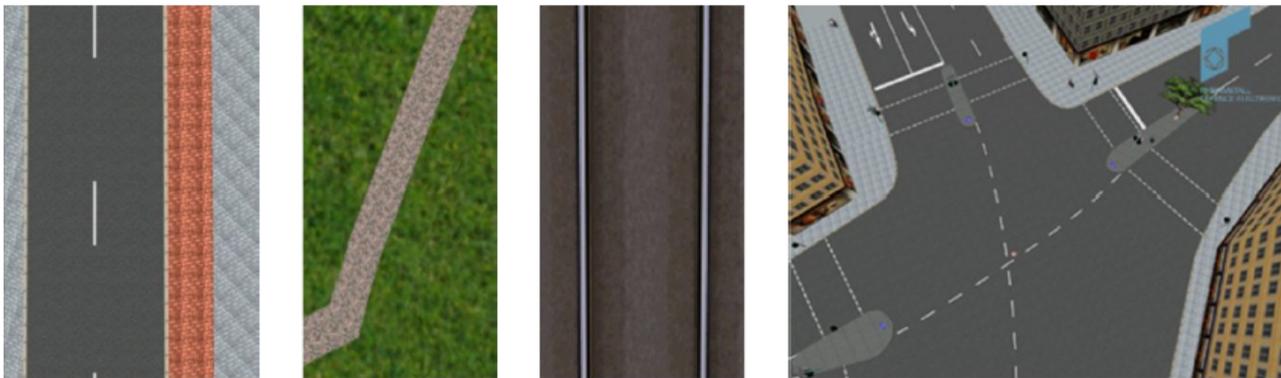


Abbildung 3.10 – Repräsentationen der Klasse *TransportationComplex* - Von links nach rechts: *Road*, *Track*, *Rail*, *Square* (Graphik: Rheinmetall Defence Electronics) [Gröger et al., 2012, S.124]

Die Klasse *TransportationComplex* lässt sich weiter in *TrafficAreas* und *AuxiliaryTrafficAreas* unterteilen. *TrafficAreas* sind jene Elemente, die in Bezug auf die Straßennutzung wichtig sind wie Fahrspuren, Fußgängerzonen und Fahrradwege. *AuxiliaryTrafficAreas* beschreiben Elemente der Straße wie Bordsteine, Mittelstreifen oder Grünflächen. Jeder *TransportationComplex* besitzt die Attribute *class*, *function* und *usage*, deren mögliche Werte in Codelisten spezifiziert werden können. Das Attribut *class* beschreibt die Klassifikation des Objekts, *function* beschreibt die vorgesehene Funktion und *usage* die tatsächliche Nutzung wie Autobahn, Landstraße oder Flughafen. Dieses Attribut wird verwendet falls die Nutzung von der ursprünglich angedachten Funktion des Objekts abweicht. Die Attribute *function* und *usage* können mehrfach vorkommen. *TrafficAreas* und *AuxiliaryTrafficAreas* können mit dem Attribut *surfaceMaterial* Informationen über die Oberflächenbeschaffenheit enthalten

[Gröger et al., 2012]. Alle vier Attribute werden als *gml:CodeType* spezifiziert, wodurch für diese Attribute eine Liste mit umfangreichen Definitionen von Codes für unterschiedliche Objektarten zur Verfügung steht. Das gesamte UML-Diagramm des CityGML Transportmodell wird in Abb. 3.11 dargestellt.

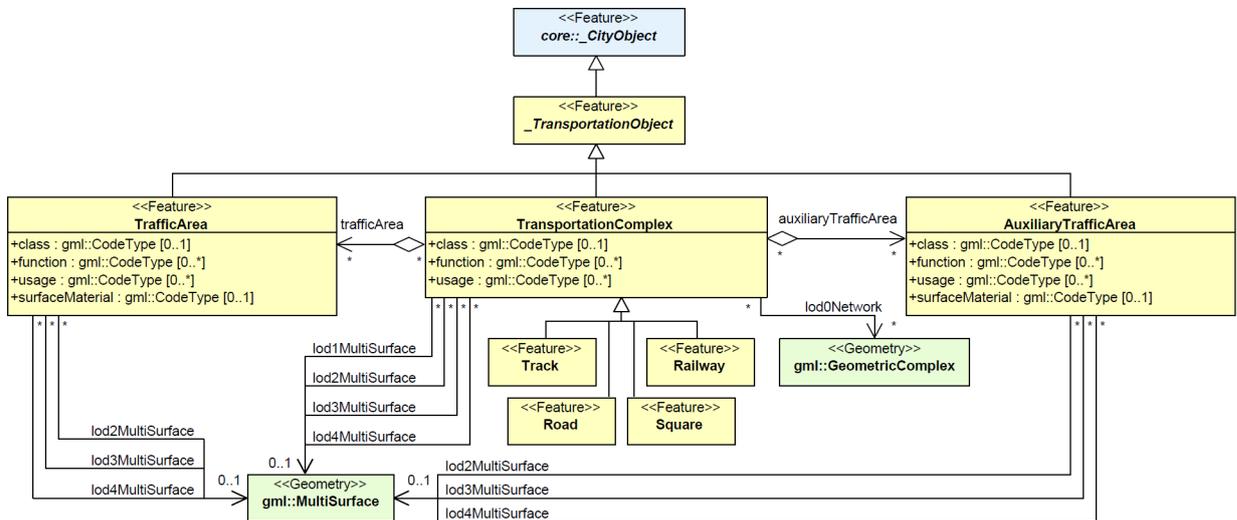


Abbildung 3.11 – UML-Diagramm des CityGML-Transportmodell [Gröger et al., 2012, S.123]

Die geometrische Repräsentation der Objekte des *TransportationComplex* variiert durch die genannten verschiedenen Detaillierungsgrade. CityGML differenziert auch hier fünf konsekutive Detaillierungsgrade. Auf der niedrigsten Detaillierungsstufe LoD 0 werden sie durch Linienobjekte in einem linienhaften Netzwerk modelliert. Bereits im LoD 1 sieht ein *TransportationComplex* nicht nur die Mittellinie, sondern auch eine explizite Oberflächengeometrie, die die aktuelle Form des Objektes repräsentiert, vor. In den LoD 2-4 wird der *TransportationComplex* thematisch weiter in *TrafficAreas* und *AuxiliaryTrafficArea* unterteilt (siehe Abb. 3.12).

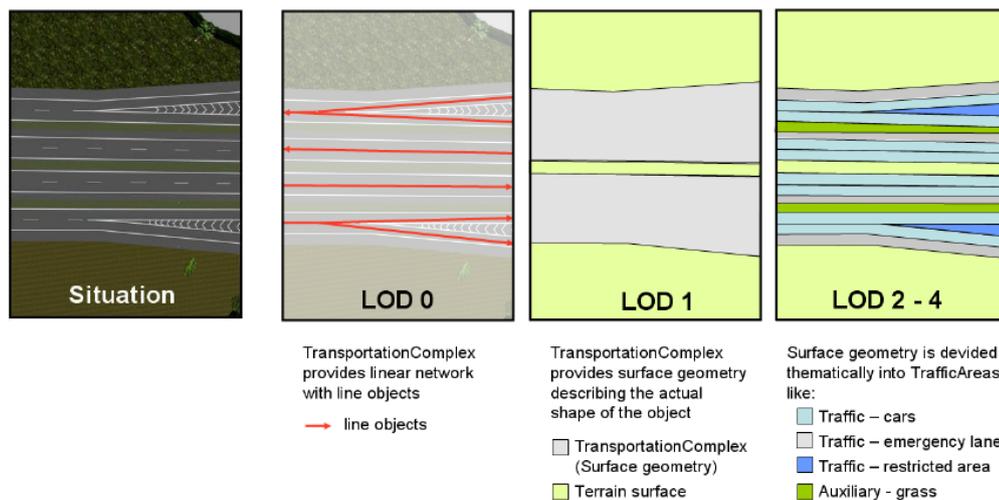


Abbildung 3.12 – Unterschiedliche LoD des TransportationComplex (Graphik: Rheinmetall Defence Electronics) [Gröger et al., 2012, S.126]

3.3.2.2 Erweiterungsmöglichkeiten des CityGML-Transportmodells nach Beil

CityGML liefert mit dem Erweiterungsmodul *Transportation* in erster Linie eine Grundlage für eine einheitliche Modellierung des Straßenraums. In einigen Bereichen ist die Definition bewusst vage definiert um Spielraum für anwenderspezifische Anforderungen zu lassen. Mittels generischer Objekte und Attribute sind Anpassungen möglich. Trotzdem bestehen mehrere Defizite und Unklarheiten, die den Standard an einigen Stellen uneindeutig werden lassen. Aufgrund dessen hat Beil [2017] Weiterentwicklungsmöglichkeiten des aktuellen Transportmodells aus dem CityGML2.0-Standard erarbeitet. Diese beinhalten einen Vorschlag zur Neueinteilung der LoD und einige Veränderungsmöglichkeiten des Modells. Zur Überarbeitung des LoD-Konzepts wurde der Vorschlag einer neuen Einteilung in lediglich vier LoD-Stufen (LoD 0-3) von Löwner et al. [2013] und Löwner et al. [2015] übernommen. Das bisherige System könnte in den Stufen LoD 0 und LoD 1 weitestgehend bestehen bleiben, lediglich die LoD 2-4 müssten bei der Neuteilung auf zwei Stufen reduziert werden.

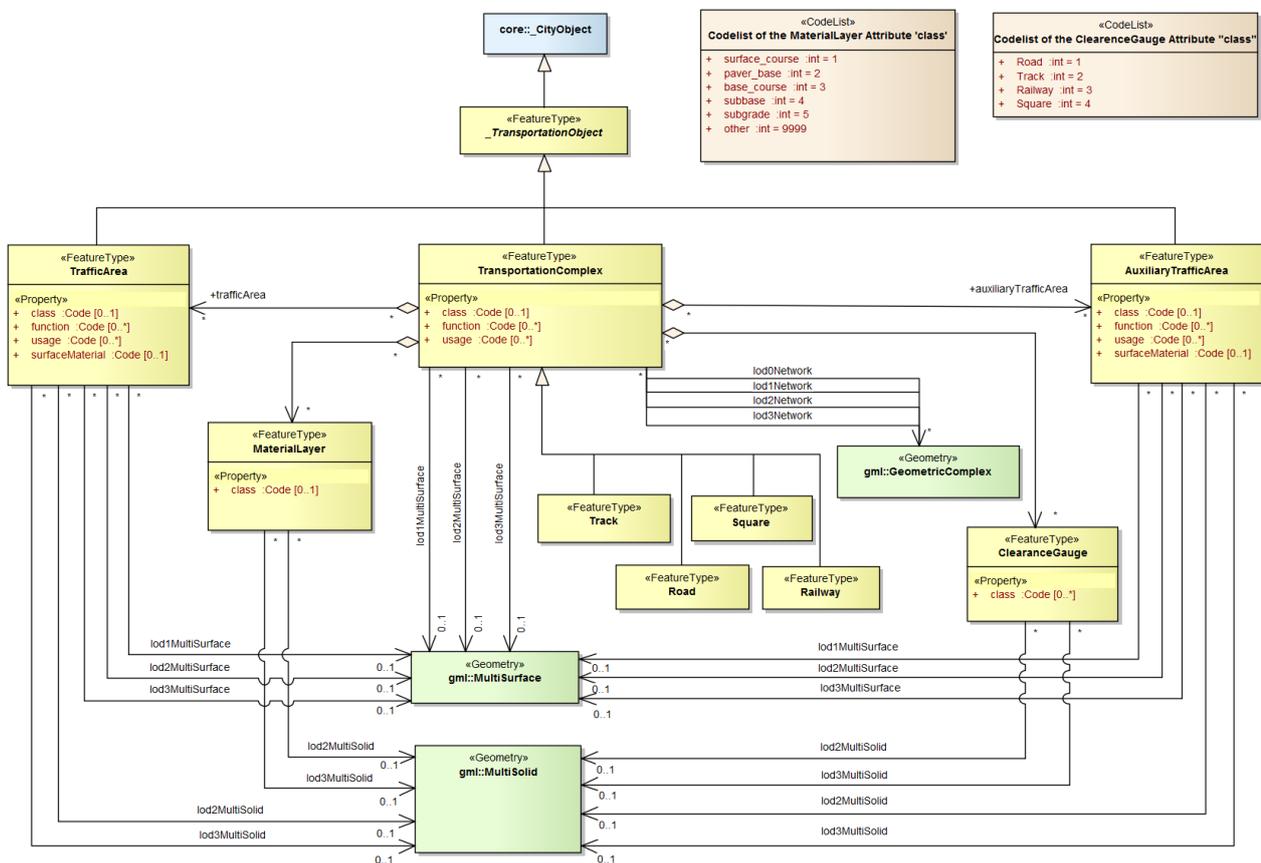


Abbildung 3.13 – UML-Diagramm eines weiterentwickelten CityGML-Transportmodells [Beil, 2017 S. 52]

Neben der Neueinteilung werden auch neue Klassen und Geometrien vorgeschlagen. Die vorgeschlagenen Erweiterungsmöglichkeiten des CityGML-Transportmodells können aus dem erweiterten UML-Diagramm in Abbildung Abb. 3.13 entnommen werden. Die Grundeinteilung in die Klassen *TransportationComplex*, *TrafficArea* und *AuxiliaryTrafficArea* bleibt mit dem Unterschied, dass die beiden letztge-

nannten Klassen bereits ab LoD 1 beschränkt zulässig sind, bestehen. Für alle Detaillierungsstufen wird eine linienhafte Darstellung des Straße durch ihr Bestandsachse möglich. Zusätzlich wird die Einführung der Geometrie *gml:MultiSolid* vorgeschlagen. Dadurch ist die Darstellung volumenhafter Straßenraumobjekte, wie zum Beispiel erhöhte Gehwege, Randsteine oder Verkehrsinseln möglich. Bisher musste diese Objektart durch mindestens drei begrenzende Flächen modelliert werden. Abbildung Abb. 3.14 zeigt die mögliche Repräsentation eines Straßenstücks in den vorgeschlagenen vier LoD-Stufen für mögliche linien- und flächenhafte Modelle. In der niedrigsten Detaillierungsstufe LoD 0 wird die Straße durch ihre Bestandsachse dargestellt. Eine sowohl linien- als auch flächenhafte Modellierung ist ab nächsthöherer Stufe LoD 1 möglich. Das orange eingefärbte Flächenstück enthält die gesamte Verkehrsfläche mit Fahrbahnrand und Gehwegen. Gehwege könnten je nach Anwendung auch ignoriert oder als eigene Fläche dargestellt werden. Dadurch wäre eine eingeschränkte Verwendung von *TrafficAreas* bereits ab LoD 1 zulässig. Eine thematische Unterteilung der Verkehrsfläche in *TrafficAreas* (Fahrbahnen, Gehwege) und *AuxiliaryTrafficAreas* (Randsteine, Markierungen) wird in LoD 2 umgesetzt. Die Fahrbahn bleibt durch ihre Bestandsachse repräsentiert, die Gehwege können durch eigene Achsen dargestellt werden. In LoD 3 können geometrische Feinheiten wie Gullydeckel und Straßenschäden durch eigene Flächen repräsentiert werden. Außerdem besitzt jede Fahrspur eine eigene linienhafte Repräsentation. Die Einführung der Klassen *MaterialLayer* und *ClearanceGauge* ermöglichen eine Repräsentation von Materialschichten und Lichtraumprofilen als Volumenobjekte ab LoD 2. Eine Klasse für die Segmentierung des Straßenraums in Abschnitte (z.B. von Kreuzungsbereich zu Kreuzungsbereich) wird auch diskutiert. Jeder Abschnitt könnte als eigenes *Road*-Objekt mit eindeutiger ID definiert werden. Über eine gemeinsame *parent.id* könnten alle Teilflächen miteinander verknüpft werden und umkehrt jeder Abschnitt durch XLinks auf alle Teilflächen verlinkt werden. Die Zuordnung von Gehwegen zu Straßensegmenten, Kreuzungsbereichen und Kreisverkehren müsste an dieser Stelle noch diskutiert werden.

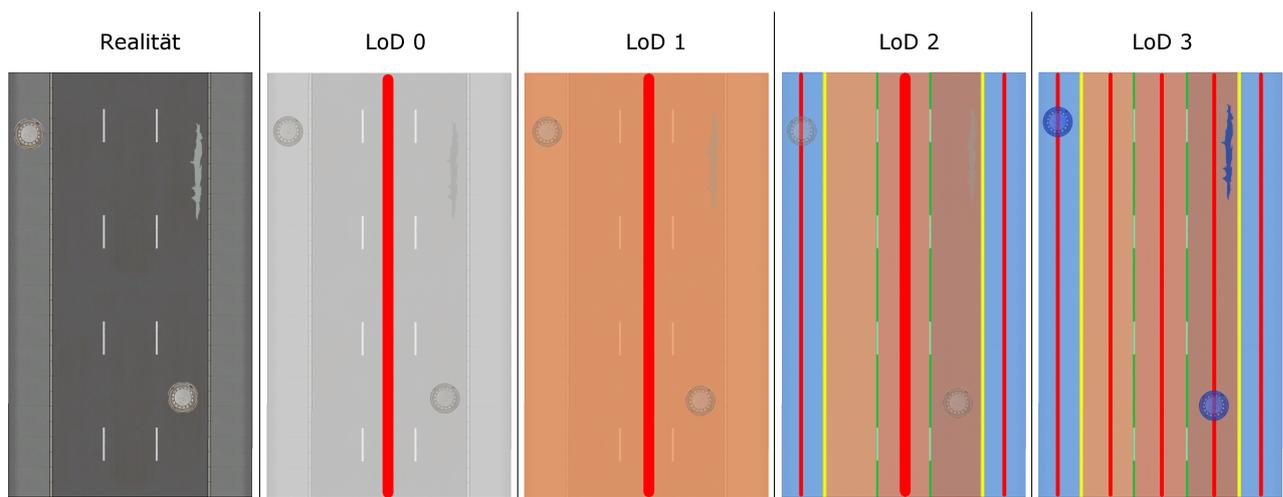


Abbildung 3.14 – Repräsentation eines Straßenstücks nach vorgeschlagenem LoD-Konzept [Beil, 2017 S. 53]

3.3.2.3 Straßenmöblierungsmodell: City furniture

Straßenmöblierungsobjekte sind unbewegliche Objekte wie Laternen, Ampeln, Verkehrszeichen, Litfaßsäulen, Bänke, Grenzpfosten oder Bushaltestellen und sind Bestandteil von Verkehrsflächen, Wohngebieten, Plätzen und Ortschaften. Die Modellierung von Straßenmöblierungsobjekten wird zum Beispiel für die Visualisierung des Stadtverkehrs, aber auch für die Analyse örtlicher Gegebenheiten genutzt. Die Erkennung spezieller Orte in einem Stadtmodell wird durch die Verwendung detaillierter Straßenmöblierungsobjekte verbessert, das Stadtmodell wirkt zusätzlich lebendiger und animierter. Das Straßenmöblierungsmodell von CityGML ist definiert durch das thematische Erweiterungsmodul *CityFurniture* [Gröger et al., 2012]. Das zugehörige UML-Diagramm wird in Abb. 3.15 dargestellt.

Die Klasse *CityFurniture* kann die Attribute *class*, *function* und *usage* beinhalten, deren mögliche Werte in entsprechenden Codelisten spezifiziert werden können. Das Attribut *function* erlaubt eine Objektklassifikation wie Ampel, Verkehrszeichen, Grenzpfosten oder Mülleimer, und kann nur einmalig vorkommen. Das Attribut *class* beschreibt zu welcher thematischen Fläche das Straßenmöblierungsobjekt gehört (z.B. Verkehr, Verkehrsregeln, Architektur) und kann mehrmals auftreten. Das Attribut *usage* beschreibt die wirkliche Verwendung des Objekts [Gröger et al., 2012].

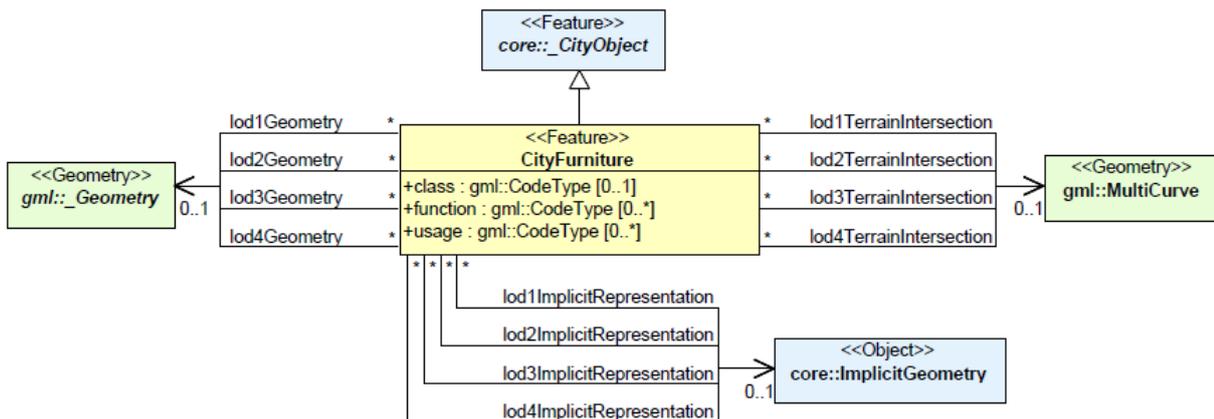


Abbildung 3.15 – UML-Diagramm der CityFurniture in CityGML [Gröger et al., 2012, S.138]

Über *ExternalReferences* können Stadtmöblierungsobjekte Links zu externen, thematischen Datenbanken enthalten. Dadurch können semantische Informationen von Objekten, die nicht in CityGML modelliert werden können, weitergeleitet und in 3D-Stadtmodellen für weitere Verarbeitungsschritte verwendet werden. *CityFurniture*-Objekte können durch explizite Geometrien (*lodXGeometry*, mit $X = 1, \dots, 4$) oder implizite Geometrien (*lodXImplicitRepresentation*, mit $X = 1, \dots, 4$) beschrieben werden. Implizite Geometrien beschreiben die Geometrie eines Objekts lediglich in einem lokalen Koordinatensystem, wodurch der Speicheraufwand reduziert werden kann [Gröger et al., 2012].

3.3.3 XLinks in CityGML

CityGML bietet die Möglichkeit zur expliziten Modellierung der Topologie, wodurch eine gemeinsame Nutzung eines Geometrieobjekts durch mehrere Objekte oder andere Geometrien möglich wird. Das räumliche Objekt wird dabei nur einmalig explizit durch ein Geometrieobjekt repräsentiert. Von allen weiteren Objekten oder Geometrien, zu denen es gehört, wird es lediglich referenziert. Auf diese Weise werden redundante Geometrien vermieden und gleichzeitig explizite topologische Beziehungen unterstützt. Grundsätzlich kann man zwischen drei Fällen unterscheiden:

- Zwei Objekte werden räumlich durch dieselbe Geometrie definiert, z.B.
- Eine Geometrie (z.B. eine Wand), wird gleichzeitig durch eine andere Geometrie (z.B. die Geometrie des Gebäudes), und ein Objekt (z.B. eine weitere Wand) verwendet.
- Zwei Geometrien werden (teilweise) durch dieselbe Geometrie repräsentiert. Eine Wandfläche zwischen Gebäude und Garage wird nur einmal repräsentiert und durch beide Objekte referenziert.

In CityGML wird für die Implementierung derartiger topologischer Beziehungen das von GML bereitgestellte XML-Konzept der XLinks verwendet. Jedem geometrischen Objekt, welches Teil mehrerer verschiedener Aggregate oder thematischer Objekte ist, wird eine eindeutige ID zugeteilt. Diese kann von GML-Geometrien durch die Verwendung des *href*-Attributs referenziert werden. Die Verwendung der XLink-Topologie hat im Gegensatz zu dem eingebauten Topologiepaket von GML3 den Vorteil einfach und flexibel zu sein und ist fast genauso leistungsstark wie das explizite GML3-Topologiemodell. Die Lösung von GML3 stellt zwar separate Topologieobjekte mit beigefügter Geometrie bereit, ist aber sehr komplex und aufwendig. Einen Nachteil der XLink-Topologie stellt die Navigation zwischen topologisch verbundenen Objekten dar. Diese ist nämlich nur in eine Richtung und nicht bidirektional, wie es der Fall bei GMLs eingebauter Topologie ist, möglich [Gröger et al., 2012]. In Abb. 3.16 wird das Konzept anhand eines Beispielgebäudes mit direkt anschließender Garage verdeutlicht.

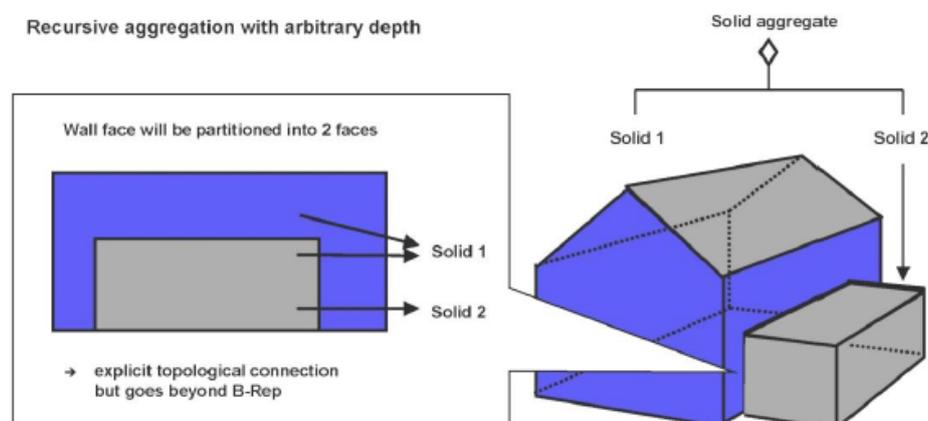


Abbildung 3.16 – Rekursive Aggregation von Objekten und Geometrien in CityGML (Graphik: IGG Uni Bonn) [Gröger et al., 2012, S.27]

3.3.4 CityGML-Erweiterungen

Grundsätzlich wurde CityGML als ein anwendungsunabhängiges Informationsmodell und Austauschformat für 3D-Stadt- und Landschaftsmodelle entwickelt. Spezielle Anwendungsfälle benötigen allerdings zusätzliche Informationen, die über das in CityGML definierte Grundlagenmodell hinausgehen. Diese müssen für die Modellierung und den Austausch aufbereitet werden. Die Kombination von Stadtmodellldaten und Anwendungsdaten ist generell mit zwei verschiedenen Ansätze möglich:

1. Generische Objekte und Attribute

Der Ansatz der generischen Objekte und Attribute erlaubt die Speicherung und den Austausch von 3D-Objekten, die nicht durch eine in CityGML eindeutig modellierte thematische Klasse erfasst oder deren erforderlichen Attribute nicht in CityGML repräsentiert werden können. Diese generische Erweiterung des CityGML-Datenmodells wird durch die Klassen *GenericCityObject* und *genericAttribute* realisiert. Beide Klassen sind innerhalb des thematischen Erweiterungsmoduls *Generics* definiert. Probleme bezüglich der semantischen Interoperabilität können vermieden werden, indem die generischen Erweiterungen nur verwendet werden, wenn keine geeigneten Klassen oder Attribute durch ein anderes CityGML-Modul zur Verfügung stehen.

2. Application Domain Extensions

Für gut strukturierte, anwendungsspezifische Informationen ist es erstrebenswert diese auf systematischem Wege zu repräsentieren. Zur Erweiterung des CityGML-Datenmodells können auch sogenannte Application Domain Extensions (ADEs) verwendet werden.

Ein ADE-Schema kann ein oder mehrere CityGML-Schemas erweitern. Die relevanten CityGML-Modulschemas müssen durch das ADE-Schema importiert werden. Das ADE-Konzept definiert eine spezielle Möglichkeit existierende CityGML-Featuretypes zu erweitern. Es erlaubt die gleichzeitige Verwendung von verschiedenen ADEs innerhalb des gleichen Instanzdokuments. Jedes ADE wird durch sein eigenes XML-Schemafile spezifiziert, welches verschiedene Erweiterungen von CityGML definierten kann.

Allerdings gehören alle Erweiterungen zu einer der beiden folgenden Kategorien:

- Neue Featuretypes werden innerhalb des ADE Namensraum definiert und basieren auf abstrakten (z.B. *_CityObject*) oder konkreten (z.B. *CityFurniture*) CityGML-Klassen. Diese neuen Featuretypes werden demnach von existierenden CityGML Featuretypes hergeleitet.
- Existierende CityGML Featuretypes werden durch anwenderspezifische Eigenschaften im ADE-Namensraum erweitert. Diese Eigenschaften können einfache oder komplexe Datentypen enthalten. Auch Geometrien oder eingebaute Features sind möglich [Gröger et al., 2012].

4 Erfassung semantischer Straßenraummodelle mittels Mobile Mapping

Mobile Mapping Systeme werden zur schnellen Erfassung von Umgebungsinformationen eingesetzt. Diese Messsysteme arbeiten mittels direkter Georeferenzierung und sogenannten Multi-Sensor-Systemen, einer Kombination aus verschiedenen zeit-synchronisierten Sensoren, die sich auf einer mobilen Plattform befinden [Schwarz und El-Sheimy, 1996; El-Sheimy, 2005]. Der Vorteil von MMS im Vergleich zur statischen Messmethode ergibt sich aus der Kombination einer hohen Datenrate mit einem wesentlich geringeren Zeitaufwand zur Erzeugung von dichten Punktwolken. Aufgrund der Integration verschiedener Sensoren zur Erfassung der Umgebung ist beispielsweise eine Ergänzung der 3D-Informationen um Farbinformationen durch Kameras möglich [El-Sheimy, 2005]. In diesem Kapitel soll zum einen der generelle Erfassungsprozess eines MMS aufgezeigt, zum anderen aber auch auf den speziellen Ablauf der Datenerfassung bei 3D Mapping Solutions eingegangen werden.

4.1 Genereller Erfassungsprozess

MMS kombinieren Sensoren der Positionierung und Orientierung mit Sensoren der Umgebungserfassung auf einer mobilen Plattform. Als Plattform für die Datenerfassung aus der Luft werden Flugzeuge, Helikopter und Unmanned Aerial Vehicles (UAVs) verwendet [Baltsavias, 1999]. Am Boden hat sich der PKW neben Zügen und Schiffen als Plattform zur mobilen Erfassung etabliert. Ellum und El-Sheimy [2002], Barber et al. [2008] und Petrie [2010] geben eine Übersicht über solche mobilen terrestrischen Systeme und nehmen eine Aufteilung in Systemhersteller und Serviceanbieter im Bereich der MMS vor. In der Anfangszeit der mobilen Datenerfassung wurden MMS wie das System MoSES der Firma 3D Mapping Solutions meist durch den Anwender entwickelt und gebaut. Der Trend geht nun zu modularen kompakten, aber leichteren Systemen für den flexiblen Einsatz auf unterschiedlichen Plattformen. Zusätzlich werden vermehrt Sensoren mit kostengünstigen Lösungen für die direkte Georeferenzierung eingesetzt. Auch geht der Trend zur kollaborativen Vermessung mit vernetzten Multisensorplattformen [Tao und Li, 2007]. In den folgenden Abschnitten wird auf generelle Schritte der Datenerfassung und Auswertung kinematischer Vermessungsdaten eingegangen.

4.1.1 Datenerfassung und Vorverarbeitung

Schwarz und El-Sheimy [1996] gliedern die Datenerfassung bei MMS in die fünf Bereiche Datenerfassung, Synchronisierung und Georeferenzierung, Integration und Datenfusion, Qualitätskontrolle, Optimierung und Automatisierung des Datenflusses. Sie sollen an dieser Stelle kurz vorgestellt werden.

Die **Datenerfassung** selbst erfolgt durch direkt auf der Plattform integrierte Sensoren der Umgebungserfassung und Sensoren zur Positionierung und Orientierung des Messsystems. Durch die direkte Befestigung der kalibrierten Navigationssensoren, meistens GNSS-Empfänger und IMU, auf der Plattform, können von nahezu jeder beliebigen Fahrzeugposition Signale von Satelliten empfangen werden [Ellum und El-Sheimy, 2002]. Die Umgebungserfassung erfolgt über CCD-Kameras, Laserscanner, Radar- und Ultraschallsensoren oder durch eine Kombination dieser Sensoren. In den meisten Fällen werden mehr als eine Kamera auf der mobilen Plattform befestigt. Dies erlaubt Stereoaufnahmen und dreidimensionale Objektvermessungen. Die digitalen Kamerabilder werden durch die Navigationssensoren direkt georeferenziert [Tao und Li, 2007]. Laserscanner messen die Distanz und Richtung vom Scanner zu einem Zielpunkt. Dafür wird die Position des Scanners als Koordinatenursprung benötigt. Im statischen Fall bleibt der Bezugspunkt fest, wodurch die Messpunkte über polares Anhängen an den Standpunkt eine Punktwolke mit lokaler Orientierung liefern. Im kinematischen Fall bewegt sich der Bezugspunkt zwischen den Messungen. Da die Position des Bezugspunktes zur Berechnung der Punktwolke für jede Messung bekannt sein muss, wurde für den kinematischen Fall die direkte **Georeferenzierung** mittels satellitengestützter Verfahren in Kombination mit zusätzlichen inertialen Sensoren zur Verbesserung der Positionsbestimmung entwickelt [Schwarz und El-Sheimy, 1996]. Durch die direkte Georeferenzierung entfällt eine aufwändige indirekte Georeferenzierung über mehrere Anschlusspunkte wie es bei einer Totalstation der Fall ist. Untersuchungen von Nebiker et al. [2013] zeigen allerdings, dass durch die Verwendung von terrestrisch bestimmten Passpunkten nachweislich eine Steigerung der Georeferenzierungsgenauigkeit erreicht werden kann. Die integrierte Georeferenzierung, eine Kombination aus indirekter Georeferenzierung mit Passpunkten und direkter Georeferenzierung mit Navigationssensoren, kann vor allem in urbanen Bereichen, in denen die Positionsbestimmung durch GNSS-Sensoren oftmals weniger genau ist, Abhilfe schaffen [Haring, 2007].

Die **Synchronisation** der Sensoren ist entscheidend für die Georeferenzierung. Zur Transformation aller erfassten Daten in ein gemeinsames Koordinatensystem wird in der Regel ein einheitliches Zeitsignal durch GNSS bereitgestellt und auf alle Sensoren übertragen [Schwarz und El-Sheimy, 1996]. Für die Synchronisation der Sensoren müssen alle verwendeten Sensoren zunächst einzeln kalibriert sein. In der Regel geschieht dies vom Hersteller bereits vorab. Die relative Orientierung der Systemsensoren zueinander, häufig auch als Systemkalibrierung bezeichnet, muss in regelmäßigen Abständen auf Testfeldern überprüft werden. Grundsätzlich kann der GNSS-Antennen-Offset durch Einmessen mit einer Totalstation bestimmt werden. Durch identische Punkte in den Daten der Umgebungssensoren ist die relative Orientierung einer Kamera bezüglich eines Laserscanners bestimmbar. Dafür können markante Kanten, Punkte oder retro-reflektierende Zielmarken verwendet werden [Har-

ing, 2007]. Unter der **Integration und Datenfusion** versteht man alle Schritte, die für das Erreichen eines gewünschten Ergebnisses, wie zum Beispiel der Erzeugung von 3D-Koordinaten aus den Sensordaten, erforderlich sind. Dieser Schritt beinhaltet die Integration von verschiedenen Datenarten aus unterschiedlichen Datenquellen [Schwarz und El-Sheimy, 1996]. Dafür ist das Wissen über die Orientierung der Umgebungssensoren (Kamera, Laserscanner) bezüglich eines übergeordneten Systems wie dem Landeskoordinatensystem notwendig. Auch müssen die einzelnen Laserscandaten beispielsweise mit Hilfe von 3D-Matching-Algorithmen wie dem ICP-Algorithmus zueinander registriert werden. Mit zusätzlichen Beobachtungen wie GNSS ist die Bestimmung absoluter Orientierungsparameter (Georeferenzierung) möglich [Haring, 2007]. Eine **Qualitätskontrolle** findet in Echtzeit während der Datenaufnahme und in der Nachverarbeitung statt. In Echtzeit wird geprüft, ob die erfassten Daten die geforderte Genauigkeit mit einer definierten Wahrscheinlichkeit einhalten. Eine Aussage darüber, wie viele der Messungen die geforderte Genauigkeit erreichen, liefert die Kontrolle während der Nachverarbeitung [Schwarz und El-Sheimy, 1996]. Solche MMS können die Fahrzeugposition mit einer Genauigkeit im Zentimeterbereich bestimmen. Durch die Erzeugung hochgenauer Punktwolken und die durch relative Vermessungsmethoden genaue und verlässliche Georeferenzierung ist eine teilweise vollautomatische Objektextraktion möglich. Kaartinen et al. [2012] realisieren einen Leistungsvergleich von mehreren mobilen Laserscansystemen durch die Verwendung eines permanent Testfeldes. Die beste erreichte Genauigkeit der Höhenkomponente lag bei 3,5 cm bei einer Reichweite von 35 m. Die beste Lagegenauigkeit sogar bei 2,5 cm für eine Reichweite von 45 m. Weitere Genauigkeitsstudien von mobilen Laserscansystemen können beispielsweise in Barber et al. [2008] und Brenner [2009] gefunden werden. Darauf aufbauend erfolgt eine **Optimierung und Automatisierung des Datenflusses**. Neben einem sehr guten Verständnis der mathematischen Grundlagen erfordert der Prozess zur Erstellung der Ergebnisse auch ein hohes Maß an menschlicher Interaktion [Schwarz und El-Sheimy, 1996].

4.1.2 Auswertung und Modellierung

Die kinematisch erfassten Laserscandaten und Kameradaten werden zur Detektion und Extraktion unterschiedlicher Objekte verwendet. Es können beispielsweise Gebäude mit Fassaden rekonstruiert werden, aber auch Elemente des Straßenraumes wie Fahrbahnen und Markierungen zur Darstellung in Karten. Die Literatur zu diesem Thema ist umfassend. An dieser Stelle wird lediglich ein Überblick über mögliche Objektextraktionsverfahren gegeben.

Yang et al. [2013] stellen eine Methode zur Extraktion und Darstellung von Straßen basierend auf mobilen Laserscanscandaten vor. Die Vollständigkeit, die Korrektheit und die Qualität der extrahierten Straßen werden mit 94.42%, 91.13% und 91.3%, angegeben. Hohe Punktdichten erlauben eine Extraktion von Objekten, die traditionell eher durch photogrammetrische Verfahren oder die klassische terrestrische Vermessung aufgenommen werden. In urbanen Gebieten werden Bordsteine oft genutzt, um die Straßenoberfläche von benachbarten Gehwegen zu separieren. Zhou und Vosselman [2012] vergleichen die Detektion von Bordsteinen aus Punktwolken von ALS und MLS. Die Vollständigkeit ist dabei in ALS-Daten aufgrund parkender Fahrzeuge höher als in MLS-Daten. Die Genauigkeit von

sichtbaren Bordsteinen ist durch die geringere Entfernung aber im terrestrischen Verfahren höher. Jaakkola et al. [2010] extrahieren aus Daten eines Multisensorsystems, welches auf UAVs oder Fahrzeugen befestigt werden kann, pfostenähnliche Objekte. Auch hier kann eine hohe Vollständigkeit von 90% und eine Korrektheit von 86% erreicht werden. Stark reflektierende Objekte wie Straßenmarkierungen werden durch Intensitätswerte der Laserpunktewolken detektiert [Yang et al., 2012]. Tao et al. [2001] geben einen Überblick über die automatische Weiterverarbeitung von Bildsequenzen aus Mobile Mapping Verfahren. Durch die Stereoanordnung des Kamerasystems können Objekte detektiert werden. MMS werden hauptsächlich zur Aufnahme des Straßenraums verwendet und teilweise zur Detektion von Gebäuden verwendet. Einen weiterführenden Überblick über Literatur im Bereich der Straßenraummodellierung gibt Kapitel 2.

4.2 Erfassungsprozess bei 3D Mapping Solutions

Die Bearbeitung der Masterarbeit erfolgt am Lehrstuhl für Geoinformatik der Technischen Universität München in Kooperation mit der Firma 3D Mapping Solutions. Sie wurde im März 2007 als Spin-off der Bundeswehr Universität München gegründet und beschäftigt mittlerweile 35 Mitarbeiter. Die Firma hat ihren Sitz in Holzkirchen, Landkreis Miesbach, Deutschland und in Pittsburgh, PA, USA. Momentan arbeitet das Unternehmen mit zwei Mappingsystemen in Europa, jeweils ein System in Amerika und in China, sowie einem System, welches derzeit in Japan aufgebaut wird. Zusätzlich sind zwei variabel einsetzbare Systeme für Messungen weltweit verfügbar.

Das beteiligte Unternehmen ist auf die kinematische Vermessung von Verkehrsnetzen, die Auswertung von kinematischen Vermessungsdaten und die Kalibrierung von Multisensorsystemen spezialisiert. Das Hauptaugenmerk liegt auf der kinematischen Vermessung von Straßennetzen, Bahnstrecken und Tunneln. Zu den Verkehrsnetzen zählen auch Rennstrecken, Teststrecken und Werksgelände, sowie Autobahnen und Innenstädte. Das Aufgabenspektrum reicht dabei von der Straßenzustandserfassung, Bilddokumentation des Verkehrsraumes und Grundlegendatenerfassung für 3D-Stadtmodelle, hin zur Erzeugung hochgenauer digitaler 3D-Oberflächenmodelle der Fahrbahn und Modellierung von Umgebungskarten. Derzeit liegt der Schwerpunkt auf der Bereitstellung von Referenzinformationen für fortgeschrittene Fahrerassistenzsysteme und auf der Erstellung hochgenauer Referenzkarten von Straßennetzen für das autonome Fahren. Die angebotenen Produkte finden zusätzlich Anwendung in der Fahrzeugnavigation und in Simulationen für die Fahrzeugentwicklung. 3D Mapping Solutions steht demnach in engem Zusammenhang mit der Automobilindustrie und arbeitet für Automobilhersteller weltweit [3D Mapping Solutions GmbH, 2017].

4.2.1 Aufbau des Messsystems und kinematische Vermessung

Für die Datenerfassung werden spezialisierte Messfahrzeuge mit dem selbst entwickeltem Mobile Mapping System MoSES (**M**obiles **S**traßen-**E**rfassungs-**S**ystem) bestehend aus einem Sensoraufbau und einem Kontrollsystem ausgestattet. Der Standardaufbau setzt sich aus einem Multisensorsystem



Abbildung 4.1 – Aufbau des Mobile Mapping Systems MoSES (Bildquelle: 3D Mapping Solutions)

mit Sensoren zur Umgebungserfassung und Sensoren zur Orientierung und Lokalisierung zusammen, wie in Abb. 4.1) dargestellt.

Zwei Hochleistungslaserscanner, ein exakt kalibriertes Multikameramesssystem und ein Digitalbarometer werden zur Erfassung der Umgebung und Digitalisierung des Straßenkorridors eingesetzt. Die Laserscanner erzeugen Punktwolken mit dreidimensionalen Punktkoordinaten und Intensitätswerten für jeden Punkt. Durch die Anordnung von zwei unterschiedlich zueinander orientierten Scannern werden Punkte zu zwei unterschiedlichen Zeiten gemessen und eine hohe innere Genauigkeit erreicht. Außerdem können dadurch tote Winkel auf ein Minimum reduziert werden und so vor allem langgestreckte Objekte wie Straßenzüge und Tunnel verlustfrei abgebildet werden. Hohe Punktdichten können durch Scanraten von bis zu 100 Hz auch bei normalen Fahrgeschwindigkeiten erreicht werden. Bei einer Geschwindigkeit von $v = 100$ km/h ist eine absolute Genauigkeit der Messpunkte von 0,02 m möglich.

Zusätzlich zu den Laserscannern wird der Verkehrsraum mit einem Kamerasystem in Stereobildpaaren aufgenommen. Auf dem Messsystem MoSES können bis zu acht Kameras angebracht werden. Eine Kombination aus Farb-, Infrarot- und Grauwertkameras mit unterschiedlichen Auflösungen ist möglich. Die aufgenommenen Bilder sind dabei exakt georeferenziert, kalibriert und werden bei der weiteren Auswertung für photogrammetrische Objektmessungen verwendet.

Die Lokalisierung des Fahrzeuges erfolgt durch einen GNSS-Empfänger zur genauen Positionsbestimmung mittels Satelliten und einen am Autoreifen befestigter Odometer. Der Odometer misst mechanisch die zurückgelegte Strecke des Fahrzeuges und ist besonders wichtig zur Erkennung von Standphasen. Zur vollständigen Kompensation der Fahrzeugbewegungen und zur Unterstützung der Trajektorie des GNSS-Empfängers wird ein hochgenaues Inertialmesssystem (Inertial Mea-

surement Unit, IMU) eingesetzt. Das Kontrollsystem im Fahrzeug vervollständigt das Messsystem.

Im Anschluss an die Befahrung wird die Trajektorie des Fahrzeuges bestimmt. Sie umfasst die dreidimensionale Position und die zugehörigen Lagewinkel im Raum. Zur Bestimmung der Trajektorie nutzt das Messsystem MoSES eine Kombination aus einem Differentielles Globalen Positionierungssystem (DGPS), IMU und Odometer sowie Barometer und Inklinometer als zusätzliche Sensoren. Im Falle von MoSES steht zu diesem Zweck das System Applanix POS/LV 420 zur Verfügung [Gräfe et al., 2001]. Die IMU bildet die Bewegung des Fahrzeuges im Raum ab. Mithilfe optimaler Schätzverfahren (Kalman-Filter) werden alle Sensordaten weiterverarbeitet. Die Genauigkeit der Trajektorie wird durch die IMU und die anschließende Filterung der Messwerte garantiert. Eine gleich bleibend genaue und kontinuierliche Ermittlung der Trajektorie ist für die anschließende Orientierung der Sensordaten und Georeferenzierung notwendig. GNSS-Empfangslücken und Messunsicherheiten von drei Minuten können durch den Einsatz der IMU und anschließender Kalman-Filterung in Lage und Höhe der Trajektorie von unter 0,5 m ausgeglichen werden.

Für die kinematische Vermessung ist ein festes geometrisches Bezugskordinatensystem für alle Sensormodule notwendig. Die Georeferenzierung erfolgt durch die Sensoren der Positionierung. Für eine integrierte Georeferenzierung mit dem zusätzlichen Einsatz von Passpunkten kann die Genauigkeit der Positionsbestimmung für bestimmte Anwendungsfälle erhöht werden. Zusätzlich müssen alle Einzelsensoren kalibriert werden. Alle ermittelten Messdaten des Sensorsystems müssen hochpräzise zeitlich zur Trajektorie synchronisiert sein. Damit sind die Voraussetzungen für die kinematische Vermessung erfüllt. Dafür werden die geometrische Lage aller Sensoren zueinander, die Kalibrierungsparameter aller Sensoren und die exakte Synchronisation aller Sensormodule vorausgesetzt. Eine regelmäßige Kalibrierung der Einzelsensoren und des Gesamtsystems sind für gleichbleibende Genauigkeitsmaße notwendig. Das Gesamtsystem wird auf exakt vermessenen Teststrecken, das Kamerasystem und die Laserscanner durch spezielle Testfelder geprüft. Mit externen Kontrollmessungen werden die Ergebnisse regelmäßig überprüft.

Dieser Messaufbau ermöglicht eine vollständige, dreidimensionale Erfassung aller Bewegungen des Fahrzeuges bei Aufnahmegeschwindigkeiten von bis zu $v = 100$ km/h. Die Messleistung ist abhängig vom Einsatzgebiet und liegt zwischen 100 km/Tag und 400 km/Tag. Die relativen Genauigkeitsmaße des Messsystems liegen bei $< 0,05$ m für die photogrammetrische Auswertung und bei $< 0,005$ m im Bereich der Laserscandaten. Die absolute Genauigkeit der Positionsbestimmung liegt bei 0,02 m. Die Genauigkeitsmaße können nach Anwendungsfall variieren. Eine Genauigkeit im Meter-Bereich setzt ein DGPS mit einfachen Korrekturdaten, die global verfügbar sind (z.B. Omnistar), voraus. Diese Genauigkeit ist für Navigationsdaten und die einfache Aufnahme von Straßennetzen ausreichend. Für die Aufnahme von Straßennetzen für die Straßeninformationsbank oder für Fahrassistenzsysteme wird eine Genauigkeit im Dezimeter-Bereich verlangt. Diese wird durch den zusätzlichen Einsatz von Referenzstationen wie z.B. SAPOS in Deutschland erreicht. Bei der Messung von Fahrbahndecken oder Oberflächenmodelle kann durch die Nutzung von Passpunkten eine statische Genauigkeit im Zentimeter-Bereich erreicht werden. Mit diesem Messsystem werden Straßennetze, Bahnstreck-

en, Tunnel, Rennstrecken, Teststrecken und Werksgelände vermessen. Das Ergebnis der kinematischen Vermessung mit diesem Messsystem ist eine Trajektorie, sowie georeferenzierte Punktwolken und Kameradaten. Diese werden innerhalb der Firma in anschließenden Schritten zu hochgenauen Oberflächenmodellen und Umgebungskarten weiterverarbeitet [3D Mapping Solutions GmbH, 2017].

4.2.2 Auswertung und Modellierung

Die Weiterverarbeitung der durch die Befahrung kinematisch erfassten Punktwolken und Kameradaten wird je nach Kundenanforderung an das Endprodukt entweder von der Abteilung *Geodetic Analytics* (früher: Ingenieurvermessung) oder der Abteilung *HD Maps* (früher: OpenDRIVE) übernommen. Erstere hat sich vor allem auf die Erzeugung hochgenauer digitaler 3D-Oberflächenmodelle der Fahrbahn spezialisiert. Der Schwerpunkt der zweiten Abteilung liegt auf der Bereitstellung von Referenzinformationen für fortgeschrittene Fahrerassistenzsysteme und auf der Erstellung hochgenauer Referenzkarten von Straßennetzen für das autonome Fahren. Die angebotenen Produkte finden dabei zusätzlich Anwendung in der Fahrzeugnavigation und in Simulationen für die Fahrzeugentwicklung. Im Folgenden wird auf die Weiterverarbeitungsschritte der Abteilung *HD Maps* eingegangen. Die Referenzkarten der Straßennetze werden dem Kunden hauptsächlich im OpenDRIVE-Standard bereitgestellt. Weit verbreitet ist dieser hauptsächlich in der Automobilindustrie bei der Entwicklung und zur Simulation im Bereich des autonomen Fahrens. Durch diesen Modellierungs- und Datenaustauschstandard werden der Verlauf und die Umgebung einer Fahrbahn parametrisch unter Verwendung der XML definiert. Auch Austauschformate wie DXF und DWG von Autodesk oder andere Simulationsformate für den CarMaker von IPG werden unterstützt.

Der Datenerfassungsprozess von den Rohdaten bis zum fertigen OpenDRIVE-Dokument wird in Abbildung Abb. 4.2 dargestellt. Zur Vorverarbeitung gehört die Korrektur der Trajektorie, die Aufbereitung der Kameradaten und das Matching der Punktwolken. Unter Matching wird die Registrierung (Orientierung) der Punktwolken zueinander verstanden. Die korrigierte Trajektorie und die vorverarbeiteten Kameradaten und Punktwolken dienen als Grundlage für die Auswertung der Daten, d.h. für die Extraktion aller Elemente der Straße und der Objekte entlang der Straße.

Die aufgenommenen Kamerabilder sind exakt georeferenziert und werden für photogrammetrische Objektmessungen verwendet. Informationen über relevante Objekte des Straßenraumes wie Straßenmarkierungen, Verkehrsschilder und Ampeln können mittels der intern entwickelten photogrammetrischen Software 3D Road View gewonnen werden. Die Software dient der Bilddarstellung, der Datenverwaltung und der photogrammetrischen Bildmessung des Multikamerasystems.

Verkehrszeichen und Ampeln werden anhand der Kameradaten klassifiziert. Die Verkehrszeichen werden nach den Bestimmungen des jeweiligen Aufnahmelandes eingeordnet. In Deutschland wird der Verkehrszeichenkatalog der Bundesanstalt für Straßenwesen (BASt) verwendet. Der Katalog enthält alle bundesweit amtlich zugelassenen Verkehrszeichen. Dazu gehören neben Verkehrszeichen, die zur Straßenverkehrsordnung gehören, auch andere von dem zuständigen Bundesministerium veröffentlichte Verkehrszeichen. Die Zeichen werden im Katalog in die Hauptkategorien

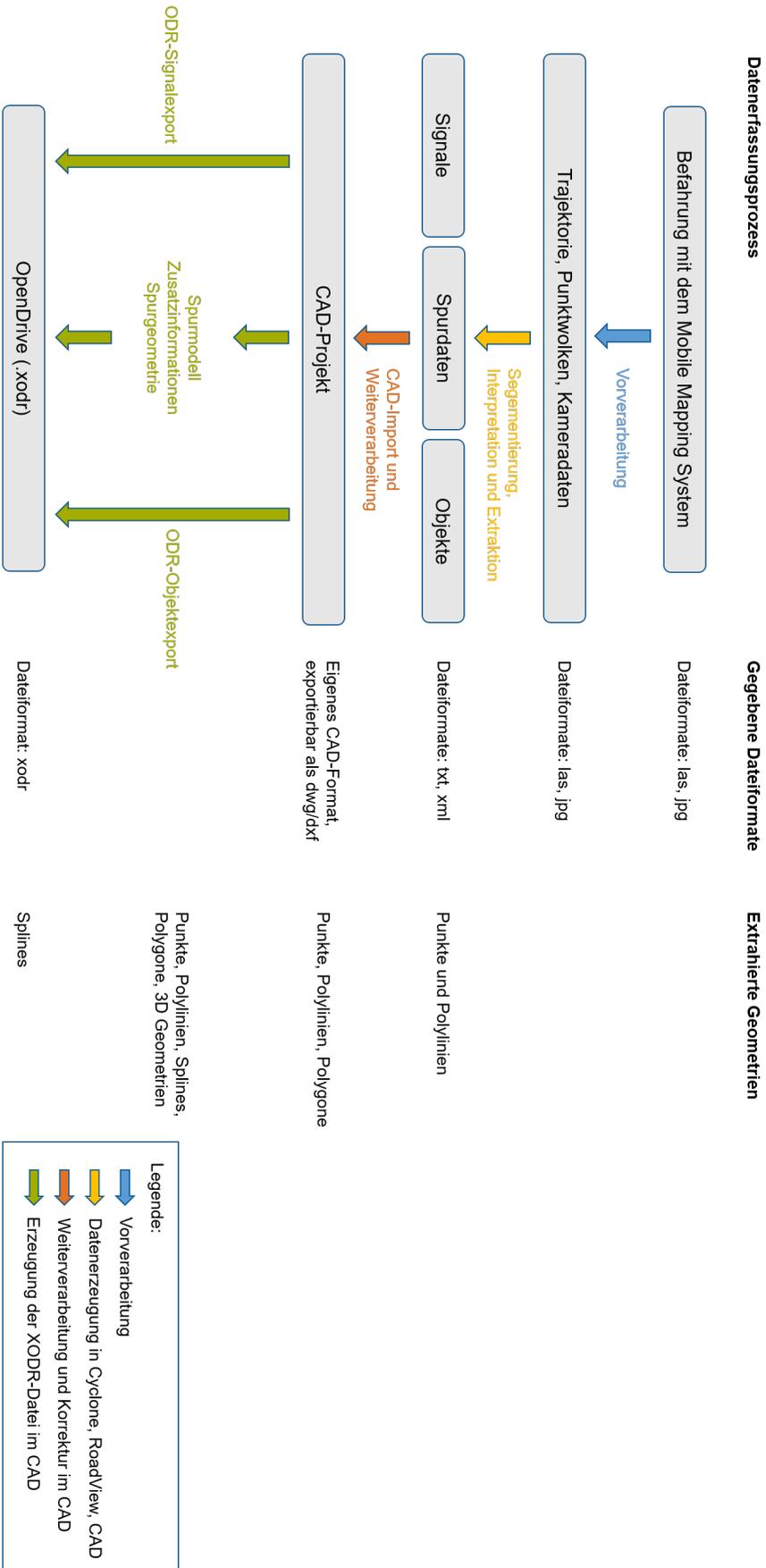


Abbildung 4.2 – Erfassungsprozess bei 3D Mapping Solutions

Gefahrenzeichen, Vorschriftzeichen, Richtzeichen, Verkehrseinrichtungen und Zusatzzeichen und mithilfe nummerierter Unterkategorien eingeteilt. Die Kategorisierung der Ampeln erfolgt nach der in der OpenDRIVE-Spezifikation dargestellten Tabelle unter 6.11 Signal Types.

Aus den Punktwolken werden in der Punktwolkenverarbeitungssoftware Cyclone von Leica alle möglichen Fahrspuren und Objekte wie Bäume und Straßenlaternen extrahiert. Fahrbahnrande und Straßenmarkierungen werden mittels Linienextraktion gewonnen. Die Datenextraktion wird dabei schon an das gewünschte Endprodukt, in diesem Fall OpenDRIVE, angepasst. Auch kundenspezifischen Wünsche werden mit aufgenommen. Tabelle 4.1 listet alle möglichen Spurtypen der OpenDRIVE-Spezifikation auf. Die Spurtypen *special 1-3* ermöglichen das Einfügen von alternativen Spurtypen. Zusätzlich werden von 3D Mapping Solutions die beiden internen Formate *green* für Grünflächen und *gutter* für Rinnen ergänzt.

Tabelle 4.1 – Mögliche Spurtypen in OpenDRIVE

OpenDRIVE Spurtyp	Beschreibung
none	unbekannt
driving	Fahrspur
sidewalk	Gehweg
biking	Fahrradweg
parking	Parkfläche
border	Randspur
shoulder	befestigte Seitenstreifen
stop	Standspur
restricted	nicht zugängliche Flächen
roadWorks	Bauarbeiten auf der Spur
bidirectional	bidirektionale Spur
median	mittlerer Trennstreifen
entry	Auffahrt
exit	Abfahrt
offRamp	Verbindungsrampe bei Autobahnen
onRamp	Verbindungsrampe bei Autobahnen
special1	-
special2	-
special3	-
tram	Tramspur
rail	Zugspur

Tabelle 4.2 zeigt alle in OpenDRIVE implementierten, möglichen Objektarten. Intern werden die Objekttypen *roadmark* für Straßenmarkierungen und *crosswalk* als Fußgängerüberweg verwendet. In einem speziellen CAD-Programm werden die bis dahin extrahierten Daten eingelesen, korrigiert und aufbereitet. Interne Softwarepakete ermöglichen die Konvertierung der CAD-Daten in das OpenDRIVE-Format. Ein wichtiger Bestandteil der Auswertung ist die Umwandlung der extrahierten Daten in den OpenDRIVE-Standard. Über Referenzachsen wird dabei die relative Lage der Fahrbahn und Umgebung definiert. Fahrspuren werden für jede Achse über die Breitenangabe berechnet. Außerdem wird die Fahrbahn in Abschnitte eingeteilt, wobei ein neuer Abschnitt beginnt, sobald sich die Geometrie

Tabelle 4.2 – Mögliche Objekttypen in OpenDRIVE

OpenDRIVE Objekttypen		
none	bike	vegetation
obstacle	motorbike	barrier
car	tram	building
truck	train	parkingSpace
van	pedestrian	wind
bus	pole	patch
trailer	tree	

des Fahrbahnverlaufs ändert. Dabei werden verschiedene Spurtypen wie Fahrbahnen, Gehwege, Parkplätze und Grünflächen differenziert. Das OpenDRIVE-Dokument wird durch die im CAD-Projekt gesammelten Informationen erstellt. Das Dokument beinhaltet Informationen über Spurtypen und Spurgeometrien, stellt aber auch Zusatzinformationen wie Geschwindigkeiten und Neigungen der Fahrspur bereit.

Die Weiterverarbeitung der extrahierten Plandaten wird dementsprechend an die Anforderungen des OpenDRIVE-Formats angepasst. Aus den dreidimensionalen Kurven werden zweidimensionale Plandaten erstellt, die wiederum der Spurberechnung für das OpenDRIVE-Dokument dienen. Aus den gegebenen Spurdaten und den Punktwolken werden zweidimensionale und dreidimensionale Achsen für jede Fahrbahn berechnet, die als Referenzlinien verwendet werden. Für jede Referenzlinie werden wiederum Spurdaten berechnet, die zur Beschreibung der unterschiedlichen Spuren im OpenDRIVE-Format dienen. Aus der Spurberechnung heraus entstehen im CAD zweidimensionale Schraffuren, die unter anderem zur Überprüfung des richtigen Spurtyps verwendet werden. Zusätzlich werden für die visuelle Überprüfung dreidimensionale Geometrien aus den berechneten Spurdaten generiert (vgl. Abb. 4.3). Die Signale und Objekte werden den Achsen zugewiesen und erhalten beim Export eine eindeutige Punktnummer.

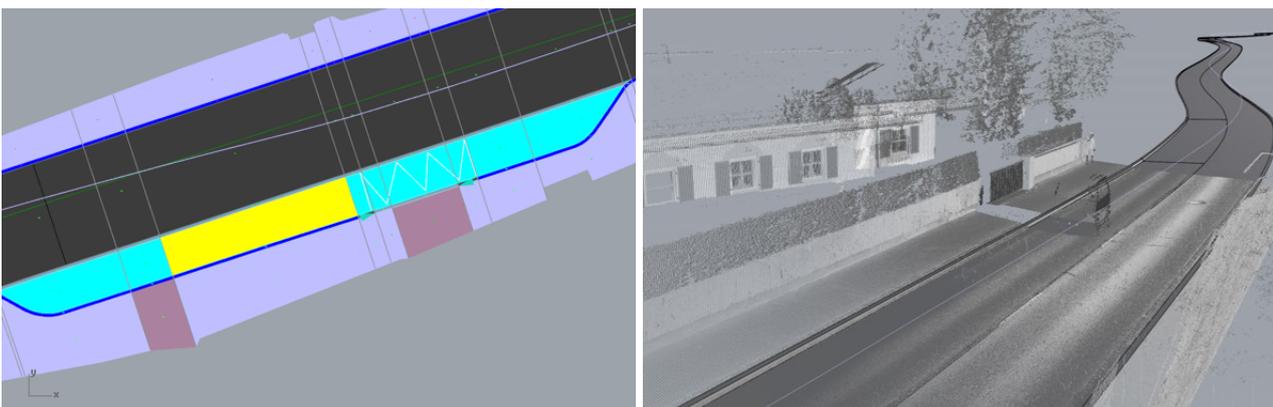


Abbildung 4.3 – Darstellung der 2D Schraffuren (links) und 3D Geometrien in der Punktwolke (rechts) (Bildquellen: 3D Mapping Solutions)

Das OpenDRIVE-Dokument letztendlich Straßen und Kreuzungen mit Fahrspuren, Referenzachsen mit zugewiesenen Objekte und Signalen und Zusatzinformationen wie zulässige Geschwindigkeiten.

Außerdem werden Vorgänger-, Nachfolger- und Nachbarschaftsbeziehungen der Referenzachsen und Fahrspuren sowie Fahrbahnneigungen beschrieben. Das Ergebnis ist demnach ein hochgenauer Bestandsplan der Umgebung und ein logisches Verkehrsnetz mit topologischer Information. Zusätzlich können Straßenoberflächenmodelle im OpenCRG-Format hinzugefügt werden. Die Umgebungskarte kann als digitales Abbild der Realität und als Abgleich zwischen Ist- und Soll-Situation verwendet werden. Es dient aus diesen Gründen vermehrt als Basis für das autonome Fahren. Die Semantik wird abschließend durch einen firmeneigenen Validator und dem OpenDRIVE-Validator geprüft. Visuell kann das Ergebnis im OpenDRIVE-Viewer dargestellt werden.

5 Abbildung des OpenDRIVE-Datenmodells auf das CityGML-Datenmodell

An dieser Stelle sollen die beiden Modellierungsstandards OpenDRIVE und CityGML in Bezug auf die Straßenraummodellierung verglichen werden. Als Grundlage werden die in Kapitel 3 beschriebenen Spezifikationen herangezogen. Hintergrund der Arbeit ist eine mögliche Erweiterung des Erfassungsprozesses aus Abschnitt 4.2 zur Visualisierung der Daten. Dafür soll eine Erweiterung des OpenDRIVE-Datenmodells zur zusätzlichen Generierung eines CityGML-Modells diskutiert werden. Aus diesem Grund erfolgt in diesem Kapitel eine unidirektionale Betrachtung von OpenDRIVE zu CityGML. An manchen Stellen wird dennoch das Verhältnis der beiden Datenmodelle beschrieben. Im ersten Abschnitt werden die Inhalte und Anwendungsbereiche der beiden Standards dargestellt. Die Ähnlichkeiten und Unterschiede sowie der Aufbau der Spezifikationen werden im zweiten Abschnitt diskutiert. Der direkte Vergleich der beiden Datenmodelle wird auf den vier Modellierungsebenen, die in Kapitel 2 Abb. 2.1 angesprochen wurden, vorgenommen. Der Vergleich beruht demnach auf den Merkmalen Geometrie, Topologie, Semantik und visuelle Erscheinung. Ziel dieses Kapitels ist es, Ähnlichkeiten und Gegensätze von OpenDRIVE und CityGML darzulegen. Das Kapitel schließt mit einem semantischen Vergleich der für diese Arbeit wichtigen Klassen und Attribute. Für die spätere Konzeptentwicklung dient dieses Kapitel als Grundlage.

5.1 Inhalte und Anwendungsbereiche

OpenDRIVE wurde von der VIRESS Simulationstechnologie GmbH in enger Zusammenarbeit mit Daimler Driving Simulator entwickelt. Der Industriestandard wird von einem Team aus Simulationssanwendern weiterentwickelt, unterstützt und gepflegt. Die Spezifikation beinhaltet eine einfache XML-basierte hierarchische Struktur zur Beschreibung aller geometrischen Objekte eines Straßennetzwerkes, die die Logik des Straßenverlaufs beeinflussen. OpenDRIVE repräsentiert demnach hauptsächlich die geometrischen und topologischen Beziehungen zwischen Objekten. Das Hauptanwendungsgebiet liegt bei der digitalen Beschreibung von Verkehrsnetzen. Der OpenDRIVE-Standard wird in der Automobilindustrie und Forschung zur Straßenbeschreibung und Erzeugung virtueller Landschaften auf Grundlage realer Daten verwendet. Er wird in Fahr- und Verkehrssimulatoren und in Fahrerassistenzsystemen, speziell in der Navigation, eingesetzt. **CityGML** ist ein internationaler OGC-Standard für die Repräsentation, Speicherung und den Austausch von virtuellen 3D-Stadtmodellen und ist ein Anwendungsschema für die GML. CityGML repräsentiert nicht nur das geometrische

Erscheinungsbild von Stadtmodellen, sondern befasst sich auch mit der Repräsentation von semantischen, topologischen und visuellen Eigenschaften. Es erlaubt eine Übermittlung von dreidimensionalen Objekten im Bereich der Geodaten.

OpenDRIVE und CityGML sind beides offene, XML-basierte Modellierungsstandards. Der grundlegende, thematische Unterschied liegt in der Art und dem Umfang der Modellierung. Während OpenDRIVE ein Industriestandard zur logischen Beschreibung von Straßennetzen ist, wird CityGML zur Verwaltung und Repräsentation virtueller 3D-Stadtmodelle verwendet. Der größte Unterschied der beiden Spezifikationen liegt in der Beschreibung der Geometrieobjekte. OpenDRIVE beschreibt alle Objekte des Straßenraums mittels linearer Referenzierung. Alle Objekte werden mit Entfernungen zu einer Referenzlinie angegeben. Der Vorteil davon ist, dass viele unterschiedliche Attributwerte einer Straße erfasst werden ohne dass bei einer Änderung des Attributwerts eine Segmentierung der Straße in kleinere Teilbereiche erforderlich ist. Diese Art der Beschreibung ist in CityGML nicht möglich, was eine Abbildung der beiden Datenmodelle sichtlich erschwert. Auf die Modellierung der Geometrie wird in Unterabschnitt 5.3.1 genauer eingegangen.

5.2 Formaler Aufbau der Datenmodelle

Im Folgenden werden Ähnlichkeiten und Unterschiede des formalen Aufbaus der Spezifikationen beschrieben. Als Hauptunterschied der beiden Datenmodelle ist der thematische Inhalt zu nennen. Dieser ist im CityGML-Datenmodell wesentlich umfassender. Mittels CityGML ist die Modellierung einer ganzen Stadt möglich, während OpenDRIVE lediglich der Modellierung von Straßensystemen dient. Zur Beschreibung des Inhalts ist das CityGML-Datenmodell thematisch in ein Kernmodul (Core module) und in mehrere Erweiterungsmodul (Extension modules) unterteilt. Für jedes Modul besteht eine XML-Schemadatei. Im Vergleich dazu werden alle OpenDRIVE-Straßenelemente in einer einzelnen XML-Schemadatei beschrieben.

Der Unterschied in der formalen Dokumentation der beiden Spezifikationen besteht darin, dass CityGML sehr ausführlich als UML-Schema ausgearbeitet und dieses in einem XML-Schema umgesetzt wird. Alle Klassen und Attribute mit Vererbungs- und Aggregationsbeziehungen werden durch die objektorientierte Dokumentation des CityGML-Formats in UML-Diagrammen beschrieben.

Umgekehrt gibt es für OpenDRIVE ein XML-Schema und die formale Dokumentation besteht lediglich aus einer einfachen, hierarchischen Beschreibung des Schemamodells. Die OpenDRIVE-Spezifikation stellt in einer Tabelle eine vereinfachte Übersicht über alle Elemente, die innerhalb eines OpenDRIVE-Dokuments auftreten können, dar. Dazu wird die maximale Anzahl eines gegebenen Elementtyps und die Hierarchie der einzelnen Elemente angegeben. Alle Elemente werden durch Ebenen organisiert. Primäre Elemente sind durch die 1. Ebene definiert. Kinderelemente der vorausgehenden Ebene werden mit einem Wert größer 0 beschrieben. Zusätzlich wird auf eine Kompositionsbeziehung zwischen den Elementen hingewiesen. Kinderelemente dürfen demnach nicht vorhanden sein, wenn die jeweiligen Elternelemente nicht verwendet werden [Dupuis et al., 2015].

Eine weiterreichende Beschreibung über die Beziehungen zwischen Objekten gibt es in der Dokumen-

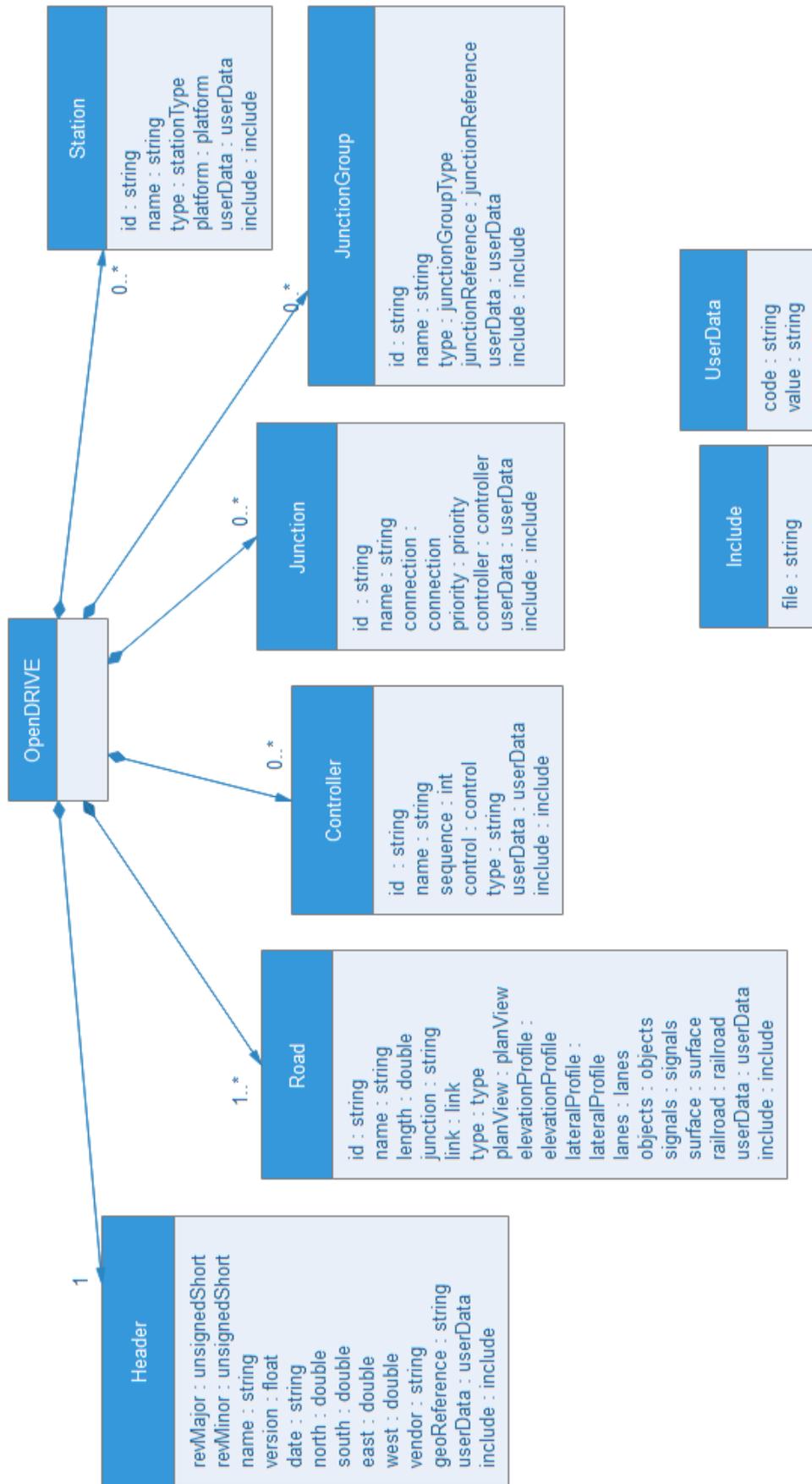


Abbildung 5.1 – Darstellung des OpenDRIVE-Schemas 1.4 als UML-Diagramm

tation allerdings nicht. An dieser Stelle wird deshalb und zur besseren Vergleichbarkeit mit der CityGML-Spezifikation aus der XML-Schemadatei von OpenDRIVE ein UML-Diagramm erzeugt (siehe Abb. 5.1). Die Verbindungen unter den Klassen werden als gerichtete Assoziationsbeziehungen beschrieben. Vererbungsstrukturen treten demnach in OpenDRIVE nicht auf. Die Kompositionsverbindungen, die in der Spezifikation definiert werden, werden ebenfalls durch die entsprechende Pfeilnotation sichtbar. Das Wurzelement der XML-Schemadatei wird durch die Klassen *Header*, *Road*, *Controller*, *Junction*, *JunctionGroup* und *Station* beschrieben, die als Elemente der ersten Ebene gelten. Lediglich die Klassen *Header* und *Road* müssen in einem OpenDRIVE-Dokument vorkommen, wobei letztere auch mehrmals auftreten kann. Während im Kopf des Dokuments die Georeferenzierung definiert wird, beinhaltet das *Road*-Element alle möglichen Eigenschaften einer Straße. Jedes Element kann mit benutzerdefinierten Daten erweitert werden. Diese Daten werden in einem sogenannten Zusatzelements mit dem Namen *userdata* gespeichert. Mithilfe des *include*-Elements an entsprechender Stelle können mehrere Dateien kombiniert werden [Dupuis et al., 2015]. Die definierten Elemente werden in Unterabschnitt 5.3.3 genauer beschrieben.

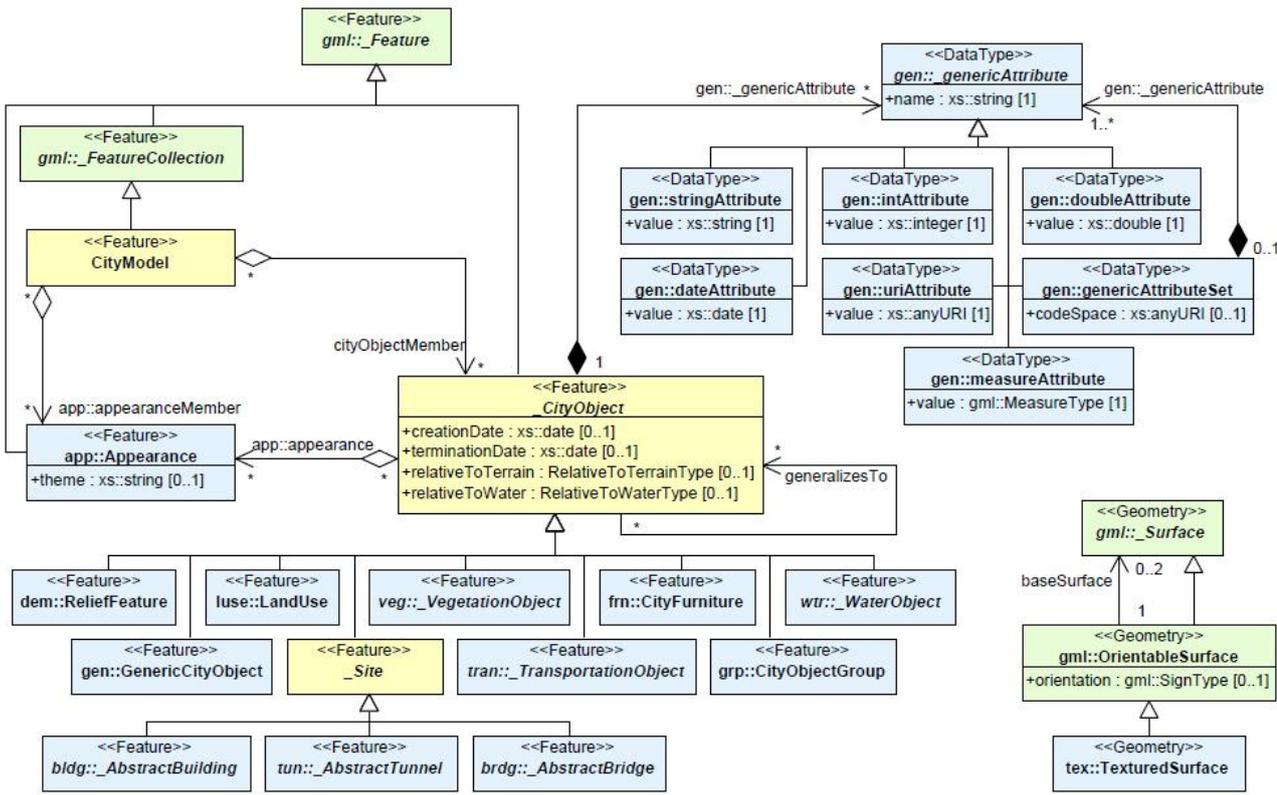


Abbildung 5.2 – Darstellung des UML-Diagramms des CityGML-Kernmoduls [Gröger et al., 2012, S. 51]

Zum Vergleich stellt Abb. 5.2 das UML-Schema des CityGML Kernmoduls dar. Die Basisklasse aller thematischen Klassen innerhalb des CityGML-Datenmodells ist die abstrakte Klasse *_CityObject*, welche eine Unterklasse der GML-Klasse *_Feature* darstellt. Alle Unterklassen und damit Erweiterungsmodule erben die in der Klasse *_CityObject* definierten Attribute. In CityGML wird insgesamt viel Wert auf

eine ausführliche Beschreibung und Dokumentation gelegt. Alle möglichen Klassen, Attribute und Beziehungen werden sehr detailliert festgehalten. Die Beschreibung eines neuen Stadtobjekts ist einfach, da alle Objekte von der Klasse *_CityObject* abgeleitet werden. CityGML differenziert die thematischen Erweiterungsmodule Appearance, Bridge, Building, CityFurniture, CityObjectGroup, Generics, LandUse, Relief, Transportation, Tunnel, Vegetation, WaterBody und TexturedSurface. Weitere Objekte, die dadurch nicht explizit modelliert werden können, können durch das Konzept der generischen Objekte und Attribute repräsentiert werden [Gröger et al., 2012]. Die Erweiterung von benutzerdefinierten Daten ist im Bereich des Straßenraums in OpenDRIVE möglich. Mithilfe des Objekttyps *building* können beispielsweise Gebäudegrundrisse ins Modell mit aufgenommen werden, eine weiterführende Darstellung und Erweiterung um ein neues Stadtobjekt ist jedoch nicht vorgesehen.

5.3 Unterschiede der Datenmodelle bezüglich des Straßenraums

OpenDRIVE und CityGML unterscheiden sich nicht nur im generellen Aufbau und der Modellstruktur. Auch die Art der Repräsentation des Straßenraums weicht voneinander ab. Der Aufbau des Straßenraums kann in mehreren Ebenen aufgeteilt werden. Chaplier et al. [2010] klassifizieren die notwendigen Informationen zur Beschreibung des Straßenraums in einen topologischen, einen logisch semantischen, einen physikalischen und einen visuellen Layer (vgl. Abschnitt 2.2). Im Folgenden werden die beiden Modellierungsstandards in Bezug auf diese vier Ebenen untersucht und verglichen.

5.3.1 Geometrie

In OpenDRIVE werden drei Koordinatensysteme verwendet, welche in Unterabschnitt 3.2.1 genauer erklärt werden. Der geographische Bezug muss in OpenDRIVE im Kopfelement *header* definiert werden. Es werden alle notwendigen Informationen zur Konvertierung der kartesischen OpenDRIVE-Koordinaten in ein entsprechendes geographisches Referenzsystem bereitgestellt:

```
<geoReference>
  <![CDATA[+proj=utm +zone=32 +ellps=WGS84 +datum=WGS84 +units=m +no_defs]]>
</geoReference>
```

CityGML verwendet zur geometrischen Beschreibung der Objekte absolute Weltkoordinaten. Das räumliche Referenzsystem wird als *srsName*-Attribut definiert:

```
<gml:Envelope srsName="EPSG:25832" srsDimension="3">
  <gml:lowerCorner>482922.2737 5729886.5465 107.591</gml:lowerCorner>
  <gml:upperCorner>483103.9661 5730080.3028 126.390799999994</gml:upperCorner>
</gml:Envelope>
```

Ein Geometrieobjekt erbt die Geometrie der Elternklasse oder wird durch die Definition des At-

tributs georeferenziert. Für jede Instanzdatei muss ein gültiges Referenzkoordinatensystem definiert sein.

Zur Beschreibung von Straßenverläufen haben sich geometrische Elemente wie Geraden, Kreisabschnitte und Klothoiden etabliert. Die Trassierung einer Straße erfolgt durch die Aneinanderreihung dieser Elemente. Auch in der OpenDRIVE-Formatspezifikation werden diese Elemente als Hauptbeschreibungselemente zur Beschreibung der Straßenreferenzlinie verwendet. Zur Visualisierung des Straßenraums wird in vielen Fällen allerdings die flächenhafte Beschreibung bevorzugt wie sie auch in CityGML implementiert ist. Die Darstellung enthält dann meistens zweidimensionale Polygone oder Volumenmodelle.

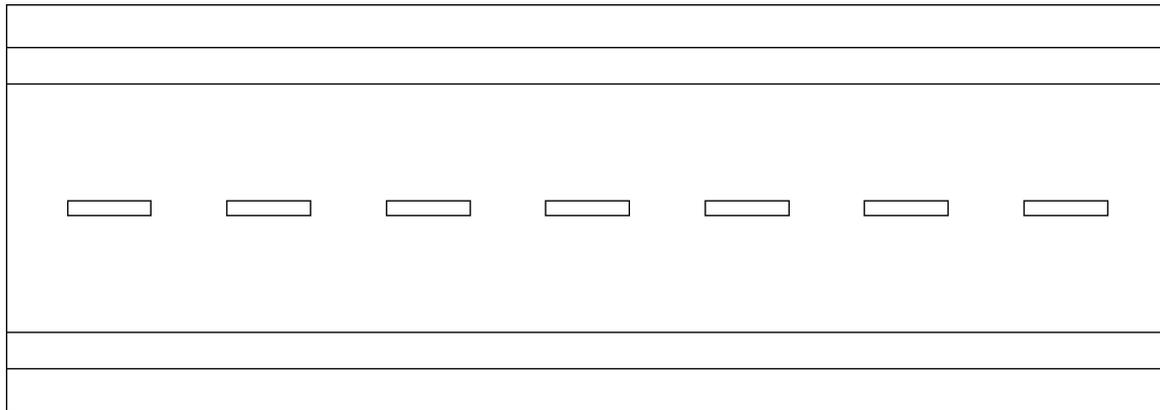


Abbildung 5.3 – Flächenhafte Beschreibung eines Straßenstücks durch Geometrieobjekte

Abb. 5.3 zeigt beispielhaft den Grundriss einer flächenhaften Beschreibung eines Straßenstücks durch einfache Geometrieobjekte.

Ein Unterschied der beiden Datenmodelle liegt demnach in der Darstellung. Während mit OpenDRIVE sozusagen eine linienhafte Darstellung mit linearer Referenzierung möglich ist, werden Geometrieobjekte in CityGML grundsätzlich als Flächen dargestellt. Eine linienhafte Visualisierung des Straßenraums ist in CityGML zwar möglich, eine lineare Referenzierung ist jedoch nicht implementiert.

Erlaubte Geometrietypen:

OpenDRIVE beschreibt auf analytischem Weg die Straßengeometrie durch Fahrspuren und Objekte entlang der Straße, die die Logik des Straßenverlaufs beeinflussen. Alle Straßen bestehen aus einer Referenzlinie, die die grundlegende Geometrie definiert. Die Geometrie der Referenzlinie wird als Sequenz von Primitiven unterschiedlicher Typen beschrieben. Zu den verfügbaren Primitiven gehören die Gerade mit einer konstanten Null-Krümmung, die Klothoide mit einem linearen Krümmungswechsel und die Kurve als konstante Nicht-Null-Krümmung entlang der Lauflänge. Zusätzlich können kubische Polynome und parametrische kubische Kurven verwendet werden. Durch die Referenzlinie wird demnach eine linienhafte Repräsentation des Straßenkörpers vorgenommen. Die Spurgeometrien werden mittels linearer Referenzierung mit Breitenangaben beschrieben. Die Breitenangaben werden durch polynomiale Funktionen 3. Grades mit der Formel $width = a + b*ds + c*ds^2 + d*ds^3$

definiert. Rechteckige oder kreisförmige Objekte und Signale werden durch die Beschreibung des Mittelpunktes im Trackkoordinatensystem beschrieben. Polygonförmige Objekte werden durch die Eckpunkte definiert.

CityGML verwendet zur Repräsentation von Geometrieobjekten die geometrischen Primitive der ISO 19107 mit der Einschränkung, dass Kurven nur linear und Flächen nur planar sein dürfen. Für einzelne Klassen sind verschiedene Geometrien und LoDs erlaubt. Die zur Beschreibung des Straßenraums notwendigen Objekttypen und die dazugehörigen Geometrietypen werden in Tabelle 5.1 dargestellt. Neben der Beschreibung des Straßenkörpers durch das Transportation Modul werden auch die Geometrien der Straßenraumobjekte Verkehrsschilder (CityFurniture Modul) oder Bäume (Vegetation Modul) aufgelistet.

Tabelle 5.1 – Mögliche LoDs und Geometrien zur Beschreibung des Straßenraums in CityGML

CityGML Objekttypen	Mögliche LoD und Geometrien	Geometrie
Transportation Complex	lod0Network	gml:GeometricComplex
	lod[1-4]MultiSurface	gml:MultiSurface
TrafficArea, AuxiliaryTrafficArea	lod[2-4]MultiSurface	gml:MultiSurface
CityFurniture	lod[1-4]Geometry	gml:_Geometry
	lod[1-4]TerrainIntersection	gml:MultiCurve
	lod[1-4]ImplicitRepresentation	core:ImplicitGeometry
Vegetation module	lod[1-4]Geometry	gml:_Geometry
	lod[1-4]ImplicitRepresentation	core:ImplicitGeometry

In der Hauptklasse TransportationComplex ist demnach auch eine linienhafte Repräsentation des Straßenraums durch die GML-Geometrie *gml:GeometricComplex* möglich. Sobald der Straßenraum detaillierter in TrafficAreas und AuxiliaryTrafficAreas unterteilt wird, wird jede Geometrie explizit als Flächenmodell durch die GML-Geometrie *gml:MultiSurface* dargestellt. Dreidimensionale Geometrien werden als Boundary Repräsentation (B-Rep) mit absoluten Koordinaten als Eckpunkte dargestellt.

Tabelle 5.2 – OpenDRIVE-Geometrietypen in CityGML und ISO 19107

Geometrietypen in OpenDRIVE	CityGML	ISO 19107
Gerade	+	+
Klothoide	-	+
Kreisbogen	-	+
Kubisches Polynom	-	+

Tabelle 5.2 zeigt, welche der in der OpenDRIVE-Spezifikation implementierten Geometrietypen auch in CityGML und in der ISO 19107 vorhanden sind. Alle OpenDRIVE-Geometrietypen werden auch durch die ISO 19107 unterstützt. Theoretisch könnten diese demnach auch für CityGML zugelassen werden. Der Nutzen einer Einführung der Geometriefunktionen ist allerdings fraglich, da die meisten GIS solche Geometrietypen bisher nicht unterstützen.

Im Folgenden wird eine Einteilung in Spurgeometrie und Objekt- bzw. Signalgeometrie unternommen, um die konkrete geometrische Darstellung der Objekte in den beiden Standards zu vergleichen. Diese Unterteilung basiert auf der Einteilung der OpenDRIVE-Formatspezifikation in Spuren, Objekte und Signale und dient an dieser Stelle der Übersicht und besseren Vergleichbarkeit der Standards.

Darstellung der Spurgeometrie:

In OpenDRIVE kann innerhalb eines Abschnitts jede Fahrspur durch mehrere Breiteneinträge beschrieben werden, wobei jede Spur durch mindestens einen Eintrag definiert sein muss. Jeder Breiteneintrag ist solange gültig bis ein neuer Eintrag definiert wird. Ein neuer Breiteneintrag ist jedes Mal erforderlich, wenn sich die mathematische Funktion des Polynoms wie in Abb. 5.4 dargestellt ändert.

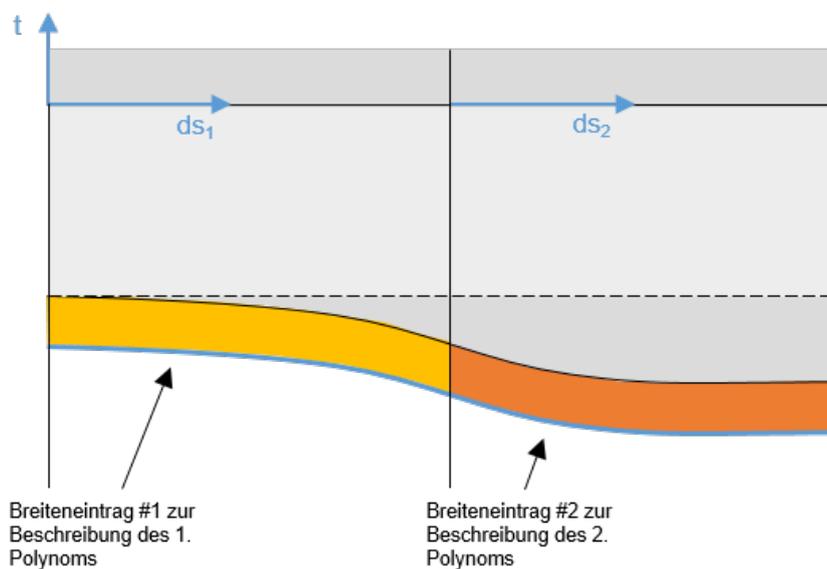


Abbildung 5.4 – OpenDRIVE: Darstellung der Spurgeometrie durch Breiteneinträge

In CityGML werden Spuren innerhalb der Klassen *TrafficArea* und *AuxiliaryTrafficArea* als *gml:MultiSurface* beschrieben. Eine *MultiSurface* besteht aus Polygonen, die jeweils durch einen planaren Ring begrenzt sind. Der planare Ring wird durch absolute Koordinaten der Eckpunkte beschrieben.

Ein direkter Vergleich der Darstellung eines Fahrbahnrandes wird anhand folgender Abbildungen (Abb. 5.5, Abb. 5.6, Abb. 5.7) vorgenommen. Dabei wird der in Fahrtrichtung rechte Rand verglichen und zwischen drei unterschiedlich modellierten Fahrbahnbegrenzungen unterschieden. Der Vergleich erfolgt ausschließlich auf ebenen planaren Polygonen, da eine andere Darstellung in CityGML nicht möglich ist. Im einfachsten Fall wird ein Straßenstück durch ein planares Rechteck mit geraden Begrenzungslinien, wie in Abb. 5.5 dargestellt, beschrieben. In OpenDRIVE wird die Fahrbahnbegrenzung durch mit einer polynomialen Funktion 3. Grades beschrieben. Im Vergleich dazu wird ein geradliniger Fahrbahnrand in CityGML durch die Eckpunkte charakterisiert.

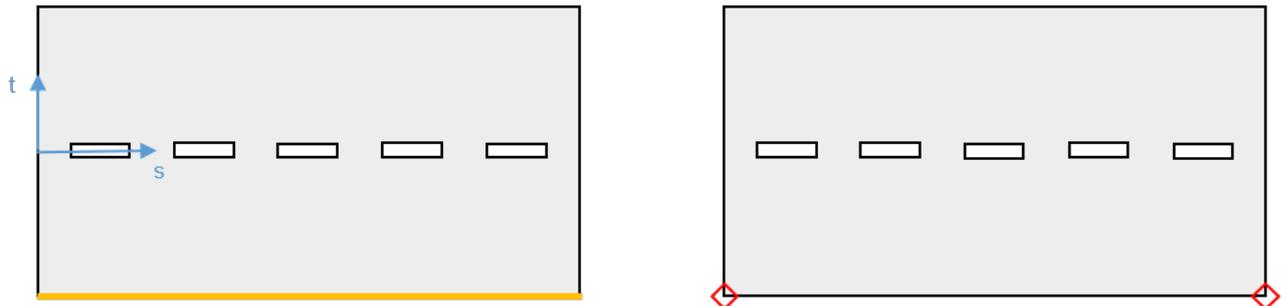


Abbildung 5.5 – Beschreibung des rechten Fahrbahnrandes eines Straßenstücks durch OpenDRIVE (links) und CityGML (rechts)

Die Beschreibung mit lediglich zwei Koordinateneinträge der Eckpunkte ist in diesem Fall leichter als die Beschreibung mit einer komplizierten mathematischen Formel für das gegebene Polynom. In den wenigstens Fällen ist eine Beschreibung durch eine einfach Gerade als Begrenzungsfläche jedoch ausreichend. Zur Beschreibung von Unregelmäßigkeiten am Fahrbahnrand oder von Kurvenstücken wird eine detailliertere Umrandung benötigt, wie in Abb. 5.6 dargestellt.

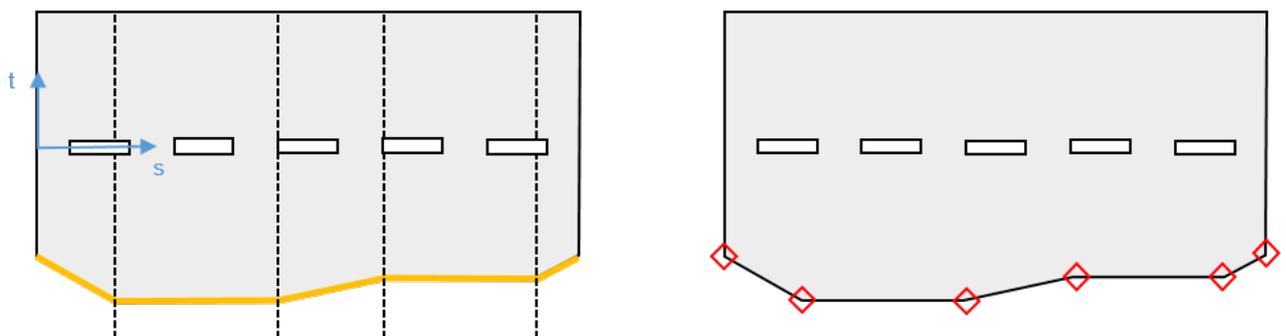


Abbildung 5.6 – Beschreibung eines Straßenstücks mit Unregelmäßigkeiten am rechten Fahrbahnrand durch OpenDRIVE (links) und CityGML (rechts)

In diesem Fall müssen in OpenDRIVE fünf polynomiale Funktionen, jeweils eine pro Geradenstück, beschrieben werden. Unkomplizierter ist hierbei die Beschreibung des Polygons in CityGML. Es werden lediglich weitere Koordinaten zur Beschreibung der Eckpunkte eingefügt.

Für eine wirklichkeitsgetreue Darstellung wäre eine Fläche mit polynomialen Begrenzungslinien nötig, da eine Straße in der Wirklichkeit nicht durch einzelne Stützpunkte beschrieben werden kann (vergleichbar Abb. 5.7). Die Beschreibung einer Fläche mit polynomialen Begrenzungslinie ist in CityGML grundsätzlich nicht möglich. Um eine solche Fläche annähernd zu beschreiben, müssten sehr viele Stützpunkte eingefügt werden. Die Beschreibung dieses Straßenrands geschieht in OpenDRIVE lediglich durch eine einzige Polynomfunktion 3. Grades.

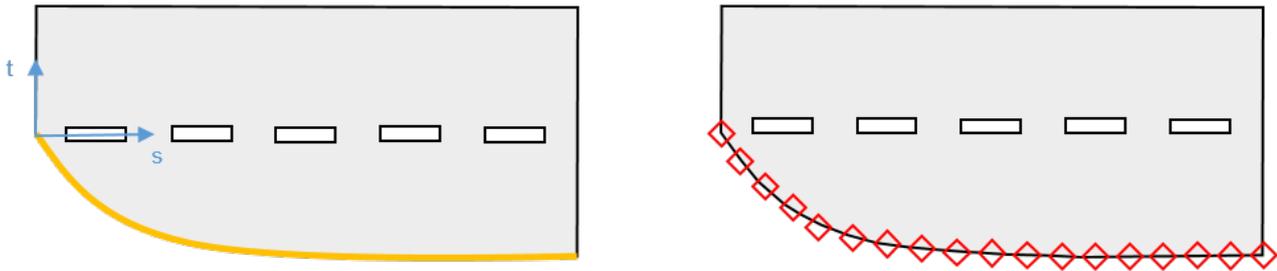


Abbildung 5.7 – Beschreibung des rechten Fahrbahnrandes eines Straßenstücks mit polynomialer Begrenzung durch OpenDRIVE (links) und CityGML (rechts)

Zusammenfassend ist die Beschreibung eines Straßenstücks in CityGML weniger mathematisch aufgebaut und demnach auch leichter verständlich. Unregelmäßigkeiten am Straßenrand können mit CityGML grundsätzlich leichter beschrieben werden. Für eine wirklichkeitsgetreue Darstellung des Straßenrands ist jedoch der OpenDRIVE-Standard besser geeignet.

Darstellung der Geometrie von Objekten und Signalen:

Zum Straßenraum gehört nicht nur der Straßenkörper selbst, sondern auch Objekte und Signale, die sich entlang der Straße befinden. In OpenDRIVE wird die Position von Objekten und Signalen mit Punktkoordinaten im Track-Koordinatensystem angegeben. Zusätzliche Attribute können zum Beispiel Länge, Breite, Radius, Höhe und Orientierung angeben, wodurch eine Beschreibung von linearen Objekten und Objekten mit rechteckigen oder kreisförmigen Grundflächen möglich ist.

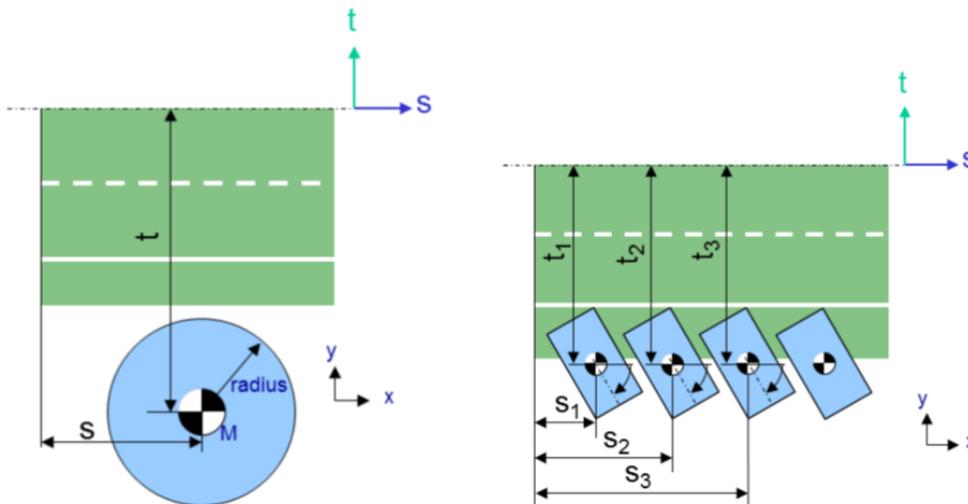


Abbildung 5.8 – Beschreibung von kreisförmigen oder zylindrischen Objekten (links), Beschreibung von Parkbuchten (rechts) ([Dupuis et al., 2015], S.66)

Polygonförmige Objekte können durch die Eckpunkte im Track-Koordinatensystem mit zusätzlicher Höheninformation beschrieben werden, wie in Abb. 5.9 dargestellt wird.

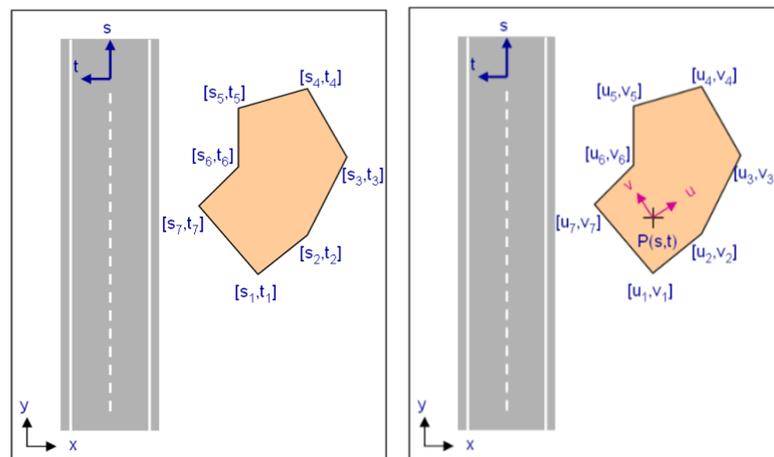


Abbildung 5.9 – Beschreibung von Polygonen im Track-Koordinatensystem (links) und im lokalen Koordinatensystem (rechts) ([Dupuis et al., 2015], S.68/69)

Auch in CityGML können Objekte entlang der Straße repräsentiert werden. Straßenmöblierungsobjekte sind unbewegliche Objekte wie Laternen, Ampeln, Verkehrszeichen, Litfaßsäulen, Bänke, Grenzpfosten oder Bushaltestellen und werden durch das Erweiterungsmodul *CityFurniture* repräsentiert. Einzelne Bäume werden durch *SolitaryVegetationObject* dargestellt. Die erlaubten Geometrien werden in Tabelle 5.1 aufgelistet. Straßenmöblierungsobjekte können durch alle möglichen GML-Geometrien beschrieben werden. Zusätzlich können erstere auch als *MultiCurve* beschrieben werden. Außerdem ist die Beschreibung durch implizite Geometrien in den LoD 1 bis 4 möglich.

5.3.2 Topologie

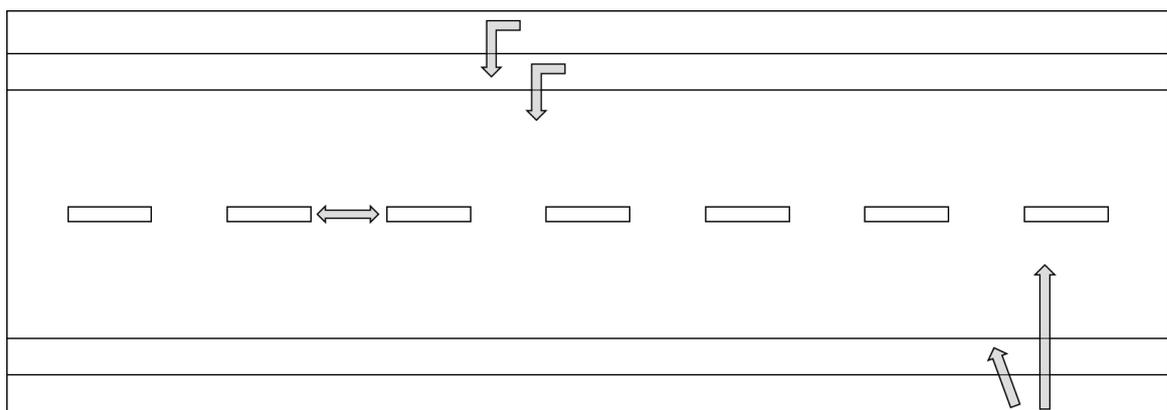


Abbildung 5.10 – Beschreibung eines Straßenstücks durch Geometrie und Topologie

Die Topologie beschreibt die Lagebeziehungen zwischen Geobjekten, wie in Abb. 5.10 vereinfacht dargestellt wird. In OpenDRIVE werden alle Spuren, Objekte und Signale durch eine Referenzlinie

linear referenziert. Aufeinanderfolgende Referenzlinien werden durch Angabe von Vorgänger und Nachfolger in Beziehung gesetzt:

```
<link> <predecessor id="-2" /> <successor id="-2" /> </link>
```

Für den Fall, dass eine Vorgänger-Nachfolger-Verbindung zu Mehrdeutigkeiten führt, werden Kreuzungen als Verbindungselemente eingeführt. Die Verbindungen zwischen den Bestandteilen können dadurch genau beschrieben werden. Die Topologie zwischen einzelnen Spuren wird ebenfalls durch Vorgänger- und Nachbarschaftsbeziehungen beschrieben. Zusätzlich können Nachbarschaftsbeziehungen zwischen Spuren definiert werden (vgl. Abschnitt 3.2.3.3). Objekte erhalten lediglich einen Verweis auf die zugehörige Referenzlinie und keine Angaben über den Spurtyp auf dem sie sich befinden.

CityGML erlaubt eine explizite Modellierung von Topologie. Dadurch ist beispielsweise eine gemeinsame Nutzung eines Geometrieobjekts durch mehrere Objekte oder Geometrien möglich und eine redundante Darstellung der gleichen Geometrie kann vermieden werden. Zur Implementierung topologischer Beziehungen wird das XML-Konzept der XLinks verwendet (vgl. Unterabschnitt 3.3.3). Dafür wird jedem Objekt eine eindeutige ID zugewiesen, die von anderen Objekten durch das Attribut *href* referenziert werden kann. Im folgenden Beispiel wird eine Wandfläche beschrieben als Polygon von zwei benachbarten Gebäudeteilen verwendet. Gebäudeteil 1 definiert die Wandfläche durch `<gml:Polygon gml:id="wallSurface1234">`. Das angrenzende Gebäudeteil 2 verweist dann mit `<gml:baseSurface xlink:href="#wallSurface1234"/>` auf die gemeinsame Wandfläche.

5.3.3 Semantik

Die semantische Beschreibung beinhaltet Informationen in Form von klassifizierten Objekten mit räumlichen und nicht-räumlichen Eigenschaften. Mithilfe von Klassen und Attributen können die Bedeutung eines Objekts, sowie physikalische und funktionale Eigenschaften repräsentiert werden, wie in Abb. 5.11) durch die Benennung der einzelnen Spurtypen dargestellt wird.

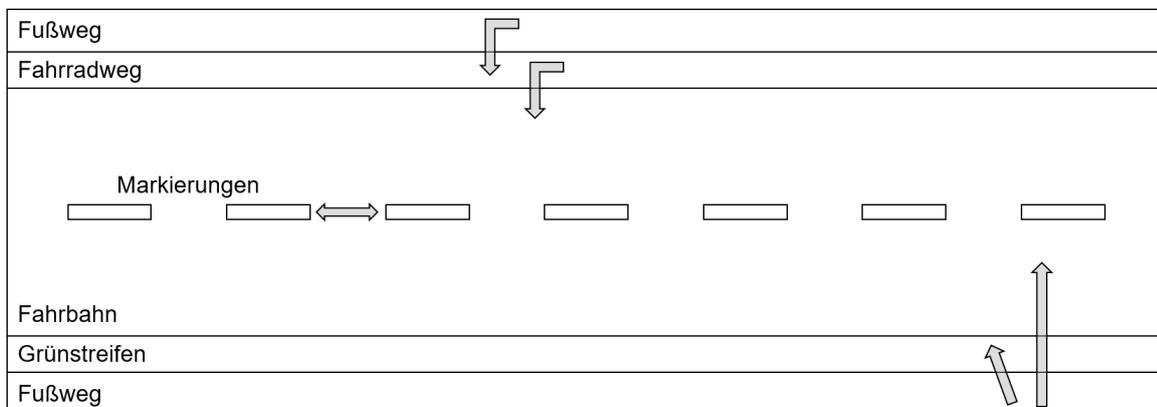


Abbildung 5.11 – Beschreibung eines Straßenstücks durch Geometrie, Topologie und Semantik

Das Wurzelement der OpenDRIVE-Schemadatei wird durch die primären Klassen *Header*, *Road*, *Controller*, *Junction*, *JunctionGroup* und *Station* beschrieben (vgl. Abb. 5.1). Ein *Road*-Element wird durch die Beschreibung der Geometrie im Grundriss (planView) und der Fahrspuren (lanes) definiert. Zusätzlich können die Topologie (link), der Straßentyp (type), das Höhen- und Querprofil (elevationProfile, lateralProfile) und die Oberfläche (surface) der Straße, sowie Objekte, Signale und Schienenwege (objects, signals, railroad) beschrieben werden (vgl. Abb. 5.12).

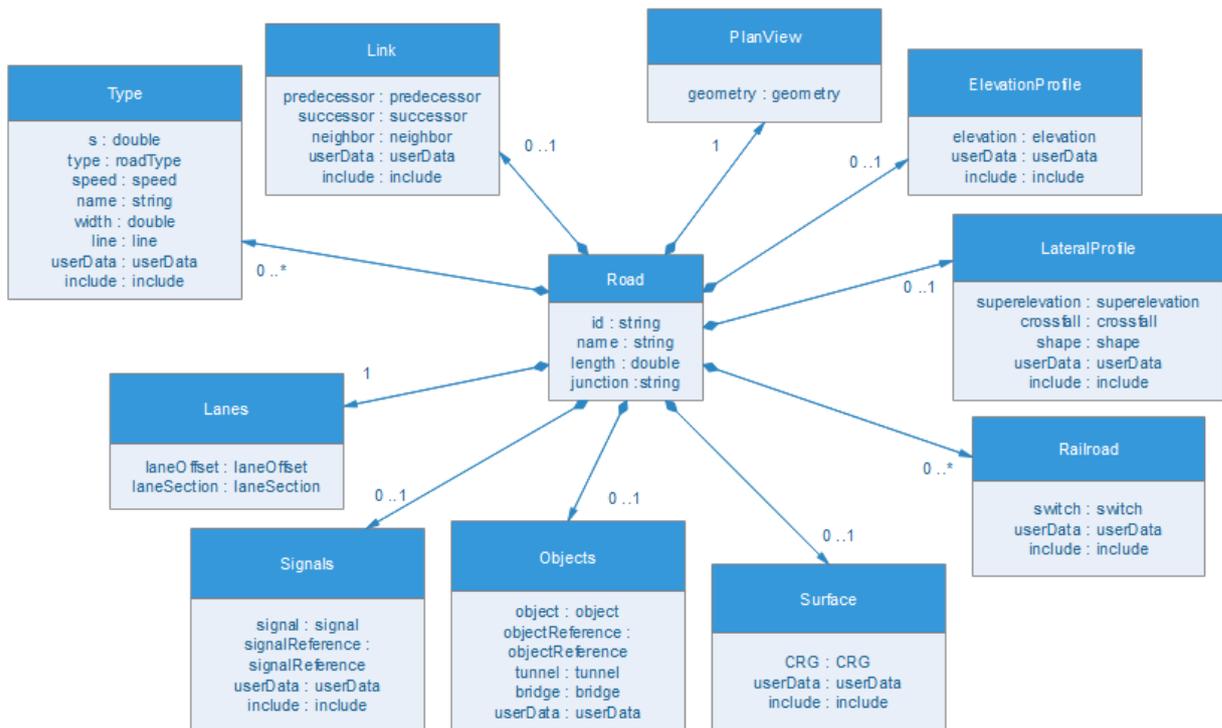


Abbildung 5.12 – Darstellung der Klasse Road des OpenDRIVE-Schemas 1.4 als UML-Diagramm

Die geometrische Beschreibung der einzelnen Elemente erfolgt durch lineare Referenzierung. Attribute definieren die semantischen Eigenschaften der Elemente. Objekte werden unter anderem durch die Attribute Typ, Name, ID, Orientierung zur Spurrichtung, Länge, Breite, Radius, Höhe und Rotationswerte beschrieben. Zur Definition von Signalen werden zusätzlich Attribute zur detaillierteren Beschreibung eingesetzt. Dadurch kann zum Beispiel der Text oder der Wert eines Verkehrsschildes abgebildet werden. Mithilfe der linearen Referenzierung ist die Wiederholung des gleichen Objekts leicht zu implementieren. Außerdem kann definiert werden für welche Spuren ein Objekt oder ein Signal gültig ist. Ein zusätzlicher Verweis zwischen Objekten wie zum Beispiel die Referenz von Ampeln auf Stoplinien oder die Abhängigkeiten zwischen Verkehrsschildern (vgl. Abb. 5.13) können definiert werden.

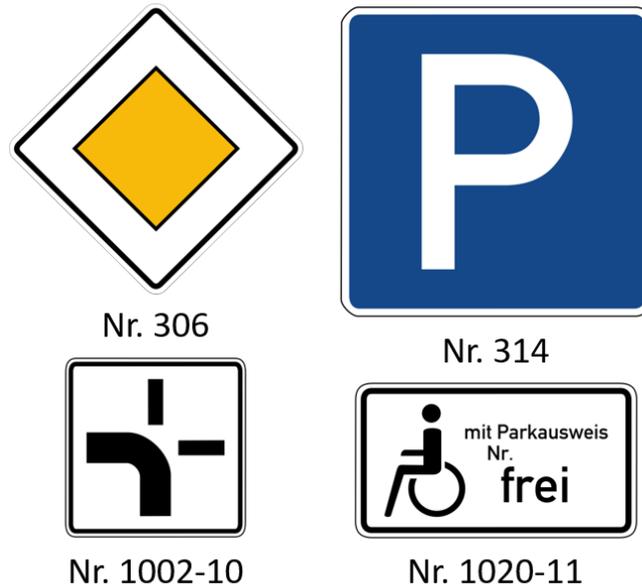


Abbildung 5.13 – Verweis zwischen Verkehrszeichen

Die jeweiligen Elementtypen werden in der Spezifikation durch feste Werte angegeben. So kann der Straßentyp beispielsweise als *town* oder *bicycle*, der Spurtyp als *driving* oder *sidewalk* und Objekte als *pole* oder *tree* beschrieben werden. Auch werden konstante Werte für den Typ und die Farbe einer Straßenmarkierung angegeben. Alle möglichen Typen können in der Spezifikation nachgelesen werden. Tabelle 4.1 und Tabelle 4.2 geben einen Überblick über die implementierten Spur- und Objekttypen. Die Auflistung ist allerdings die einzige Beschreibung dazu. Grundsätzlich definiert die Spezifikation feste Objekttypen mit möglichen Attributen. Trotzdem lässt sie an dieser Stelle Interpretationsspielraum, welche Objekte damit konkret beschrieben werden sollen.

Die Basisklasse aller thematischen Klassen innerhalb des CityGML-Datenmodells ist die abstrakte Klasse *_CityObject*, welche eine Unterklasse der GML-Klasse *_Feature* darstellt. Für jedes Erweiterungsmodul werden zur genauen Beschreibung der Objekte Codelisten definiert. Diese enthalten Werte der Attribute *class*, *function* und *usage*. Einige Erweiterungsmodul wie das Transportation Modul erlauben zusätzliche Attribute. Dadurch kann beispielsweise in den Klassen *TrafficArea* und *AuxiliaryTrafficArea* mithilfe des Attributs *surfaceMaterial* die Straßenoberfläche beschrieben werden. Weitere Objekte, die dadurch nicht explizit modelliert werden können, können durch das Konzept der generischen Objekte und Attribute repräsentiert werden. Das Generische Modul stellt Erweiterungen des CityGML Datenmodells bereit und erweitert demnach die abstrakte Basisklasse *_CityObject*. Darüber hinaus können anwendungsspezifische Erweiterungen für das CityGML Datenmodell durch die Verwendung von Application Domain Extensions (ADE) realisiert werden.

CityGML unterstützt mehr Stadtobjektarten als OpenDRIVE und lässt auch im Bereich des Straßenraums eine ähnliche feingliedriger Unterteilungen einer Straße zu. Die Beschreibung von Kreuzungsbereichen wird in OpenDRIVE genauer definiert. Alle Informationen über Objekte werden in OpenDRIVE als genau definierte Attributeinträge festgehalten. Außerdem gibt die Spezifikation Listen mit festen Einträgen für die Spurtypen, Objekttypen und Signaltypen vor. Im Bereich des Straßenraums

sind die Angaben in der OpenDRIVE-Spezifikation stark standardisiert, erfüllen jedoch den Zweck der vollständigen Beschreibung einer Straße wie sie für Simulationsanwendungen notwendig ist. Die definierten Werte zur Beschreibung eines Straßenraums können in OpenDRIVE durch UserData-Einträgen und in CityGML mittels generischer Attribute erweitert werden.

5.3.4 Visuelle Erscheinung

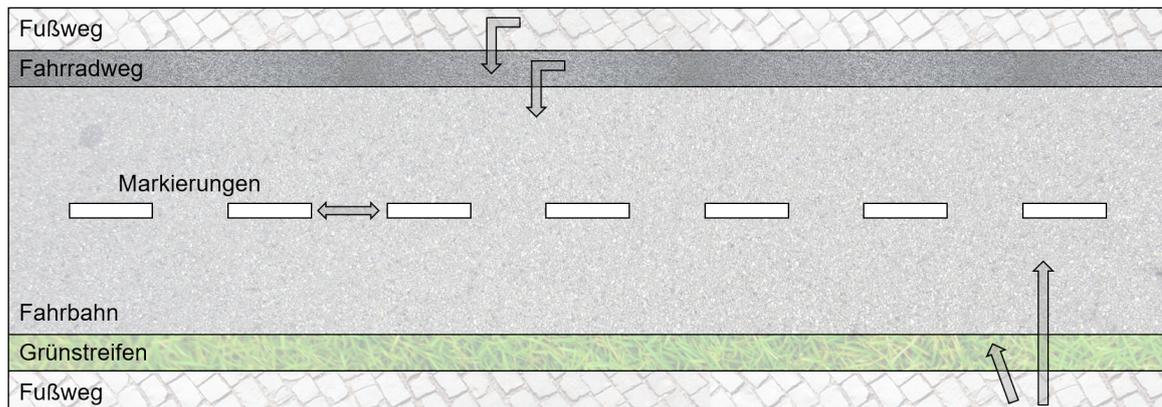


Abbildung 5.14 – Beschreibung eines Straßenstücks durch Geometrie, Topologie, Semantik und visuelle Erscheinung

In CityGML wird außerdem großer Wert auf die visuelle Erscheinung der 3D-Stadtmodelle gelegt. Dafür wird in der Spezifikation das eigenständige *Appearance Modul* definiert. Durch die Definition von objektspezifischen Farben und Texturen können sichtbare Eigenschaften der Objektoberfläche repräsentiert werden. Die Anzahl der Appearances für ein Geometrieobjekt ist nicht begrenzt, wodurch ein Objekt in CityGML verschiedene Erscheinungsformen besitzen kann. Die Themen werden nur durch eine eindeutige ID repräsentiert. Demnach können in CityGML verschiedene Themen für ein Objekt definiert werden. So ist beispielsweise eine texturierte Darstellung eines Objekts mit den unterschiedlichen Themen Sommer, Winter, Tag oder Nacht möglich. Außerdem können durch die Beschreibung eines X3D-Materials die Farbe und die Reflektionseigenschaften einer Oberflächegeometrie definiert werden [Gröger et al., 2012].

Die Erscheinung der Straßenoberfläche wird in OpenDRIVE lediglich semantisch durch Attribute beschrieben. Verschiedene Attributeinträge definieren das Material einer Fahrspur. Das Oberflächenmaterial einer Fahrspur und eines Objekts kann durch einen Materialcode, die Rauigkeit der Oberfläche und einen Wert, der die Reibung definiert, beschrieben werden. Das Erscheinungsbild von Straßenmarkierungen wird durch die Attribute *color* und *material* beschrieben [Dupuis et al., 2010].

Am Merkmal der visuellen Erscheinung erkennt man die unterschiedlichen Anwendungsfälle der beiden Spezifikationen. Das CityGML-Datenmodell ist für eine detailreiche visuelle Darstellung ausgelegt. Eine Unterscheidung in Themen ist in OpenDRIVE nicht implementiert, da die visuelle Erscheinung bei Simulationsanwendungen neben der Geometrie und Topologie zweitrangig ist. Die Materialbeschreibung geht nicht über die Definition einzelner Attributwerte hinaus.

5.4 Semantische Zuordnung

Abschließend an den Vergleich der zwei Datenmodelle wird schematisch ein mögliche Abbildung der einzelnen Klassen und Attribute dargestellt. An dieser Stelle wird lediglich auf die in dieser Arbeit relevanten Elemente des Straßenraums eingegangen. Ein Vergleich der Stadtobjekte wie Brücke, Tunnel und Zug, welche sowohl in OpenDRIVE als auch in CityGML implementiert sind, werden nicht näher beschrieben.

In den Spezifikationen werden konkrete Werte für einige Attribute angegeben, die an dieser Stelle dem semantischen Vergleich dienen. Während die Beschreibung der Straßenart in OpenDRIVE durch das Attribut *type* der Klasse Road beschrieben wird, wird in CityGML die Klasse TransportationComplex mit dem Attribut *class* verwendet (vgl. Tabelle 5.3). Die Spurtypen in OpenDRIVE, die durch das *type*-Attribut der Klasse Lane implementiert werden, werden mithilfe der CityGML-Attribute *function* und *usage* in den Klassen TrafficArea und AuxiliaryTrafficArea gespeichert. Zur Beschreibung von Straßenobjekten werden in OpenDRIVE die Klassen *object* und *signal* definiert. Vergleichbar dazu müssen in CityGML die Klassen CityFurniture und SolitaryVegetationObjekt mit den Attributen *function* und *usage* herangezogen werden.

Tabelle 5.3 – Darstellung der OpenDRIVE-Klassen mit vergleichbaren CityGML-Klassen

OpenDRIVE Klasse	Attribut	CityGML Klasse	Attribut
Road	type	TransportationComplex	class
Lane	type	TrafficArea	function, usage
		AuxiliaryTrafficArea	function, usage
Object	type	CityFurniture	function, usage
		SolitaryVegetationObject	function, usage
Signal	type	CityFurniture	function, usage

Für die Gegenüberstellung werden zunächst alle möglichen Spurtypen der Klasse *Lane* der OpenDRIVE-Spezifikation mit passenden Attributwerten der CityGML-Klassen *TransportationComplex*, *TrafficArea* und *AuxiliaryTrafficArea* aufgelistet. Eine mögliche Zuordnung zeigt Tabelle 5.4. In OpenDRIVE werden Fahrspuren durch das *type*-Attribut beschrieben. Vergleichbar wird in CityGML das Attribut *function* verwendet. Die Betrachtung erfolgt grundsätzlich von Elementen des OpenDRIVE-Datenmodells zu Elementen des CityGML-Modells. Die möglichen Elemente zur Beschreibung der Straße sind im CityGML-Datenmodell sehr umfangreich, wodurch beinahe alle in OpenDRIVE implementierten Spurtypen auf CityGML-Attribute abgebildet werden können. Eine eindeutige Abbildung aller CityGML-Spurtypen in OpenDRIVE ist nicht gegeben und wird an dieser Stelle auch nicht weiter untersucht. Fahrspuren wie Trambahnschienen werden in CityGML nicht explizit durch das *function*-Attribut beschrieben, können aber durch das zusätzliche Attribut *usage* mit dem Attributnamen 8 für Tram eindeutig charakterisiert werden. Die Attributnamen *none*, *special1*, *special2* und *special3* in OpenDRIVE lassen Spielraum für unbekannte oder weitere Spurtypen. In Tabelle 5.4 wurden zusätzlich Grünflächen und Rinnsteine

mitaufgenommen. Diese sind zwar beide noch nicht als Spurtyp in OpenDRIVE implementiert, sind aber für eine vollständige Erfassung des Straßenraums notwendig.

Ein Vergleich der Straßenraumobjekte kann grundsätzlich zwischen den OpenDRIVE-Klassen *Object* und *Signals* und der CityGML-Klasse *CityFurnitureObject* erfolgen. Zusätzlich muss die CityGML-Klasse *SolitaryVegetationObject* für die Abbildung von Vegetation wie Bäumen und Büschen, die in OpenDRIVE der Klasse *Object* untergeordnet sind, herangezogen werden. Die CityGML-Spezifikation definiert auch hier viel umfangreichere und detailliertere Codelisten, was die Zuordnung der OpenDRIVE-Objekttypen deutlich erschwert. Zum einen werden im OpenDRIVE-Standard Objekttypen wie *car*, *truck*, *van* oder *bus* angegeben, die in den vergleichbaren CityGML-Modulen nicht vorhanden sind oder die ohne weitere Objektbeschreibung nicht eindeutig zuzuordnen sind. Zum anderen werden Gebäude in OpenDRIVE lediglich durch den Objekttyp *building* beschrieben, während CityGML ein komplettes Erweiterungsmodul zur Beschreibung von Gebäuden vorsieht. Eine relativ eindeutige Zuordnung der OpenDRIVE-Objekttypen *pole* und *tree* ist möglich, obwohl CityGML auch in diesem Fall viel detaillierter zwischen Pfostenarten und Baumarten unterscheidet. Außerdem lassen die Typen *none* und *obstacle* Spielraum in der Beschreibung von unbekanntem oder nicht implementierten Objekten.

Die Beschreibung der Straßenoberfläche kann in beiden Datenmodellen mit einem zusätzlichen Attribut erfolgen, dessen Werte in den beiden Standards vergleichbar sind. Weitere Objektattribute sind im Vergleich dazu in OpenDRIVE detaillierter beschrieben als in CityGML. Ein Objekt wird in OpenDRIVE neben dem Typ durch weitere festgelegte Attribute wie Länge, Breite, Höhe, Radius und Rotation charakterisiert. Während mögliche Attribute in der OpenDRIVE-Spezifikation vollständig aufgelistet werden, verzichtet CityGML in den meisten Modulen auf eine genaue Festlegung und lässt durch Generische Attribute Interpretationsspielraum. Dieses Merkmal lässt sich auf die unterschiedlichen Anwendungsbereiche der beiden Standards zurückführen. Für Simulationsanwendungen sind präzise Wertangaben zur vollständigen Beschreibung von Straßen und Objekten notwendig. Weiterführende Beschreibungen wie unterschiedliche Baumarten sind dabei erstmal zweitrangig.

Der Vergleich gibt einen Überblick über gleiche oder ähnliche Klassen und Attribute der beiden Datenmodelle. Alle weiteren in OpenDRIVE implementierten Attribute müssen in CityGML durch Generische Attribute abgebildet werden. Lediglich die Klasse *SolitaryVegetationObject* bietet für die Darstellung der Objekte weitere Attribute. Für die spätere Umwandlung der für OpenDRIVE aufbereiteten MMS-Daten in das CityGML-Format dient diese Übersicht als Grundlage.

Tabelle 5.4 – Betrachtung möglicher OpenDRIVE-Spurtypen mit passenden CityGML-Attributen

Beschreibung	OpenDRIVE		Attributname	CityGML		Beschreibung
	Klasse	Attributname		Klasse	Attributname	
Fahrspur	Road	driving	TrafficArea	1	driving_lane	
Gehweg		sidewalk		2	footpath	
Fahrradweg		biking		3	cyclepath	
Zugspur		rail		8	rail	
Tramspur		tram		-	-	
Auffahrt an einer Autobahn		entry		34	motorway_entry	
Abfahrt an einer Autobahn		exit		35	motorway_exit	
unbekannt		none		9999	unknown	
befestigte Seitenstreifen		shoulder	AuxiliaryTrafficArea	1010	hard shoulder	
<i>Grünstreifen</i>	-	-		1020	<i>green area</i>	
mittlerer Trennstreifen		median		1030	middle lane	
Standspur		stop		1040	lay by	
Parkfläche		parking		1100	parking bay	
<i>Rinnsstein</i>	-	-		1210	<i>drainage</i>	
Randspur		border		1220	kerbstone	
eingeschränkte Flächen		restricted		-	-	
Bauarbeiten auf der Spur		roadWorks		-	-	
bidirektionale Spur		bidirectional		-	-	
Verbindungsrampe bei Autobahnen		offRamp		-	-	
Verbindungsrampe bei Autobahnen		onRamp		-	-	
-	-	special1		-	-	
-	-	special2		-	-	
-	-	special3		-	-	

6 Konzeptentwicklung zur Visualisierung von Mobile Mapping Daten

In Kapitel 4 wurde ein umfassender Überblick über den Erfassungsprozess von Mobile Mapping Daten im Allgemeinen gegeben und im Speziellen auf den Prozess der Straßenraummodellierung bei 3D Mapping Solutions eingegangen. Aus den gewonnenen Erkenntnissen über den Ablauf soll in diesem Kapitel ein Konzept zur Visualisierung der kinematisch erfassten Vermessungsdaten des Unternehmens entwickelt werden. Zusätzlich fließen die zuvor erarbeiteten Grundlagen der Modellierungsstandards OpenDRIVE und CityGML in die Überlegungen mit ein. Im ersten Abschnitt dieses Kapitels werden mögliche Wege der Weiterverarbeitung im Prozessablauf diskutiert und daraus im zweiten Abschnitt die tatsächliche Umsetzung abgeleitet.

6.1 Mögliche Wege der Weiterverarbeitung

Die hochgenauen Vermessungsdaten von 3D Mapping Solutions werden als Referenzinformationen und Referenzkarten im OpenDRIVE-Standard bereitgestellt. Eine Visualisierung der Daten ist momentan mithilfe des ODR-Viewer Scenario Editors möglich. Außerdem können die Referenzlinien als linienhafte Streckenbeschreibung in das KML-Format konvertiert werden. Eine realitätsnahe Visualisierung, welche für die Präsentation der MMS-Daten und bei der Kundenakquise vorteilhaft wäre, ist durch diese beiden Möglichkeiten nicht gegeben. Auf Grund dessen werden in diesem Abschnitt drei mögliche Wege zur Visualisierung der kinematisch erfassten Straßendaten vorgestellt. Dafür wird in Abb. 6.1 nochmals der im Abschnitt 4.2 dargestellte Erfassungsprozess bei 3D Mapping Solutions aufgegriffen.

Umwandlung des OpenDRIVE-Formats in ein Computergrafikformat

Die naheliegendste Lösung wird in Abbildung 6.1 dargestellt und liegt in der Umwandlung des Endprodukts im OpenDRIVE-Format in ein Computergrafikformat. Möglicherweise gehen jedoch in diesem Schritt die umfangreichen Sachdaten verloren. Einen Ausweg bietet die Verwendung des KML-Formats bei dem die Sachdaten teilweise bestehen bleiben können, da KML es erlaubt zu jeder Geometrie eine große Anzahl von Attributen zu speichern. Daten im KML-Format können außerdem von vielen Programmen wie ArcGlobe oder GoogleEarth visualisiert werden, wodurch sich ein weiterer Vorteil aus der Verwendung des KML-Formats als Computergrafikformat ergibt.

Die Beziehungen zwischen den Straßenraumobjekten, die vor allem in OpenDRIVE eine wichtige

Rolle spielen, können mit KML allerdings nicht umgesetzt werden. So würde beispielsweise die Information zu welcher Straße eine Ampel gehört verloren gehen. Auch weitere Probleme können bei der Umwandlung auftreten. Im ersten Schritt muss eine allgemeine Regel für die Handhabung der in OpenDRIVE verwendeten Spline-Geometrien gefunden werden. Die Umwandlung der linienhaften in flächenhafte Geometrietypen bleibt jedoch auch dann nicht trivial. Außerdem werden für die eigentliche Visualisierung zusätzliche Software-Pakete wie ArcGlobe oder Google Earth benötigt. Für eine effiziente Visualisierung großer Datenmengen ist eine räumliche Kachelung notwendig, die in diesem Ansatz jedoch nicht unterstützt wird. Eine zusätzliche manuelle Aufteilung in kleinere Gebiete ist daher notwendig. Von entscheidendem Vorteil ist hierbei trotzdem die gute Unterstützung durch GoogleEarth und ArcGlobe zu nennen, die durch automatisches Nachladen bestehende räumliche Kachelungen berücksichtigen.

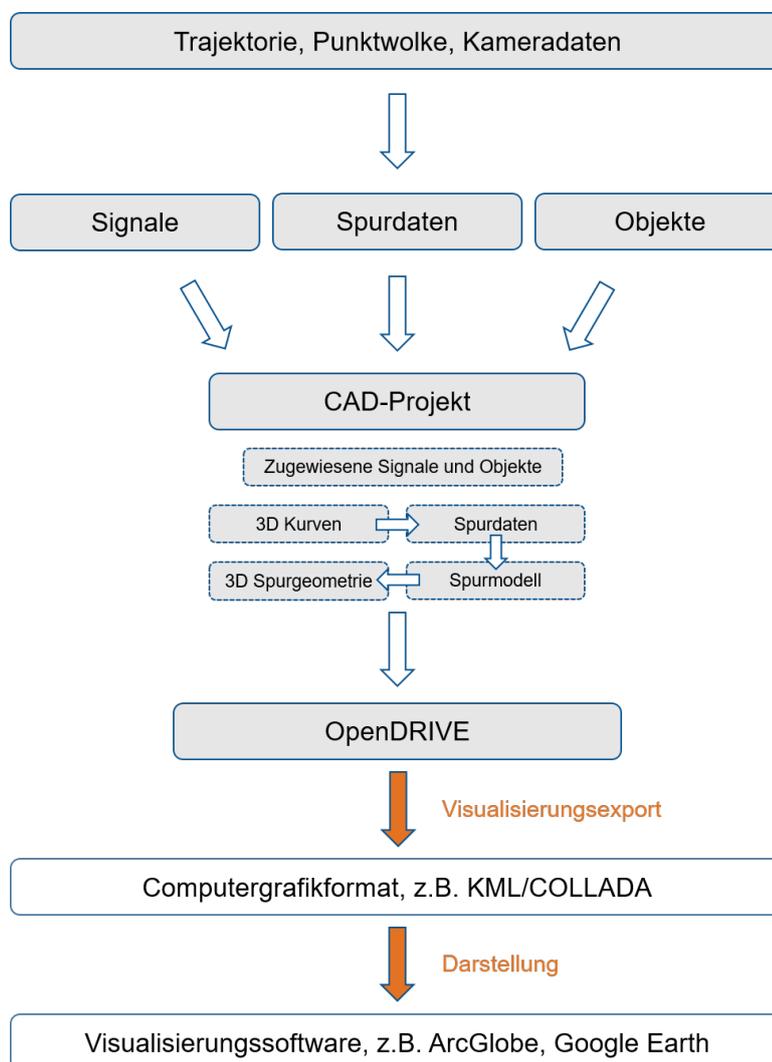


Abbildung 6.1 – Visualisierung der Daten durch Umwandlung des OpenDRIVE-Formats in ein Computergrafikformat

Konvertierung des OpenDRIVE-Formats in das CityGML-Format

Eine Alternative zur vorgestellten Konvertierung in das KML-Format mit anschließender Visualisierung, stellt die Transformation nach CityGML als zusätzlicher Schritt zwischen OpenDRIVE und KML dar, welche in Abb. 6.2 dargestellt wird. Der Vorteil an dieser Variante liegt unter anderem darin, dass für CityGML Werkzeuge zur automatischen Kachelung vorhanden sind. Für eine effiziente Visualisierung können so große Datenmengen in kleinere Teilbereiche gekachelt werden. Die bestehende Problematik der notwendigen Umwandlung der geometrischen Repräsentation kann dadurch jedoch nicht umgangen werden. Die Umwandlung von OpenDRIVE zu CityGML durch einen Konverter ist daher schon aufgrund der unterschiedlich umgesetzten Implementierung der Geometrie herausfordernd, wie auch aus Unterabschnitt 5.3.1 hervorgeht.

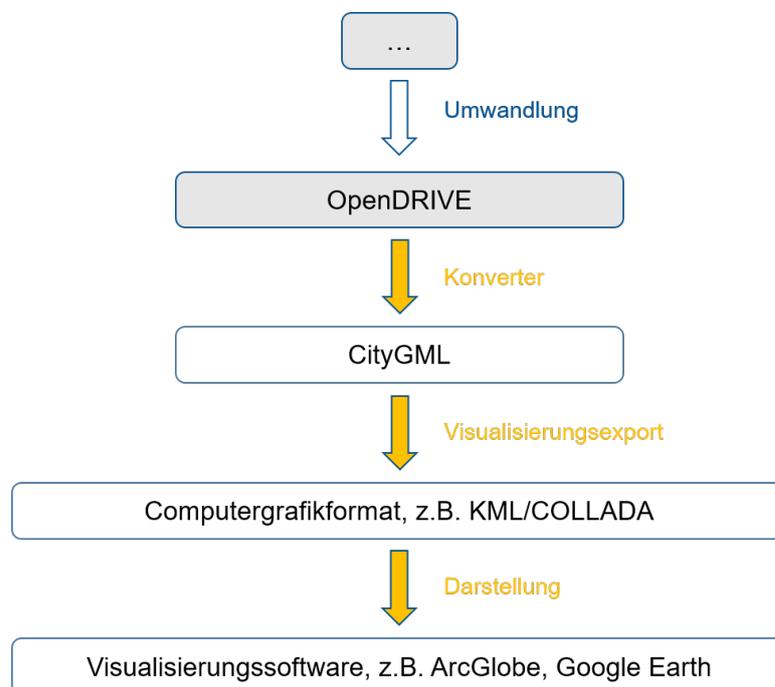


Abbildung 6.2 – Visualisierung der Daten durch Konvertierung des OpenDRIVE-Formats in das CityGML-Format

Konvertierung der Zwischenprodukte in das CityGML-Format

Um Probleme bei der Konvertierung von OpenDRIVE zu CityGML zu umgehen, kann zur Visualisierung der Daten auf Zwischenprodukte, die im Erfassungsprozess von 3D Mapping Solutions entstehen, zurückgegriffen werden. Vorteilhaft ist hierbei die vereinfachte Konvertierung der Geometrie, da die Geometrien an dieser Stelle noch flächenhaft vorliegen und daher dem gewünschten Format ähnlicher sind. Der Umweg über das OpenDRIVE-Format und den damit verbundenen Schwierigkeiten bei der Umwandlung der Geometrie kann so umgangen werden, wie aus Abbildung 6.3 deutlich wird.

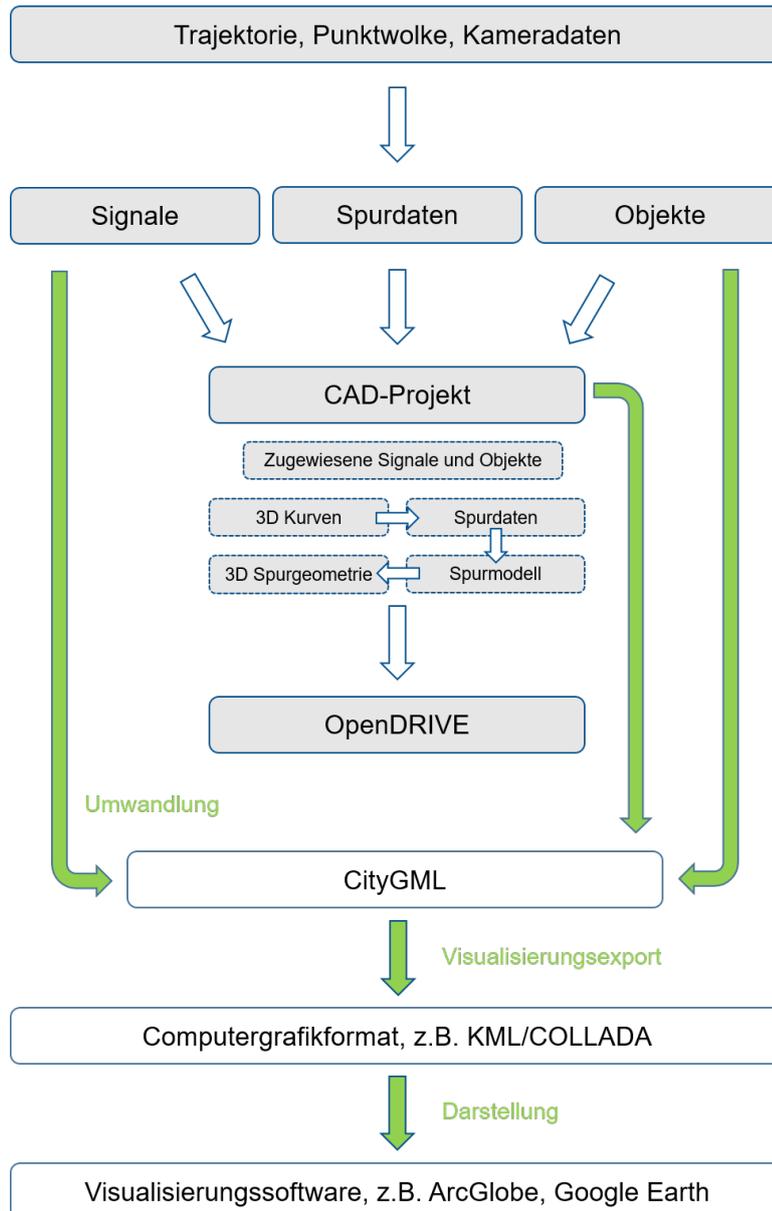


Abbildung 6.3 – Visualisierung durch Konvertierung der Zwischenprodukte in das CityGML-Format

6.2 Tatsächliche Umsetzung

Wie aus der vorangegangenen Analyse hervor geht, eignet sich insbesondere die letzte der genannten Varianten zur praktischen Umsetzung. Dies greift den in Abschnitt 1.3 beschriebenen Visualisierungsweg von Straßendaten auf, der auch von Beil [2017] umgesetzt wurde. Neben dem OpenDRIVE-Dokument ergeben sich aus der Visualisierung zwei neue Produkte, entsprechend gekennzeichnet in Abb. 6.4. Vergleicht man die ersten beiden beschriebenen Varianten wird der Mehrwert des Weges über CityGML klar. Neben OpenDRIVE wird ein umfangreiches CityGML-Dokument erstellt. Dies beinhaltet alle Objekte der Straße und Objekte entlang der Straße. Zusätzlich

wird ein KML-Dokument zur Visualisierung erzeugt. Dieser Mehrwert verändert sich zwischen den letzten beiden Varianten zwar nicht, jedoch können Probleme der geometrischen Umwandlung reduziert bzw. vermieden werden, wenn statt des OpenDRIVE-Dokuments Daten aus früheren Erfassungsprozessstufen genutzt werden.

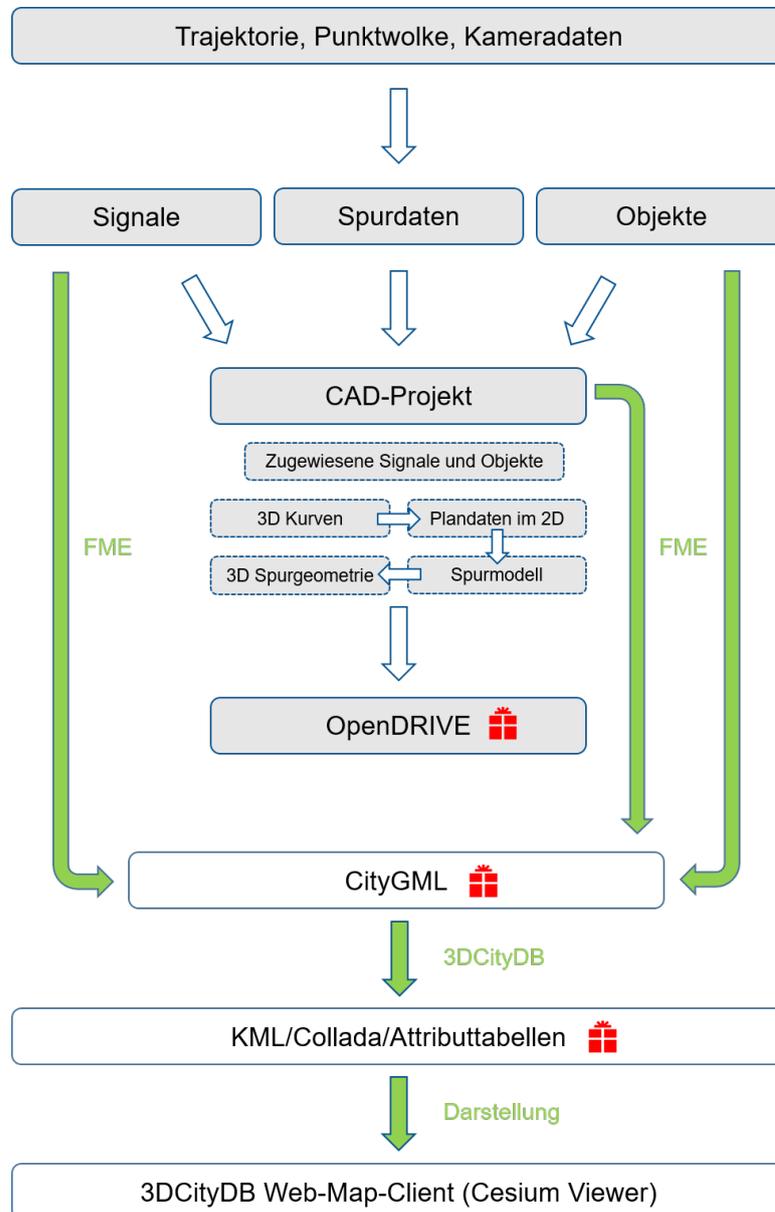


Abbildung 6.4 – Visualisierung der MMS-Daten über die Konvertierung in das CityGML-Format und Darstellung durch KML im Web-Map-Client der 3DCityDB

Die Daten werden in der folgenden Umsetzung hauptsächlich aus dem CAD-Projekt, welche dort bereits für das OpenDRIVE-Dokument aufbereitet wurden, entnommen. In einigen Fällen kann es sinnvoll sein, weitere Dateien aus dem Extraktionsprozess der Rohdaten zu verwenden. Abb. 6.4 zeigt den praktisch umgesetzten Weg mit den verwendeten Software-Paketten. Alle eingesetzten Softwareprodukte werden in Abschnitt 7.1 detailliert vorgestellt. Die Verarbeitung der Daten zu einem detaillierten

Straßenraummodell und die Umwandlung nach CityGML erfolgt in Feature Manipulation Engine (FME). Einzelne modellierte Straßenraumobjekte können bereits in FME visuell überprüft werden. Die erzeugten CityGML-Dateien werden in die 3DCityDB eingelesen, dort verwaltet und als KML-Dateien exportiert. Für die Anreicherung der geometrischen Objekte mit Attributen werden zusätzlich Tabellen generiert. Demnach wird die Visualisierung der durch Mobile Mapping aufgenommenen Daten über die Konvertierung in das CityGML-Format mit anschließender Darstellung durch KML im Web-Map-Client der 3DCityDB erreicht.

7 Erweiterung der Auswerteprozesse zur Straßenraummodellierung

Dieses Kapitel beschreibt das konkrete Vorgehen der Weiterverarbeitung und Visualisierung der für den OpenDRIVE-Standard aufbereiteten Mobile Mapping Daten des Unternehmens 3D Mapping Solutions. Im vorhergehenden Kapitel 6 wurde ein Konzept zur Erweiterung des Erfassungsprozesses erarbeitet. Abb. 7.1 zeigt ein vereinfachtes Ablaufdiagramm der Datenbearbeitung mit den verwendeten Software-Paketen. Diese werden zu Beginn dieses Kapitels beschrieben. Danach erfolgt die genaue Beschreibung der Modellierung, Umwandlung und Darstellung der Daten.

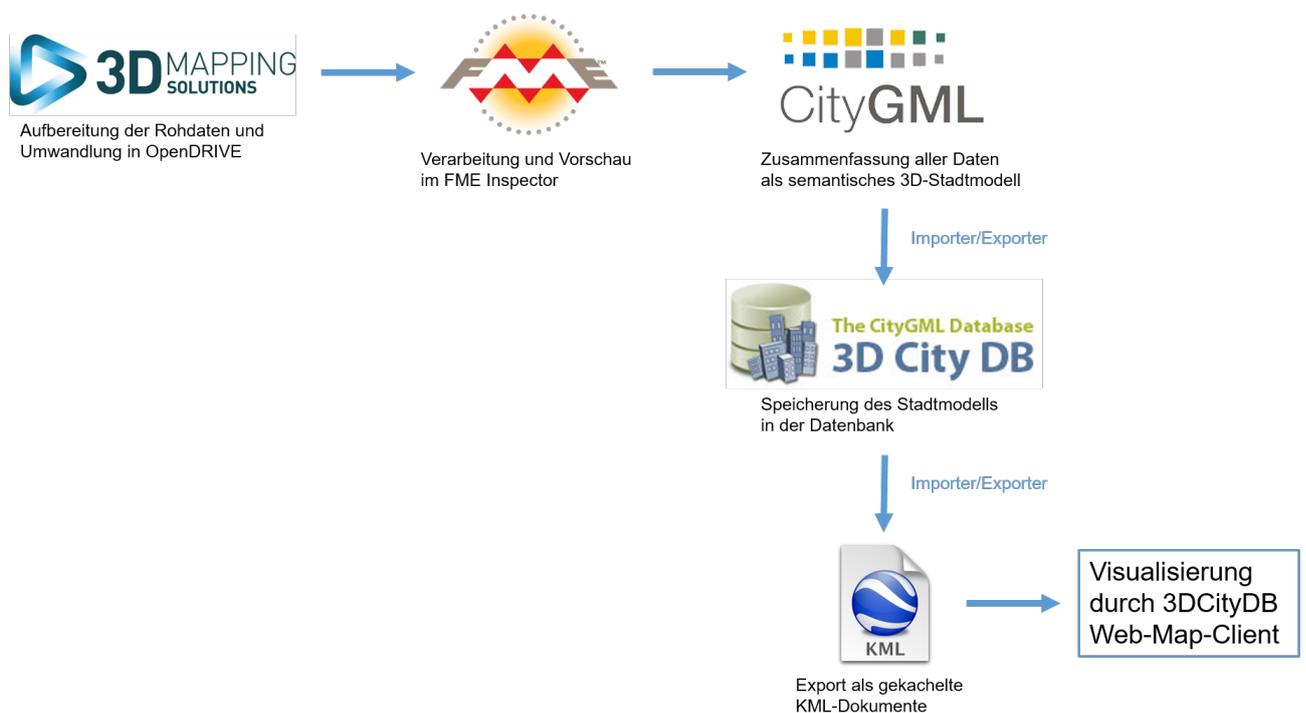


Abbildung 7.1 – Ablaufdiagramm der Datenbearbeitung mit verwendeter Software

7.1 Verwendete Software-Pakete

Die Aufbereitung der MMS-Daten für die Konzeptumsetzung und die anschließende Visualisierung erfolgt in der Firma 3D Mapping Solutions mit interner Software. Die Vorgehensweise zur Datenerfas-

sung und die verwendete Software kann Abschnitt 4.2 entnommen werden. Die Berechnung erfolgte auf Rechnern mit folgender Ausstattung:

Betriebssystem:	Microsoft Windows 10 (6.1.7601 Service Pack 1 Build 7601)
Prozessor:	2x Intel Xeon E5-2609 0 (4 Kerne)
RAM:	32 GB

Im Folgenden werden die für die Berechnungen verwendeten Software-Pakete vorgestellt.

7.1.1 Feature Manipulation Engine

Die Software FME der kanadischen Firma Safe Software Inc. ist ein Werkzeug zur Modelltransformation, welches insbesondere in der Geoinformatik Verwendung findet. Mit dieser Softwarelösung können komplexe Verarbeitungsprozesse von räumlichen und nicht-räumlichen Daten modelliert und automatisiert ausgeführt werden. Die Software basiert auf dem Spatial Extract, Transform, Load (ETL) Ansatz. FME unterstützt das Importieren (Extract) von einer Vielzahl verschiedener Datenquellen wie herkömmlichen Datenformaten von GIS, CAD-Systemen, Web Services, räumlichen Datenbanken, 3D-Formate und einer Vielzahl von nicht-räumlichen Formaten [con terra GmbH, 2015].

Das CityGML-Format wird im Gegensatz zu OpenDRIVE unterstützt. Da jedoch die Möglichkeit zum Einlesen generischer XML-Dokumente vorliegt, ist auch die Verarbeitung von OpenDRIVE-Dokumenten grundsätzlich möglich. Außerdem beinhaltet FME zahlreiche Transformer für die Manipulation von Koordinaten, Objekte und andere Eigenschaften und für räumliche Analysen (Transform). Insbesondere die Umwandlung von Datenformaten, die Transformation von Datenmodellen und die Integration unterschiedlicher Datentypen stehen dabei im Vordergrund. Abschließend ist die Datenausgabe in zahlreiche Zieldatenmodelle bzw. Formate und das unmittelbare Einlesen in entsprechende GI-, CAD- oder Business-Systeme möglich (Load) [con terra GmbH, 2015]. FME eignet sich aus diesem Grund besonders gut zur Umwandlung verschiedener Datenformate. Da die Zusammenstellung und Verbindung heterogener Daten eine typische Problemstellung bei der Erstellung von Stadtmodellen und daran anknüpfenden Fragestellungen ist, wird dabei häufig auf FME zurückgegriffen [Beil, 2017; Burger, 2015]. Für die praktische Umsetzung der erarbeiteten Konzepte für diese Arbeit wurde FME in der Version 2017.0 eingesetzt.

7.1.2 3D City Database

Allgemein bringt die Verarbeitung und Speicherung von 3D-Stadtmodellen aufgrund sehr großer Datenmengen besondere Herausforderungen mit sich. Dies gilt auch für die gegebenen Daten. Einige äußerst nützliche Werkzeuge hierfür stellt die OpenSource-Software 3DCityDB zur Verfügung, die vom Lehrstuhl für Geoinformatik der TUM in Kooperation mit den Unternehmen virtualcitySYSTEMS und M.O.S.S. Computer Graphik Systeme GmbH entwickelte wurde. Die 3DCityDB beinhaltet speziell

für umfangreiche 3D-Stadtmodelle ein Datenbankschema zum Importieren, Verwalten, Analysieren, Visualisieren und Exportieren sämtlicher relevanter Daten. Das Datenbankschema ergibt sich aus einer Übertragung des objektorientierten Datenmodells CityGML 2.0 auf ein räumlich erweitertes relationales Datenbankmanagementsystem (Spatial Relational Database Management System, SRDBMS). Durch eine Java-basierte Client-Anwendung, dem 3DCityDB Importer/Exporter, ist das Einlesen und Ausgeben beliebig großer CityGML-konformer Datensätze möglich. Das Erzeugen von 3DVisualisierungsmodellen in KML, COLLADA und gITF wird unterstützt. Selbst sehr große 3DStadt- und Landschaftsmodelle können durch Kachelung der Datensätze in GIS oder digitalen virtuellen Globen wie Google Earth und Cesium WebGL Virtual Globe visualisiert werden. Zusätzlich können Attributtabelle in unterschiedlichen Formaten generiert werden [Kolbe et al., 2016].

7.1.3 3DCityDB Web-Map-Client

Der 3DCityDB Web-Map-Client (3D-Web-Client) gehört ab der Version 3.3.0 zum Softwarepaket der 3DCityDB. Dadurch sind 3D-Visualisierungen und interaktive Analysen selbst von großen semantischen 3D-Stadtmodellen möglich. Der Client wurde auf Basis des Cesium Virtual Globe, einer Open-Source-JavaScript-Bibliothek, entwickelt. Er bietet plattformübergreifende Funktionalitäten wie das Anzeigen von 3D-Grafikinhalten im Webbrowser ohne zusätzlich notwendige Plugins. Bei der Entwicklung des Clients wurden verschiedene spezialisierte Erweiterungen für den Cesium Virtual Globe entwickelt, um den Nutzern die Möglichkeit zu geben 3D-Stadtmodelle komfortabler anzusehen und zu erkunden. Die mit dem Import/Export-Tool der 3DCityDB exportierten KML/gITF-Modelle können so in einem Webbrowser visualisiert werden. In Verbindung mit den erzeugten Attributtabelle sind Abfragen der thematischen Attribute jedes Stadtobjekts möglich [Kolbe et al., 2016]. Der 3D-Web-Client ermöglicht neben der reinen Visualisierung auch die Bearbeitung von Objekte der virtuellen 3D-Stadtmodelle einschließlich relevanter thematischer Informationen im Webbrowser [Yao und Kolbe, 2014]. Dies begründet zugleich die geplante Verwendung des 3D-Web-Client in dieser Arbeit. Die durch das Mobile Mapping erfassten Daten können so vollumfänglich betrachtet und für Kunden zugänglich gemacht werden.

7.2 Analyse der Grundlagendaten

Zur Bearbeitung dieser Masterarbeit wurden von der Firma 3D Mapping Solutions kinematisch erfasste und für ein OpenDRIVE-Dokument verarbeitete Daten bereitgestellt. Anhand dieser Grundlagendaten soll der in Kapitel 6 erweiterte Erfassungsprozess erprobt und mit einer dreidimensionalen Visualisierung des Straßenraums dargestellt werden.

Der Erfassungsprozess von 3D Mapping Solutions bis zur Erzeugung eines validen OpenDRIVE-Dokuments wurde bereits in Abschnitt 4.2 beschrieben. Für die Visualisierung werden alle Daten des CAD-Dokuments in eine Ebenenstruktur eingeordnet und als DWG-Datei exportiert. Beim Import der DWG-Datei in FME werden die Elemente der im CAD erzeugten Ebenenstruktur als Attribut

fme_feature_type übergeben. Die meisten Informationen können daher aus diesem Attribut extrahiert werden. Informationen, die im CAD-Projekt bzw. in der DWG-Datei nicht explizit in der Ebenenstruktur gespeichert sind, werden nicht mit übergeben und müssen in FME auf anderem Weg berechnet werden. Abb. 7.2 zeigt beispielhafte die Struktur in der DWG-Datei und in FME.

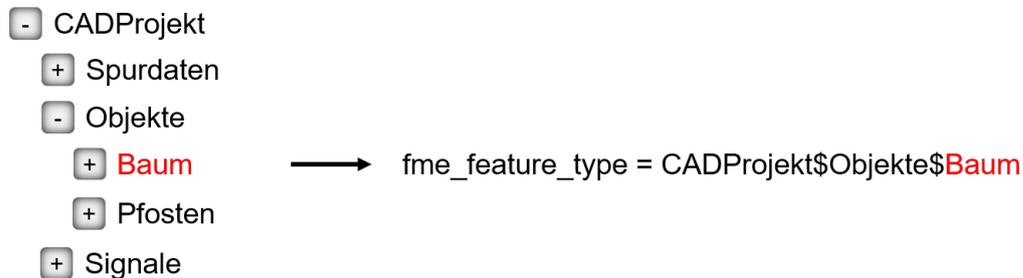


Abbildung 7.2 – Beispielhafte Ebenenstruktur in der DWG-Datei (links) und in FME (rechts)

Neben der DWG-Datei werden auch die txt- und xml-Dateien aus dem Schritt der Datenextraktion und die fertiggestellte OpenDRIVE-Datei bereitgestellt. Für die Modellierung der Objekte wurden 3D-Modelle aus dem SketchUp 3D Warehouse verwendet. Quellen zu den verwendeten Modellen werden in Anhang C aufgelistet. Diese werden als SketchUp 2017 Modelle heruntergeladen, in FME angepasst und als Collada-Dateien abgespeichert.

7.3 Weiterverarbeitung in FME

Im Folgenden werden alle Weiterverarbeitungsschritte bis zum fertigen 3D-Straßenraummodell aufgezeigt. Dabei wird zwischen der Modellierung des Spurmodells und der Modellierung der Objekte unterschieden.

7.3.1 Modellierung des Spurmodells

Im ersten Schritt werden die OpenDRIVE-Referenzlinien als linienhafte Beschreibung der Straße visualisiert. Zur flächenhaften Modellierung des Straßenkörpers werden die extrahierten Spurdaten auf Höhe 0 und auf tatsächlicher Höhe weiterverarbeitet. Die erstellten Modelle werden Kapitel 9 verglichen.

Darstellung der Referenzlinien

Zur Darstellung der Referenzlinien werden die in der DWG-Datei enthaltenen Achsen als Geometrien verwendet. Die Attributwerte werden direkt aus dem OpenDRIVE-Dokument entnommen. Eine Veränderung der *xodr*-Dateiendung in *xml* macht das Einlesen in FME möglich. Über den eindeutigen Namen der Achsen können Attribute auf die Geometrien übertragen werden. Jeder Kreuzungsbereich wird durch eine eindeutige ID beschrieben. Die Referenzlinien einer Kreuzung verweisen durch das Attribut *opendrive_road_junction* auf die ID des gesamten Kreuzungsbereiches. Handelt es sich bei einer

Referenzlinie um die Referenzlinie eines normalen Straßenstücks, welches keinem Kreuzungsbereich zugeordnet werden kann, so wird der Wert als -1 angegeben. Zusätzlich werden die Vorgänger- und Nachfolgerachsen aus dem OpenDRIVE-Dokument extrahiert. Einen Überblick über die Attribute liefert Tabelle 7.1.

Tabelle 7.1 – Generierte Attribute der Referenzlinie

Attributname	Wert	Beschreibung
<code>opendrive_type</code>	2dRefline, 3dRefline	Dimension der Referenzlinie
<code>opendrive_road_id</code>	xxx	Eindeutige ID der Referenzlinie
<code>opendrive_road_junction</code>	xxx	ID des zugehörigen Kreuzungsbereichs,
	-1	Defaultwert für Referenzlinien einer Straße
<code>od_road_link_predecessor</code>	xxx	ID des Vorgängers
<code>od_road_link_successor</code>	xxx	ID des Nachfolgers
<code>opendrive_road_length</code>	xx	Straßenlänge in [m]
<code>opendrive_road_usage</code>	motorway, town, bicycle, pedestrian, etc.	Straßentyp

Spurmodell auf Höhe 0

Aus der Spurberechnung im CAD ergeben sich Schraffuren, die beim Import der DWG-Datei in FME als ebene Polygone mit Höhe 0 angezeigt werden. Informationen zum Spurtyp werden explizit nicht übergeben. Die Polygone werden allerdings je nach Spurtyp in einer bestimmten Farbe dargestellt. Im ersten Schritt werden die Polygone mit Hilfe des Farbwerts nach Spurtyp klassifiziert und mit dem Attribut `opendrive_lane_type` angereichert (Spurtypen siehe Tabelle 4.1).

Für die Darstellung in einem 3D-Modell müssen erhöhte Flächen wie Bordsteine und Gehwege dreidimensional dargestellt werden. Die gegebenen Flächen enthalten jedoch keinerlei Höheninformationen. Zur Modellierung der Höhen werden aus diesem Grund folgende Annahmen für die OpenDRIVE-Spurtypen getroffen:

- Fahrspur (driving), Rinnstein (gutter) mit Höhenoffset $h = 0$ m
- normaler Bordstein (border), Grundfläche (green) mit Höhenoffset $h = 0,15$ m
- abgesenkter Bordstein (border), Parkfläche (parking), befestigter Seitenstreifen (shoulder) mit Höhenoffset $h = 0,025$ m
- Bordsteinrampe (border), Gehweg (sidewalk) mit angepasstem Höhenoffset

An abgesenkten Bordsteinen müssen angrenzende Flächen speziell behandelt werden. In den meisten Fällen grenzen Gehwege und Bordsteinrampen an (vgl. Abb. 7.3). Die Eckpunkte, die direkt an der abgesenkten Bordsteinfläche liegen, in der Abbildung gekennzeichnet durch orange Kreise, erhalten einen Höhenoffset von 0,025 m. Alle anderen Eckpunkte der angrenzenden Flächen werden um 0,15 m erhöht, in der Abbildung gekennzeichnet durch grüne Kreise.

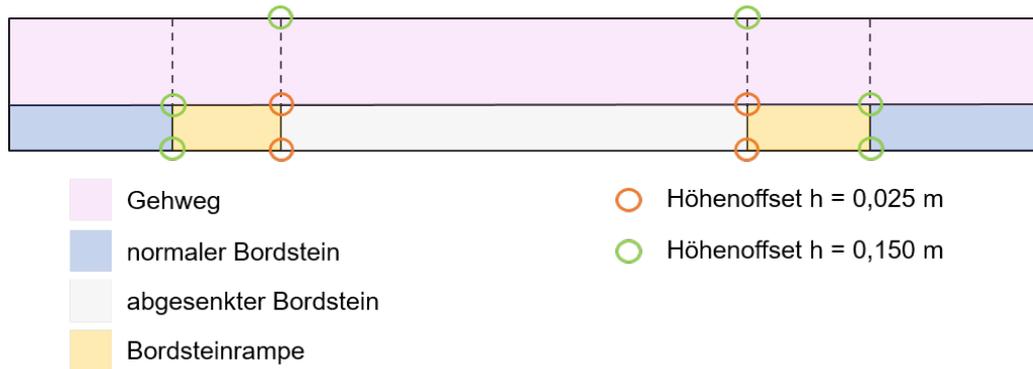


Abbildung 7.3 – Modellierung dreidimensionaler Randstein- und Gehwegflächen

Die Flächen werden dazu in ihre Eckpunkte zerlegt. Mit einer Nachbarschaftsanalyse werden direkt anliegende Eckpunkte von den weiter entfernt liegenden Eckpunkten separiert und in der Höhe angepasst. In den verwendeten Daten können für die angenommenen Höhenangaben der Spurtypen *green*, *parking* und *shoulder* vereinzelt Fehler auftreten und zu fehlerhaften Darstellungen führen. Dies wird an dieser Stelle nicht weiter berücksichtigt.

Im nächsten Schritt werden die Spurflächen um weitere Attribute angereichert. Aus dem Objektnamen kann der Name der Referenzlinie extrahiert werden zu der die jeweilige Fläche gehört. In OpenDRIVE werden die Spuren mit Werten aufsteigend in positiver t -Richtung, d.h. gegen die Fahrtrichtung und absteigend in negativer t -Richtung, d.h. in Fahrtrichtung, angegeben (vgl. Abb. 7.4). Durch das Vorzeichen kann somit die Orientierung der Fahrspur bestimmt werden.

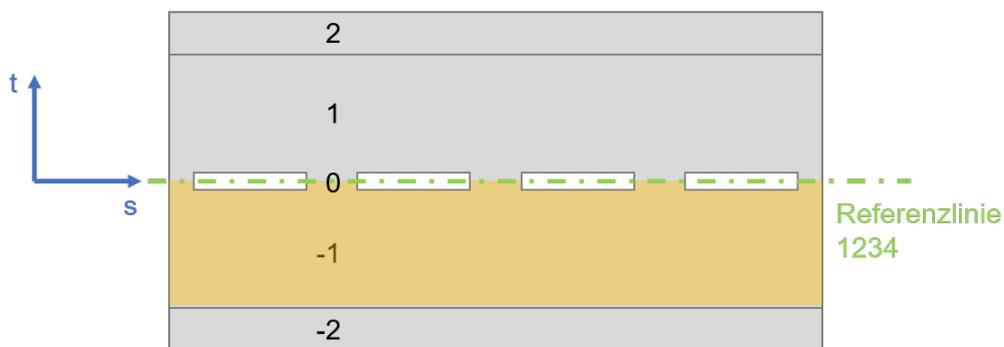


Abbildung 7.4 – Beschreibung möglicher Attributwerte einer Fahrspur

Für die in Abb. 7.4 farblich gekennzeichnete Fahrspur können so beispielhaft folgende Attributwerte extrahiert werden:

- `opendrive_lane_type = driving`
- `opendrive_lane_id = -1`
- `opendrive_road_id = 1234`
- `opendrive_lane_orientation = -`

Für die Darstellung werden Fahrspuren mit gleichem Spurtyp innerhalb einer Referenzlinie zusammengefasst. Benachbarte Flächen mit gleichem Spurtyp werden durch ein Attribut mit gleichem

Wert gekennzeichnet. Unter zusätzlicher Berücksichtigung des Spurtyps, der Spurorientierung, der zugehörigen Referenzlinie und des generierten Nachbarschaftsattributs werden zusammenhängende Flächen verbunden und als eine Fläche dargestellt (vgl. Abb. 7.5).

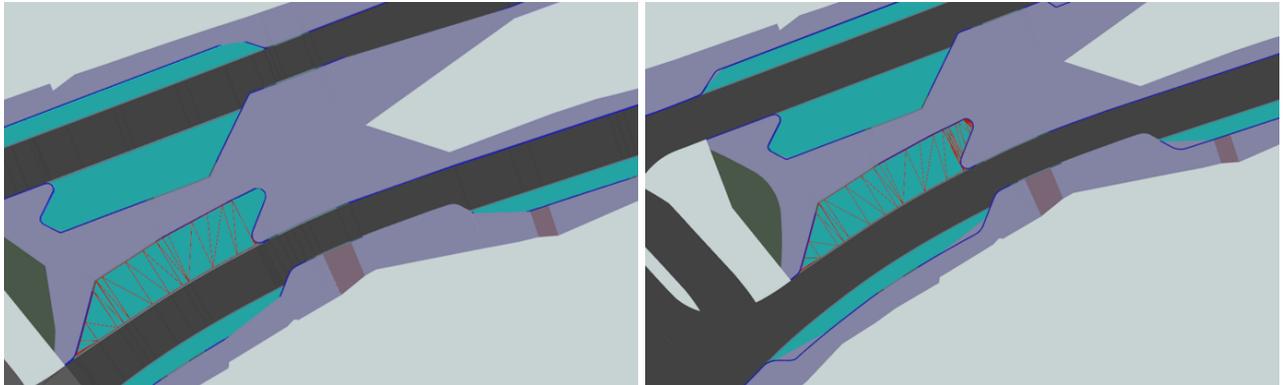


Abbildung 7.5 – Darstellung einer Parkfläche vor der Generierung eines Zusatzattributs (links) und als zusammenhängende Fläche (rechts)

Der Spurtyp *shoulder* wird für die Darstellung als Gehweg (sidewalk) und der Spurtyp *gutter* als Fahrbahn (driving) implementiert. Ein Fußgängerouting auf Gehwegen oder Abfragen bezüglich Gehwegen wäre andernfalls nur bedingt möglich. Dafür wird ein zusätzliches Attribut eingeführt, welches jedoch nicht das OpenDRIVE-Attribut *opendrive_lane_type* ersetzt.

Das Spurmodell wird mit Fahrbahnmarkierungen, welche als linienhafte Objekte importiert werden, ergänzt. Durchgehende und strichlierte Markierungen können durch die Einteilung in *solid* und *broken* unterschieden werden. Die OpenDRIVE-Spezifikation implementiert weitere Markierungsarten, die an dieser Stelle nicht unterschieden werden. Neben der Markierungsart kann auch die Breite der Fahrbahnmarkierungen aus dem Objektnamen entnommen werden. Diese Werte werden in den Attributen *roadmark_type* und *roadmark_width* gespeichert. Durch die Breitenangabe können die linienhaften Objekte zu Flächen erweitert werden und mit den zugrundeliegenden Fahrspuren verschnitten werden.

Spurmodell auf absoluter Höhe

Zur Überprüfung der Daten werden im CAD aus den berechneten Spurdaten und Referenzlinien dreidimensionale Geometrien berechnet. Diese dienen der visuellen Überprüfung des Spurmodells. Durch Export und Import der Daten werden die Volumenkörper in FME zerlegt als einzelne, ebene Flächengeometrien repräsentiert. Aus dem Objektnamen ist nur der Name der zugehörigen Referenzlinie ableitbar. Die Einteilung in Spurtypen geschieht durch die Verbindung mit den farblich repräsentierten, zweidimensionalen Flächen. Dafür werden die Mittelpunkte der Flächengeometrien konstruiert und auf die zweidimensionalen Polygone projiziert. Durch Überlappungsanalysen können Attributwerte der Polygone auf die Mittelpunkte und damit auf die 3D Geometrien übertragen werden.

Die Anreicherung der Attribute und Generierung der Fahrbahnmarkierungen erfolgt vergleichbar zur Modellierung des Spurmodells auf Höhe 0. Für eine realitätsnahe Visualisierung wird das Spurmodell mit Hilfe von Rastergrafiken texturiert dargestellt. Aus den aufgezeigten Verarbeitungsschritten ergibt sich ein texturiertes dreidimensionales Spurmodell auf Höhe 0 und auf absoluter Höhe. Bei der Modellierung werden die in Tabelle 7.2 dargestellten Attribute generiert.

Tabelle 7.2 – Generierte Spurattribute

Attributname	Wert	Beschreibung
opendrive_lane_type	driving, sidewalk, etc.	Spurtyp
opendrive_road_id	xxx	ID der Referenzlinie
opendrive_lane_id]..., -2, -1, 1, 2, ...[Spurnummer
opendrive_lane_orientation	+, -	Spurorientierung gegen/in Fahrtrichtung
roadmark_type	sol, bro	Durchgezogene/Strichlierte Markierung
roadmark_width	xx	Breite der Fahrbahnmarkierung [m]

7.3.2 Modellierung der Straßenraumobjekte

In OpenDRIVE werden alle Objekte entlang einer Straße in Objekte und Signale unterteilt. Zur weiteren Verarbeitung werden die Objekte nach Art der Extraktion unterschiedlich behandelt. Der erweiterte Auswerteprozess kann so auf alle Objekte mit der gleichen oder zumindest ähnlichen Extraktionsart, wie in Tabelle 7.3 dargestellt, übertragen werden. Zum Beispiel können alle Objekte, die durch Bodenpunkt und höchsten Punkt beschrieben werden, auf die gleiche Art und Weise wie Pfosten oder Bäume modelliert werden. Dreidimensionale Objekte werden durch explizit erstellte 3D-Modelle oder durch die Generierung dreidimensionaler Klötzchenmodelle aus den Grundflächen konstruiert und visualisiert. Die Objektdaten liegen auf absoluter Höhe vor und werden auf dieser Höhe für die Visualisierung modelliert. Für die Darstellung auf Höhe 0 wird lediglich der Höhenwert der Objekte angepasst.

Tabelle 7.3 – Modellierung der Objekte nach Art der Extraktion

Extrahierte Daten	Objektbeispiele	Darstellung
1. Bodenpunkt mit Höhe und Durchmesser	Pfosten, Baum	Explizites 3D-Modell
2. Bodenpunkt mit Höhe, Durchmesser und Orientierung	Straßenlaterne	Explizites 3D-Modell
3. Objektpunkt, Orientierung und Kategorien	Schilder, Ampeln	Explizites 3D-Modell
4. Umriss (mit Orientierung)	Parkbank	Explizites 3D-Modell
5. Umriss am Boden	Straßenmarkierungen, Fußgänger- und Fahrradfurten	Flächenhafte Darstellung
6. Umriss der Oberkante	Verteilerkasten, Mauer	Extrudiertes Klötzchenmodell

1. Objekte beschrieben durch Bodenpunkt, Höhe und Durchmesser

Objekte wie Bäume oder Pfosten werden in dieser Arbeit aus der Punktwolken extrahiert und durch den Bodenpunkt, den höchsten Punkt des Objekts und die Angabe des Durchmessers charakterisiert. Durch terrestrisches Mobile Mapping erzeugte Punktwolken zeigen Objekte am Fahrbahnrand entlang der Straße. Dabei ist die Objektseite, die zum Fahrbahnrand bzw. zur Trajektorie gerichtet ist, am besten sichtbar. Als Bodenpunkt wird im Erfassungsprozess daher nicht der Mittelpunkt, sondern lediglich der unterste sichtbare Objektpunkt gemessen.

Die Modellierung der Objekte erfolgt in zwei Schritten. Im ersten Schritt werden alle notwendigen Parameter in FME als Attribute gespeichert. Die Höhe des Objekts kann durch den gegebenen Bodenpunkt und höchsten Punkt berechnet werden. Der Durchmesser wird aus dem Objektname extrahiert und von Zentimeter in Meter umgerechnet. Der gegebene Bodenpunkt wird um die Hälfte des Durchmessers vom Fahrbahnrand gesehen nach hinten verschoben (vgl. Abb. 7.6). Dafür wird vom Objektpunkt ein Lot zur gegebenen Referenzlinie gezogen und der Punkt entlang des Lots verschoben. Dadurch kann aus dem gemessenen Punkt der am Boden liegende Objektmittelpunkt gewonnen werden. Die Attribute *object_height*, *object_diameter* und *object_radius* werden generiert und den erzeugten Objektmittelpunkten als Attributeinträge zugewiesen.

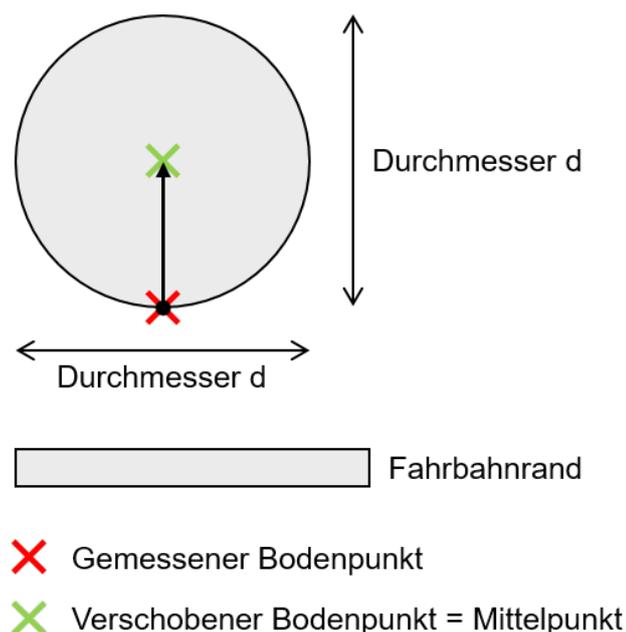


Abbildung 7.6 – Erzeugung des Objektmittelpunkts aus gemessenem Bodenpunkt

Im zweiten Schritt werden dreidimensionale Geometriemodelle mit den erstellten Punktgeometrien und zugehörigen Attributen verbunden. Als Grundobjekte werden 3D-Modelle des SketchUp 3D Warehouses als Collada-Dateien in FME eingelesen und normiert. Die Generierung der dreidimensionalen Geometriemodelle erfolgt explizit. Die 3D-Modelle werden mit den Punktkoordinaten verknüpft und anhand der Attributwerte verschoben, skaliert und mit den jeweiligen Attributen angereichert, wie in

Abb. 7.7 schematisch dargestellt.

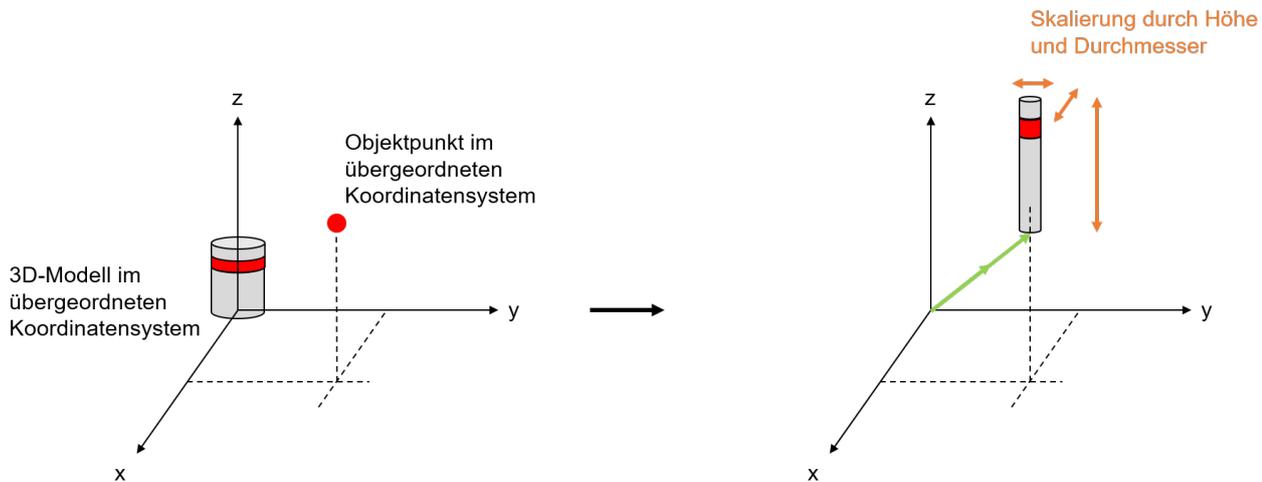


Abbildung 7.7 – Erzeugung von 3D-Modellen durch Verschiebung und Skalierung

Für die Modellierung eines Baums muss die Skalierung speziell behandelt werden, da er sowohl einen Stamm- als auch einen Kronendurchmesser besitzt. Die Skalierung der Baummodelle wird im implementierten Workflow nach Kronendurchmesser durchgeführt, die extrahierten Daten enthalten allerdings nur Werte des Stammdurchmessers. Zur korrekten Darstellung des Baumstammes müssen die Skalierungswerte demnach im Verhältnis Krone-Stamm angepasst werden. Die verwendeten 3D-Baummodelle besitzen das Krone-Stamm-Verhältnis 1:15, wodurch eine Skalierung in x und y Richtung um den 15-fachen Stammradius implementiert werden muss.

2. Objekte beschrieben durch Bodenpunkt, Höhe, Durchmesser und Orientierung

Einige Objekte werden zusätzlich zu Position, Höhe und Durchmesser mit einer räumlichen Orientierung beschrieben. Der Orientierung des Auslegers einer Straßenlaterne beispielsweise ist nicht nur für die realitätsnahe Darstellung notwendig, sondern auch für weiterführende Analysen wie der Berechnung der Lichteinstrahlung in 3D-Stadtmodellen von Bedeutung.

Die Vorgehensweise zur Generierung der Attributwerte wird vergleichbar zum vorangegangenen Ansatz durchgeführt. Zur Berechnung der Orientierung werden der Bodenpunkt und der höchste Punkt in eine Ebene gebracht und miteinander verbunden. Dadurch kann der Horizontalwinkel zwischen den beiden Punkten, der demnach auch die Orientierung des Objekts wiedergibt, berechnet werden. Zusätzlich zu den Attributen *object_height*, *object_diameter* und *object_radius* wird das Attribut *object_rotation* generiert. In diesen Fällen werden die 3D-Geometriemodelle bereits vor der Verknüpfung einheitlich nach Norden ausgerichtet. Durch die anschließende Verknüpfung der Modelle mit den Punktgeometrien können diese korrekt positioniert, skaliert und orientiert werden, wie Abb. 7.8 ergänzend zu Abb. 7.7 zeigt.

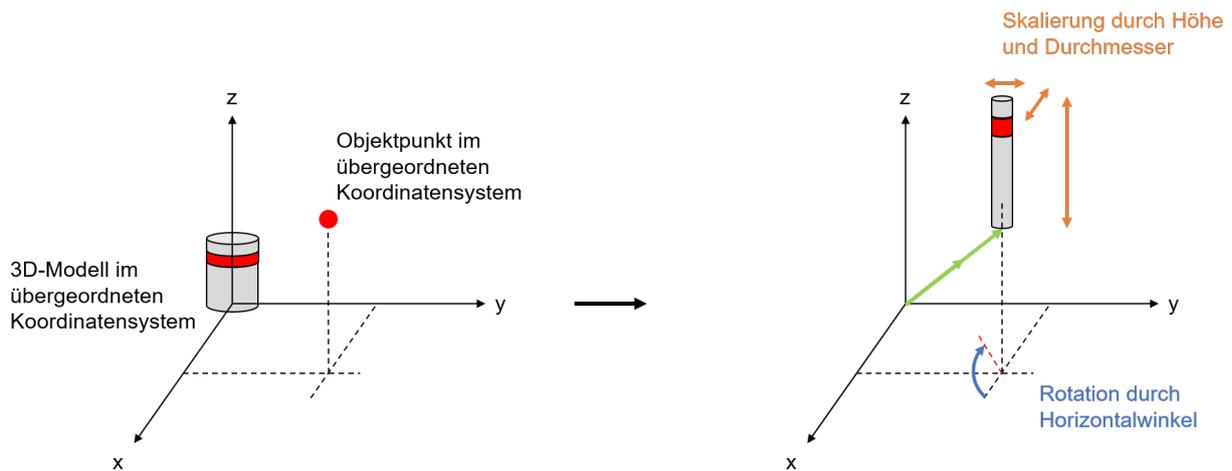


Abbildung 7.8 – Erzeugung von 3D-Modellen durch Verschiebung, Skalierung und Orientierung

3. Objekte beschrieben durch Objektpunkt, Orientierung und Kategorien

Für Objekte wie Ampeln und Verkehrszeichen ist die Orientierung ebenfalls von großer Bedeutung. Die entsprechenden Objekte werden zusätzlich durch einzelne Punktkoordinaten und Kategorien charakterisiert. Die Kategorien der Objekte werden nach dem Verkehrszeichenkatalog (VzKat) 2017 der Bundesanstalt für Straßenwesen eingeordnet. Die Kategorien, bestehend aus Hauptkategorie und Unterkategorie, können in FME aus dem Objektnamen extrahiert und als Attribute *maincategory* und *subcategory* gespeichert werden.

Die Vorgehensweise zur Generierung dreidimensionaler Modelle ist vergleichbar zum bereits vorgestellten Ansatz. Im Vorhinein werden für alle verwendeten Verkehrszeichen 3D-Modelle erstellt und als Collada-Dateien gespeichert. Die Collada-Dateien werden durch die Hauptkategorie und die Unterkategorie eindeutig klassifiziert. Beispiele für texturierte 3D-Modelle von Verkehrszeichen zeigt Abb. 7.9. Dadurch ist eine Zuordnung anhand des Namens von Modell und Objekt möglich. Die Positionierung und Orientierung erfolgt wiederum durch die Verknüpfung von Modell und Objekt und der zugehörigen Attribute, wie bereits zuvor in Abb. 7.8 gezeigt.



3D-Modell:
Richtzeichen_314-10.dae



3D-Modell:
Vorschriftzeichen_206.dae

Abbildung 7.9 – Kategorisierung der Collada-Dateien

4. Objekte beschrieben als Umriss (mit Orientierung)

Für einige Objekte wie Hydranten oder Parkbänke wird eine Fläche, die in großer Nähe der Oberkante entspricht, in der Punktwolke gemessen. Zur Modellierung eines 3D-Modells ist die Position des Objekts am wichtigsten, außerdem ist eine unterschiedliche Betrachtung je nach Objektart notwendig. Objekte mit quadratischem oder rundem Grundriss wie Hydranten können durch Position und Höhe ausreichend dargestellt werden. Durch die Berechnung des Mittelpunktes des gemessenen Umrisses und der Verschiebung auf Bodenhöhe kann der Grundrissmittenpunkt, welcher als Einfügeposition dient, berechnet werden. Die Höhe ergibt sich aus den beiden berechneten Punkten. Die Generierung erfolgt analog der 1. Objektart. Für langgestreckte Objekte wie Parkbänke ist es sinnvoll, nicht nur einen Mittelpunkt, sondern die gesamte Mittellinie zu extrahieren. Der Mittelpunkt kann zur Positionierung des Objekts an der richtigen Stelle verwendet werden. Aus der Mittellinie kann durch Horizontalwinkelberechnung die Orientierung des Objekts berechnet werden, wie Abb. 7.10 beispielhaft zeigt.

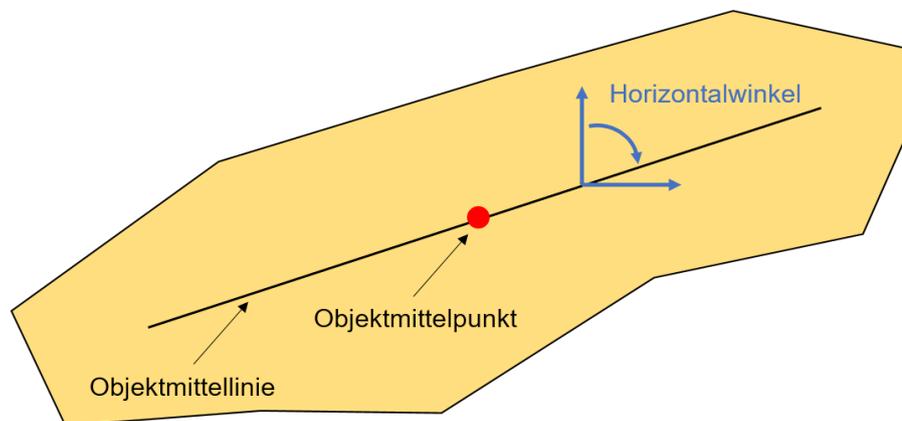


Abbildung 7.10 – Berechnung der Objektorientierung durch die Mittellinie

Die Breitenskalierung des Objekts kann aus der Länge der Mittellinie berechnet werden. Dazu wird zusätzlich das Attribut *object_length* generiert. Die Generierung der 3D Modelle kann wie in Abb. 7.8 dargestellt durchgeführt werden.

5. Objekte beschrieben als Umriss am Boden

Flächenhafte Objekte wie Fußgängerfurten, Stoplinien und andere Symbole auf der Fahrbahn werden durch ihren Umriss bestimmt. Eine Verschneidung mit den darunterliegenden Flächen oder eine Erhöhung der gesamten Objektfläche ist für die Visualisierung notwendig, um Überlappungseffekte zu vermeiden. Für diese Objektart wird das Attribut *object_area* generiert.

6. Objekte beschrieben als Umriss der Oberkante

Von dreidimensionalen Objekten mit einer polygonförmigen Grundfläche wie zum Beispiel Verteilerkästen wird die Oberkante als Umriss gemessen. Die extrahierten Flächen werden in FME für die Darstellung nach unten extrudiert und mit der Bodenfläche verschritten. Dadurch entstehen

dreidimensionale Klötzchenmodelle mit korrekten Dimensionen. Das Volumen der Objekte kann im Attribut *object_volume* festgehalten werden.

Zusätzlich zu den beschriebenen Attributen werden für alle Objekte die in Tabelle 7.4 dargestellten Attribute generiert. Die Objektkoordinaten beschreiben den Objektmittelpunkt. Als Attributwerte werden die Objekttypen aus der OpenDRIVE-Formatspezifikation verwendet (vgl. Tabelle 4.2).

Tabelle 7.4 – Generierte Objektattribute

Attributname	Wert	Beschreibung
object_type	z.B. pole	Objekttyp
object_coordinates	x,y,z	Objektkoordinaten [m]

7.3.3 Umwandlung nach CityGML

Das modellierte Straßenraummodell wird abschließend mit Attributen der jeweiligen CityGML-Klassen angereichert und mit Hilfe von FME-Writern ins CityGML-Format geschrieben.

1. Umwandlung des Spurmodells

Die Verkehrsflächen werden durch das thematische CityGML-Erweiterungsmodul *Transportation* repräsentiert. Die Referenzlinien können durch die Unterklasse *Road* als linienhafte Darstellung im LoD 0 beschrieben werden. Zu den in Unterabschnitt 7.3.1 beschriebenen Attributen werden zusätzliche Attribute generiert, die in Tabelle 7.5 aufgelistet werden. Die bei der Modellierung generierten (OpenDRIVE-) Attribute werden dabei als generische Attribute gespeichert.

Tabelle 7.5 – Attribute der Referenzlinien

Attributname	Werte	Beschreibung
gml.id	id.xxx	Eindeutige ID der Referenzlinie
gml.parent.id	fme_gen_xx	Eindeutige ID der Elternklasse
gml.name	xxx	Name der Referenzlinie
citygml.level_of_detail	0	Detaillierungsgrad
citygml.lod_name	lod0Network	Geometrie
citygml.function	1	driving
citygml.usage	1	car
citygml.surface_material	z.B. 9999	unknown

Eine flächenhafte Darstellung des Straßenraums kann ab LoD 1 umgesetzt werden. Die Grundlagendaten erlauben allerdings eine Einteilung in die Unterklassen *TrafficAreas* und *AuxiliaryTrafficAreas*, welche nur in den Stufen LoD 2-4 verfügbar sind. Die Zugehörigkeit ist vom Attribut *citygml.feature_role* ableitbar. Das generierte Spurmodell kann damit als *lod3Multisurface*-Geometrie beschrieben werden. Die zugehörigen Attribute sind in Tabelle 7.6 dargestellt.

Tabelle 7.6 – Attribute des Spurmodells

Attributname	Werte	Beschreibung
<i>gml_id</i>	id_xxx	Eindeutige ID
<i>gml_parent_id</i>	fme_gen_xx	Eindeutige ID der Elternklasse
<i>gml_name</i>	xxx	Name der Referenzlinie
<i>citygml_feature_role</i>	AuxiliaryTrafficArea TrafficArea	Objektrolle
<i>citygml_level_of_detail</i>	3	Detaillierungsgrad
<i>citygml_lod_name</i>	lod3MultiSurface	Geometrie
<i>citygml_function</i>	z.B. 1	driving_lane
<i>citygml_usage</i>	z.B. 2	car
<i>citygml_surface_material</i>	z.B. 1	asphalt

Das Attribut *gml_name* beinhaltet den Namen des Objekts. Beil [2017] schlägt als Kriterium zur Zerlegung einer Straße in mehrere Straßensegmente die Verwendung des Straßennamens vor. Alternativ könnte, sofern vorhanden, auch ein wechselnder Straßenbelag Grund für eine Zerlegung in kleinere Segmente sein. Da Straßennamen für Simulationszwecke nur eine untergeordnete Rolle spielen, werden diese in der Datenerfassung von 3D Mapping Solutions und innerhalb der OpenDRIVE-Dokumente nicht aufgenommen. In OpenDRIVE wird das Straßennetz in Referenzlinien unterteilt. Jede Referenzlinie wird dabei durch eine eindeutige ID beschrieben. Als eindeutiges Attribut *gml_name* kann hier direkt der Inhalt des Attributs *opendrive_road_id*, das die eindeutige ID des Road-Elements beschreibt, verwendet werden.

2. Umwandlung der Straßenraumobjekte

Die Straßenraumobjekte werden durch das thematische CityGML-Erweiterungsmodul *CityFurniture* beschrieben. Das CityGML-Erweiterungsmodul *Vegetation* wird zur Repräsentation von Vegetationsobjekten wie Bäume und Büsche verwendet. Die Objekte werden als *cityObjectMember* im LoD 2 modelliert. Alle generierten Attribute werden in Tabelle 7.7 und Tabelle 7.8 dargestellt.

Tabelle 7.7 – Attribute der CityFurniture-Objekte

Attributname	Wert	Beschreibung
<i>gml_id</i>	id_xx	Eindeutige ID des Objekts
<i>gml_parent_id</i>	fme_gen_xx	Eindeutige ID der Elternklasse
<i>gml_name</i>	z.B. 1270	Codename des Objekts
<i>gml_description</i>	pole	Beschreibung des Objekts
<i>citygml_feature_role</i>	<i>cityObjectMember</i>	Objektrolle
<i>citygml_lod_name</i>	lod2Geometry	Geometrie
<i>citygml_level_of_detail0</i>	2	Detaillierungsgrad
<i>citygml_class</i>	z.B. 1030	others
<i>citygml_function</i>	z.B. 1270	pole
<i>citygml_usage</i>	z.B. 1270	pole

Tabelle 7.8 – Attribute der Vegetation-Objekte

Attributname	Wert	Beschreibung
gml_id	id_xx	Eindeutige ID des Objekts
gml_parent_id	fme_gen_xx	Eindeutige ID der Elternklasse
gml_name	z.B. 1070	Codename des Objekts
gml_description	z.B. 1070	Beschreibung des Objekts
citygml_feature_role	cityObjectMember	Objektrolle
citygml_lod_name	lod2Geometry	Geometrie
citygml_level_of_detail0	2	Detaillierungsgrad
citygml_class	z.B. 1070	Laubbaum

7.4 Darstellung im 3DCityDB Web-Map-Client

Die mit FME generierten, validen CityGML-Dokumente können unter Verwendung des Importer/Exporter Tools in die 3DCityDB importiert werden. Die Datenbank ermöglicht eine effiziente Speicherung großer Datenmengen und einen schnellen Zugriff auf einzelne Daten [Yao und Kolbe, 2014]. Unter erneuter Verwendung des Importer/Exporter können die Daten im CityGML- oder KML-Format exportiert werden. An dieser Stelle ist es möglich ein einzelnes CityGML-Dokument bestehend aus den zu Beginn thematisch getrennten Datensätzen zu generieren. Dies ist für kleine Gebiete zum Austausch oder der Repräsentation des 3D-Stadtmodells durchaus sinnvoll. Für große Datenmengen ist eine Unterteilung in thematische CityGML-Datensätze zur besseren Handhabung oft von Vorteil, da die Darstellung von großen CityGML-Dokumente von den meisten Softwareprodukten bislang nicht bewältigt werden kann.

Die weitere Visualisierung im 3DCityDB Web-Map-Client benötigt die Erzeugung einer KML-Datei. Für jeden thematischen Datensatz (Spurmodell, Straßenmöblierungsobjekte, Vegetationsobjekte) wird eine eigene KML-Datei erstellt. Zusätzlich werden zur separaten Darstellung auch einige Objekttypen einzeln als KML-Datei erzeugt. Dies ermöglicht eine getrennte Darstellung der Straßenraumobjekte und separate Abfragen im Viewer. Die KML-Dateien enthalten die Geometrien der einzelnen Objekte. Der semantische Inhalt wird in Tabellen gespeichert. Diese werden vom 3DCityDB Importer/Exporter aus den CityGML-Attributen erzeugt und exportiert. Im Web-Map-Client wird die Geometrie mit den semantischen Informationen des Objekts gemeinsam dargestellt.

8 Realisierung anhand konkreter Testgebiete

Das in Kapitel 7 beschriebene Vorgehen zur Weiterverarbeitung und Visualisierung der Mobile Mapping Daten wird anhand konkreter Testgebiete in diesem Kapitel realisiert. Im ersten Abschnitt werden anhand eines Beispieldatensatzes alle Schritte der Konzeptumsetzung in FME umgesetzt und im 3DCityDB Web-Map-Client visualisiert. Mithilfe eines weiteren Datensatzes wird dieses Vorgehen im zweiten Abschnitt auf Übertragbarkeit geprüft und validiert. Das Kapitel schließt mit einer kritischen Diskussion des gesamten Konzepts und der Visualisierungsergebnisse ab.

8.1 Visualisierung und Evaluierung von Testgebiet 1

Als Beispieldatensatz dienen Daten der Stadt Grafing. Das Gebiet um den Marktplatz wurde durch die Firma 3D Mapping Solutions im Sommer 2017 kinematisch vermessen. Die Gesamtlänge der Achsen beträgt dabei 1,8 km. Das Gebiet beinhaltet vier Kreuzungsbereiche und zwölf Straßenelemente. In Verbindung mit einem Projekt der Firma CADFEM und deren Tochterfirma virtualcitySYSTEMS GmbH sowie mit der Stadt Grafing wurde ein semantisches, CityGML-basiertes 3D-Gebäudemodell der Stadt erstellt. Das Gebäudemodell wurde dafür vom Landesamt für Digitalisierung, Breitband und Vermessung (LDBV) bereitgestellt, die Texturen auf den Gebäudeflächen wurden von virtualcitySYSTEMS unter Verwendung von Schrägluftbildern angebracht. Durch dieses bereits bestehende Projekt wäre eine Verknüpfung von detaillierten, hoch genauen Straßendaten mit einem virtuellen 3D-Stadtmodell und somit eine realitätsnahe, anschauliche Präsentation der Stadt möglich.

8.1.1 Aufbereitung der Grundlagendaten

Im ersten Schritt wurden die bereitgestellten Daten auf Vollständigkeit analysiert. Die Befahrungsdaten liegen vorverarbeitet als Rohdaten (Trajektorien, Punktwolken und Kameradaten) und ausgewertet im OpenDRIVE-Format vor. Das Dokument beinhaltet von allen Straßen und Kreuzungen ein fertiges Spurmodell mit Informationen zu Spurgeometrien und Spurverbindungen. Signale und Objekte wurden bis zu diesem Zeitpunkt noch nicht modelliert. Tabelle 8.1 zeigt alle im Testgebiet verwendeten Spurtypen.

Im Rahmen dieser Arbeit wurden die Grundlagendaten für ein detaillierteres Straßenraummodell um Straßenraumobjekte und 3D Geometrien erweitert. Zur Vervollständigung des Spurmodells wur-

Tabelle 8.1 – Extrahierte Spurtypen von Testgebiet 1

OpenDRIVE Spurtyp	3DMS Spurtyp	Beschreibung
driving		normale Fahrspur
sidewalk		Gehweg
parking		Parkfläche
border		Bordstein (abgesenkt, normal, Rampe)
shoulder		befestigter Seitenstreifen
	gutter	Rinnstein
	green	Grünfläche

den durchgezogene und strichlierte Fahrbahnmarkierungen mittels Linienextraktion aus den Punktwolken generiert. Aus den Spurdaten wurden 3D Geometrien erzeugt. Die Objektextraktion erfolgte in der Punktwolkenverarbeitungssoftware Cyclone und in der photogrammetrischen Auswertesoftware 3D RoadView. Aus den Punktwolken wurden Bäume, Straßenlaternen, Pfosten, Ampelpfosten, Schilderpfosten, Verteilerkästen, Blumenkästen, Markierungen (Pfeile, Symbole, Stoplinien) und Fußgängerfurten extrahiert (vgl. Abb. 8.1).

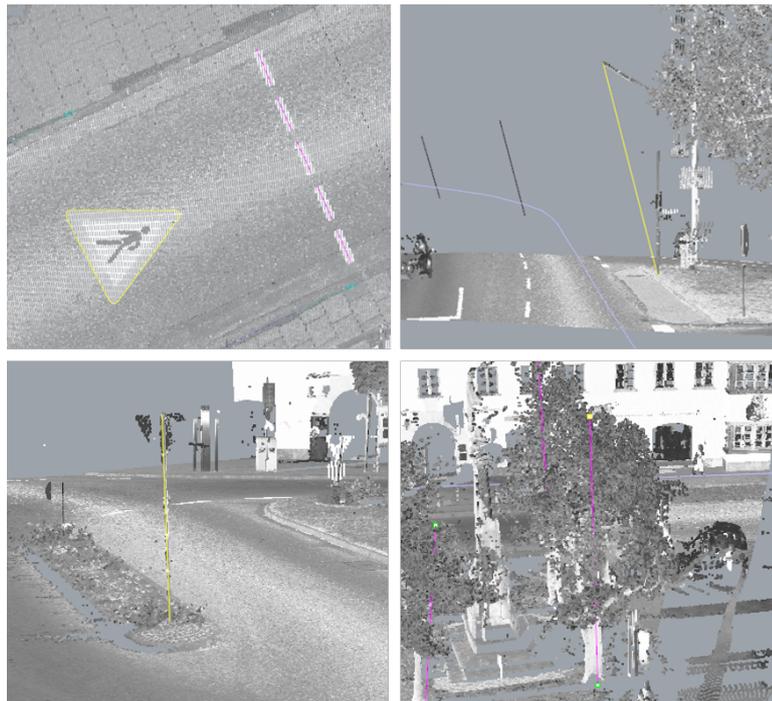


Abbildung 8.1 – Objektextraktion in der Punktwolke (Straßenmarkierungen, Straßenlaterne, Schilderpfosten, Bäume)

Die im Gebiet vorhandenen Signale (Ampeln und Verkehrszeichen) wurden mittels photogrammetrischer Objektmessung aus den Kameradaten gewonnen. In diesem Schritt wurden die Signale auch kategorisiert. Als Vorlage für die Ampeln wurden die Tabelle aus der OpenDRIVE 1.4 Formatspezifikation verwendet. Die Verkehrszeichen wurden nach dem VzKat 2017 der BASt spezifiziert.

Alle extrahierten Objekte werden in Tabelle 8.2 dargestellt. Anschließend wurden alle neu erzeugten Daten in das bestehende CAD-Projekt eingelesen und weiterverarbeitet. Die gesamte Projekt wurden abschließend in das OpenDRIVE-Format konvertiert und zur weiteren Modellierung als DWG-Datei exportiert.

Tabelle 8.2 – Extrahierte Objekte von Testgebiet 1

Objektart	Anzahl	Extraktionsart
Bäume	17	Bodenpunkt mit Höhe und Durchmesser
Pfosten	22	Bodenpunkt mit Höhe und Durchmesser
Ampelpfosten	2	Bodenpunkt mit Höhe und Durchmesser
Schilderpfosten	60	Bodenpunkt mit Höhe und Durchmesser
Straßenlaternen	8	Bodenpunkt mit Höhe, Durchmesser, Orientierung
Parkbänke	2	Umriss
Verteilerkästen	3	Umriss an der Oberkante
Blumenkästen	2	Umriss an der Oberkante
Straßenmarkierungen - Symbole	2	Umriss am Boden
Straßenmarkierungen - Stoplinien	3	Umriss am Boden
Fußgängerfurten	3	Umriss am Boden
Ampeln	7	Mittelpunkt und Kategorie
Schilder	181	Mittelpunkt und Kategorie
Gesamt	312	

8.1.2 Modellierung in FME

Die **Modellierung des Spurmodells** wurde nach den in Unterabschnitt 7.3.1 dargestellten Weiterverarbeitungsschritten durchgeführt. Im Folgenden werden die Ergebnisse dargestellt. Straßenmarkierungen und flächenhafte Objekte am Boden wurden mit der Straßenoberfläche verschnitten (vgl. Abb. 8.2). Die importierten 3D Geometrien wurden zu Volumengeometrien aufbereitet. Für die Darstellung auf Höhe 0 wurden Bordsteine und Gehwege erhöht dargestellt (vgl. Abb. 8.3). Alle Elemente wurden mit entsprechenden Attributen und Texturen angereichert. Dadurch entstand ein texturiertes dreidimensionales Spurmodell auf Höhe 0 und auf absoluter Höhe mit aus den OpenDRIVE- und CityGML-Spezifikationen abgeleiteten Attributen. Zusätzlich wurden die Referenzlinien des Testgebiets mit Attributen angereichert und visualisiert (vgl. Abb. 8.4).

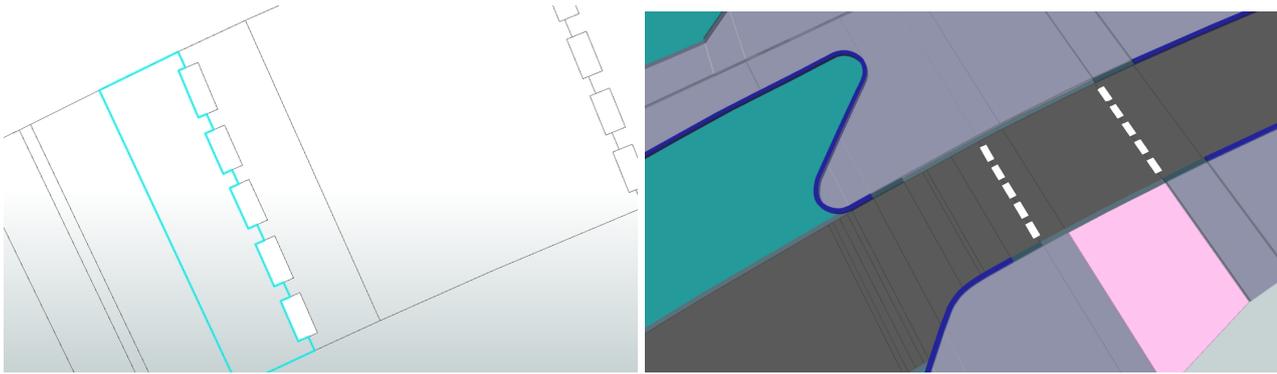


Abbildung 8.2 – Verschneidung flächenhafter Markierungen mit der Fahrbahnoberfläche

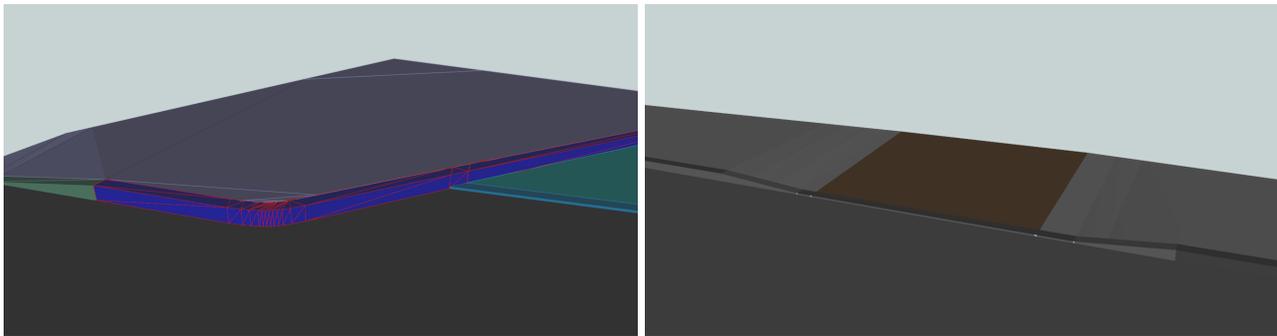


Abbildung 8.3 – Darstellung der Randsteine auf Höhe 0 (links) und auf absoluter Höhe (rechts)

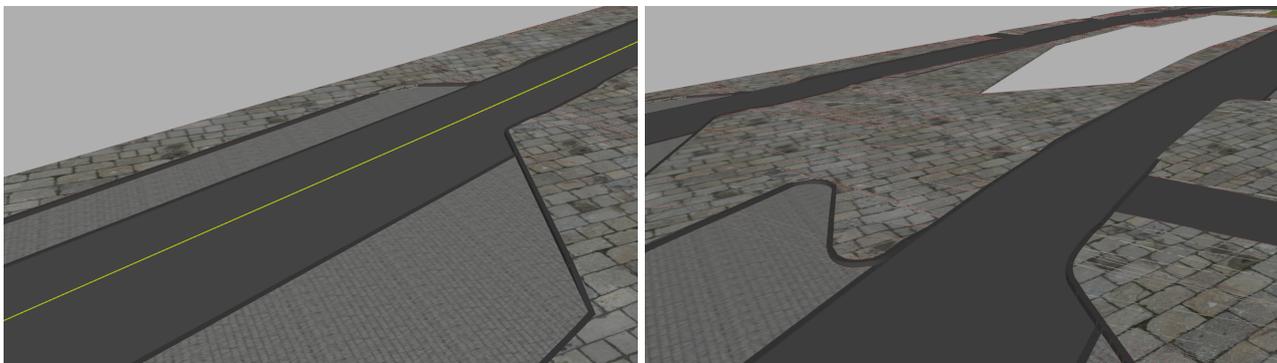


Abbildung 8.4 – Texturiertes Spurmodell auf Höhe 0 mit Referenzlinie (links)

Die Schritte zur **Modellierung der Straßenraumobjekte** wurden in Unterabschnitt 7.3.2 beschrieben. Bäume und Pfosten, die durch Höhe und Durchmesser definiert sind, wurden durch explizite 3D-Modelle visualisiert (vgl. Abb. 8.5). Parkbänke wurden durch die Berechnung der Mittellinie und den Mittelpunkt des Umrisses als 3D-Modelle generiert. Abb. 8.6 zeigt eine positionierte und orientierte Parkbank. Der Bankumriss, der durch die Extraktion in der Punktwolke entsteht, wird in der rechten Abbildung zusätzlich zum Vergleich dargestellt. Für die Darstellung von flächenhaften Objekten wie Fußgängerfurten, Symbole und Stoplinien wurden zwei Visualisierungen implementiert (vgl. Abb. 8.7). Flächenhafte Objekte wurden mit der Fahrbahnoberfläche verschneitten, aber auch extrudiert dargestellt.



Abbildung 8.5 – Darstellung von Bäumen und Pfosten



Abbildung 8.6 – Darstellung von Parkbänken



Abbildung 8.7 – Darstellung von von flächenhaften Objekten auf der Fahrbahn

Im Testgebiet wurden zwei unterschiedliche Arten von Straßenlaternen gemessen (vgl. Abb. 8.8). Die in Unterabschnitt 7.3.2 beschriebene Modellierung wurde für Straßenlaternen mit Ausleger verwendet. Straßenlaternen ohne Ausleger wurden analog zu den Generierungsschritten von Pfosten und Bäumen modelliert.

Objekte mit polygonförmiger Grundfläche wurden als extrudierte Klötzchenmodelle modelliert (vgl. Abb. 8.9) Dazu wurden jeweils die Oberkanten aus den Punktwolken extrahiert.

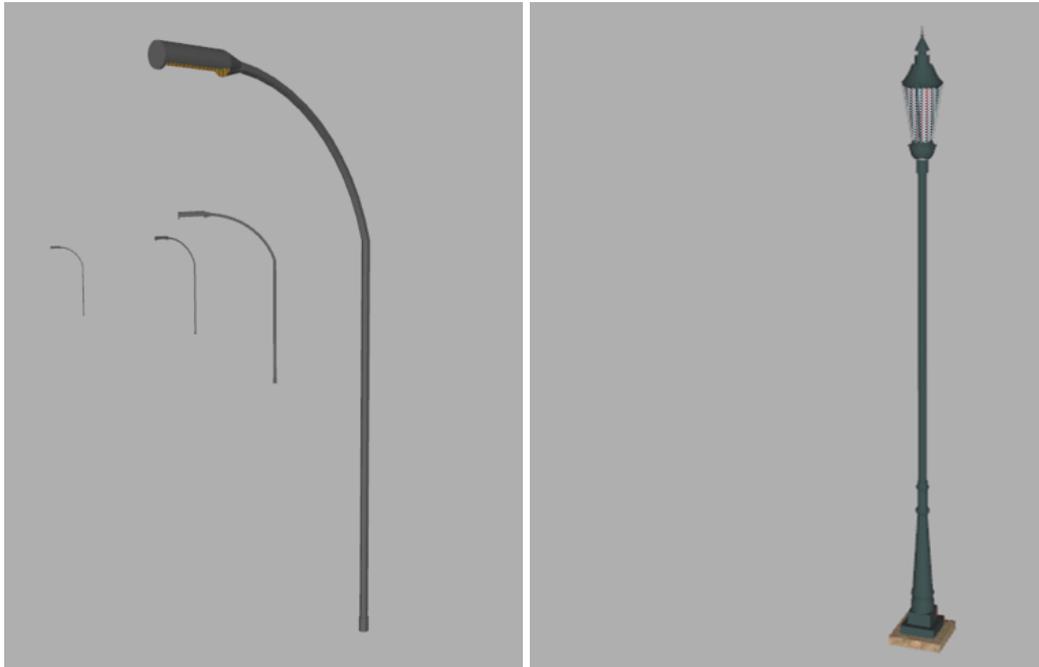


Abbildung 8.8 – Darstellung von Straßenlaternen mit Ausleger (links) und ohne Ausleger (rechts)

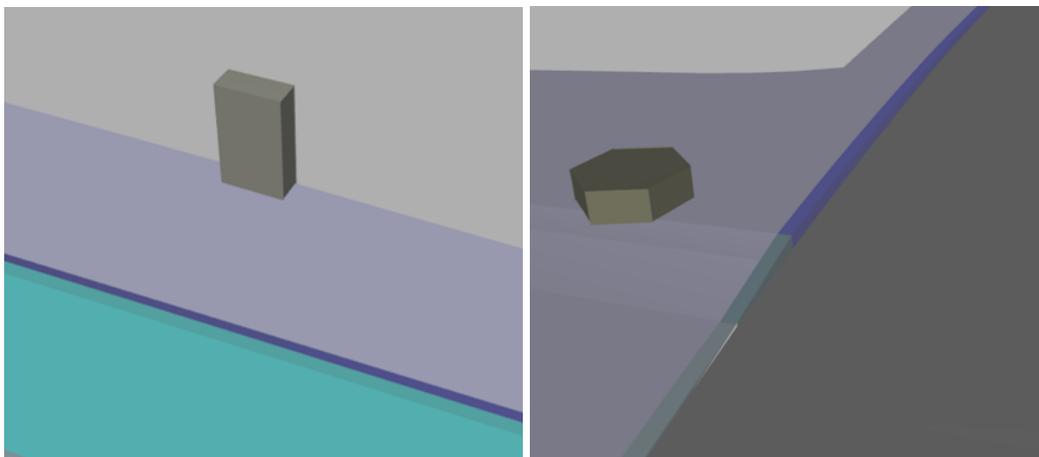


Abbildung 8.9 – Darstellung von Verteilerkästen (links) und Blumenkästen (rechts)

Für die Modellierung der Verkehrsschilder und Ampeln wurden explizite 3D-Modelle generiert. Die Haupt- und Unterkategorie wurde im COLLADA-Dateiname des Modells gespeichert. Für das Testgebiet wurden insgesamt zwei Ampeltypen und 62 verschiedene Verkehrsschilder modelliert. In Abb. 8.10 werden die modellierten, dreidimensionalen Verkehrszeichen und Ampeln dargestellt. Alle verwendeten Signale mit Haupt- und Unterkategorie werden im Anhang B aufgelistet.

Die tatsächlich generierten Attributwerte werden beispielhaft für vier modellierte Objekttypen in Anhang A aufgelistet.

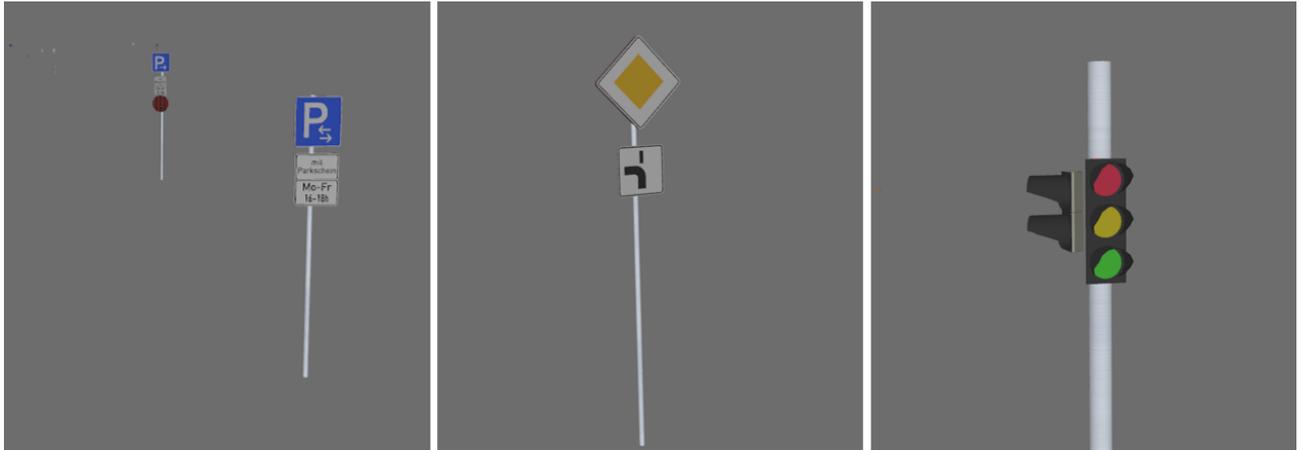


Abbildung 8.10 – Darstellung von Verkehrsschildern (links, mitte) und Ampeln (rechts)

8.1.3 Darstellung der Ergebnisse im Webviewer

Die Darstellung der Ergebnisse im Webviewer wurde analog zu Abschnitt 7.4 realisiert. Die erzeugten CityGML-Dateien wurden aus FME exportiert und in die 3DCityDB importiert. Zusätzlich wurden in diesem Schritt Attributtabelle für die Visualisierung generiert. Für die Darstellung im 3DCityDB Webclient wurde das gesamte Straßenraummodell mit allen Objekten als KML/COLLADA-Format exportiert. Die folgenden Abbildungen zeigen die Visualisierung im Webclient.

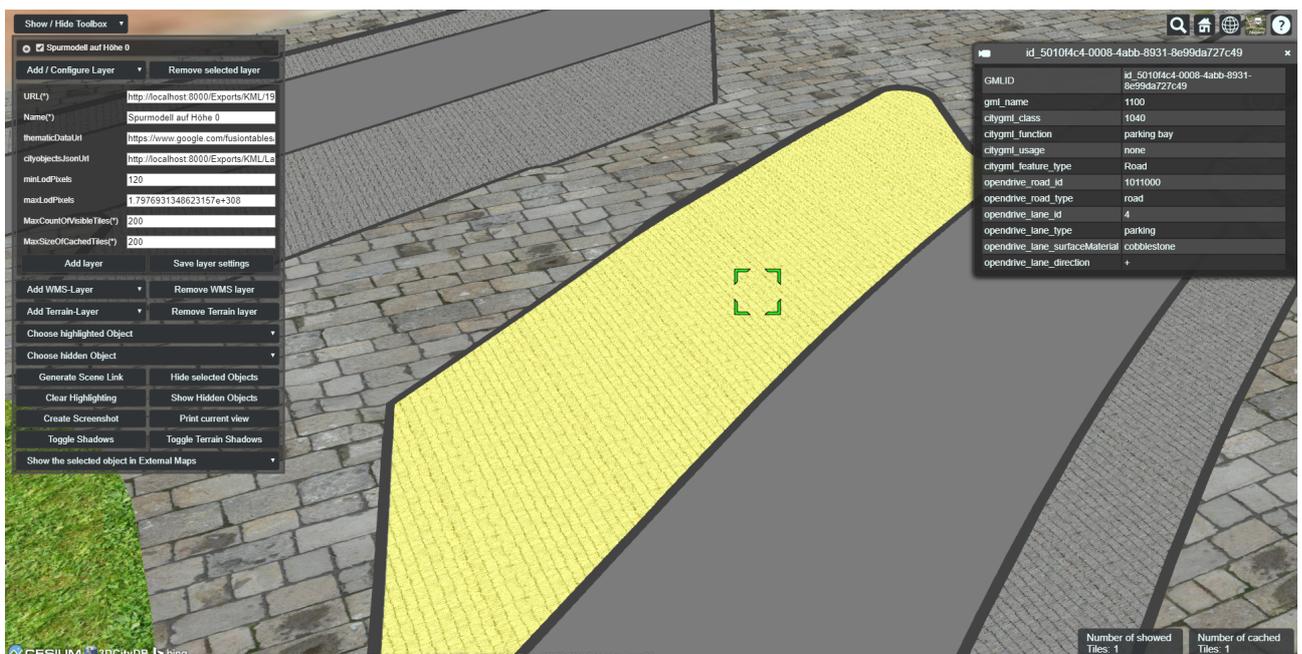
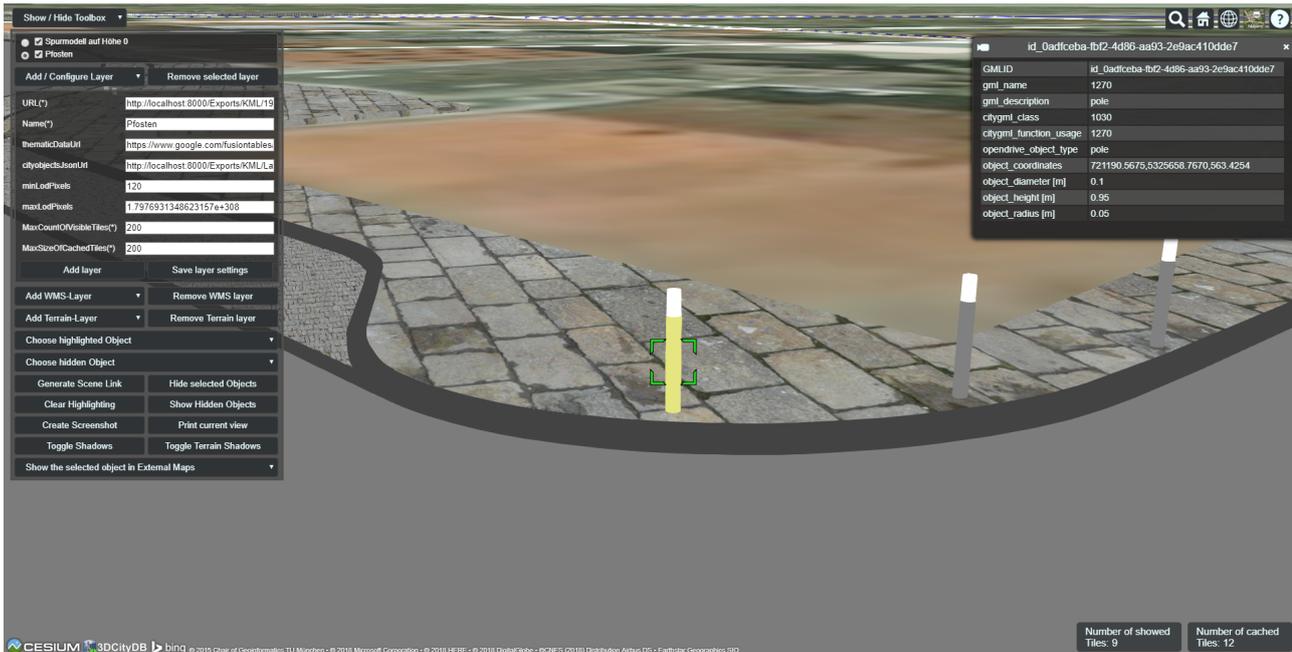


Abbildung 8.11 – Visualisierung der *Transportation*-Objekte im Webclient

Abbildung 8.12 – Visualisierung der *CityFurniture*-Objekte im WebclientAbbildung 8.13 – Visualisierung der *Vegetation*-Objekte im Webclient

Als Beispiele wurden Daten der drei verwendeten Erweiterungsmodule *Transportation* (Abb. 8.11), *CityFurniture* (Abb. 8.12) und *Vegetation* (Abb. 8.13) verwendet. Im linken Bildrand wird die gesamte Layerstruktur mit allen erzeugten Layern dargestellt. Die eingelesenen Attributtabelle erscheinen im rechten Bildrand durch Anklicken des gewünschten Objekts, welches zusätzlich farblich markiert wird.

8.2 Übertragbarkeit auf Testgebiet 2

Die erarbeiteten Prozesse sollen im besten Fall für alle Projekte von 3D Mapping Solutions, die als Endprodukt ein OpenDRIVE-Dokument vorsehen, übertragbar sein. In diesem Kapitel wird aus diesem Grund der erweiterte Erfassungsprozess auf ein zweites Testgebiet übertragen und die Machbarkeit überprüft. Das zweite Projekt befindet sich im Umland von München und wurde auch im Sommer 2017 durch 3D Mapping Solutions vermessen. Insgesamt beträgt die Gesamtlänge der Achsen 3,3 km. Für die Überprüfung des erarbeiteten Konzepts wird lediglich ein kleines Teststück von 150 m verwendet.

8.2.1 Aufbereitung der Grundlagendaten

Die bereitgestellten Daten des zweiten Testgebiets bestehen aus den vorverarbeiteten Rohdaten, dem OpenDRIVE-Dokument und der exportierten DWG-Datei. Die Daten beinhalten ein fertiges Spurmodell mit berechneten 3D Geometrien. Tabelle 8.3 zeigt alle verwendeten Spurtypen.

Tabelle 8.3 – Extrahierte Spurtypen von Testgebiet 2

OpenDRIVE Spurtyp	3DMS Spurtyp	Beschreibung
driving		normale Fahrspur
sidewalk		Gehweg
parking		Parkfläche
border		Bordstein
shoulder		(abgesenkt, normal, Rampe) befestigter Seitenstreifen
	gutter	Rinnstein
	green	Grünfläche

Im Rahmen dieser Arbeit wurden die Daten um Straßenraumobjekte erweitert (vgl. Tabelle 8.4). Die Generierungsschritte erfolgten analog zu Testgebiet 1 bei 3D Mapping Solutions mit der Punkverarbeitungssoftware Cyclone und der Bilderverarbeitungssoftware 3D Road View. Die Daten aus früheren Verarbeitungsschritten wurden zusätzlich zur Weiterverarbeitung als txt-Dateien bereitgestellt. Im bereits bestehenden CAD-Projekt wurden die erzeugten Daten für den OpenDRIVE-Export aufbereitet.

Tabelle 8.4 – Extrahierte Objekte von Testgebiet 2

Objektart	Anzahl	Extraktionsart
Bäume	3	Bodenpunkt mit Höhe und Durchmesser
Ampelpfosten	4	Bodenpunkt mit Höhe und Durchmesser
Schilderpfosten	1	Bodenpunkt mit Höhe und Durchmesser
Straßenlaternen	3	Bodenpunkt mit Höhe, Durchmesser, Orientierung
Parkbänke	2	Umriss
Blumenkästen	1	Umriss an der Oberkante
Straßenmarkierungen - Pfeile	2	Umriss am Boden
Straßenmarkierungen - Stoplinien	3	Umriss am Boden
Ampeln	8	Mittelpunkt und Kategorie
Schilder	6	Mittelpunkt und Kategorie
Gesamt	33	

8.2.2 Überprüfung der Übertragbarkeit

Für die Modellierung des Straßenraums von Testgebiet 2 wurden die für das Testgebiet 1 erarbeiteten FME-Workbenches mit angepassten Eingabe-Dateien verwendet. Für eine automatisierte Erzeugung des Spurmodells wurde eine Unterscheidung aller möglichen Spurtypen durch den Farbwert im Prozess implementiert. Die Generierung des Spurmodells konnte in beiden Höhenmodellen dadurch vollautomatisch durchgeführt werden. Die Annahmen zur Erhöhung einzelner Spurtypen auf Höhe 0 wurde aus Testgebiet 1 übernommen und konnte im Bereich der Gehwege und Bordsteinen zu fehlerfreien Ergebnissen führen. Weitere Annahmen waren durch die Verwendung gleicher Spurtypen in den beiden Testgebieten nicht notwendig. Die Objektarten unterscheiden sich in den beiden Testgebieten nicht, wodurch der Generierungsprozess auch in diesem Fall vollautomatisch erfolgen konnte. Die Übertragbarkeit des Modells konnte durch das zweite Testgebiet bestätigt werden. Die Generierung des Modells und Umwandlung nach CityGML erfolgte automatisch. Nach der Modellierung in FME und der Umwandlung in CityGML konnten die weiteren Verarbeitungsschritte zur Darstellung der Ergebnisse im Webclient übernommen werden.

In Abb. 8.14 werden die Ergebnisse des Spurmodells des zweiten Testgebiets in FME dargestellt. Abb. 8.15 zeigt ein CityFurniture-Objekt auf texturiertem Spurmodell im 3D-Web-Map-Client.

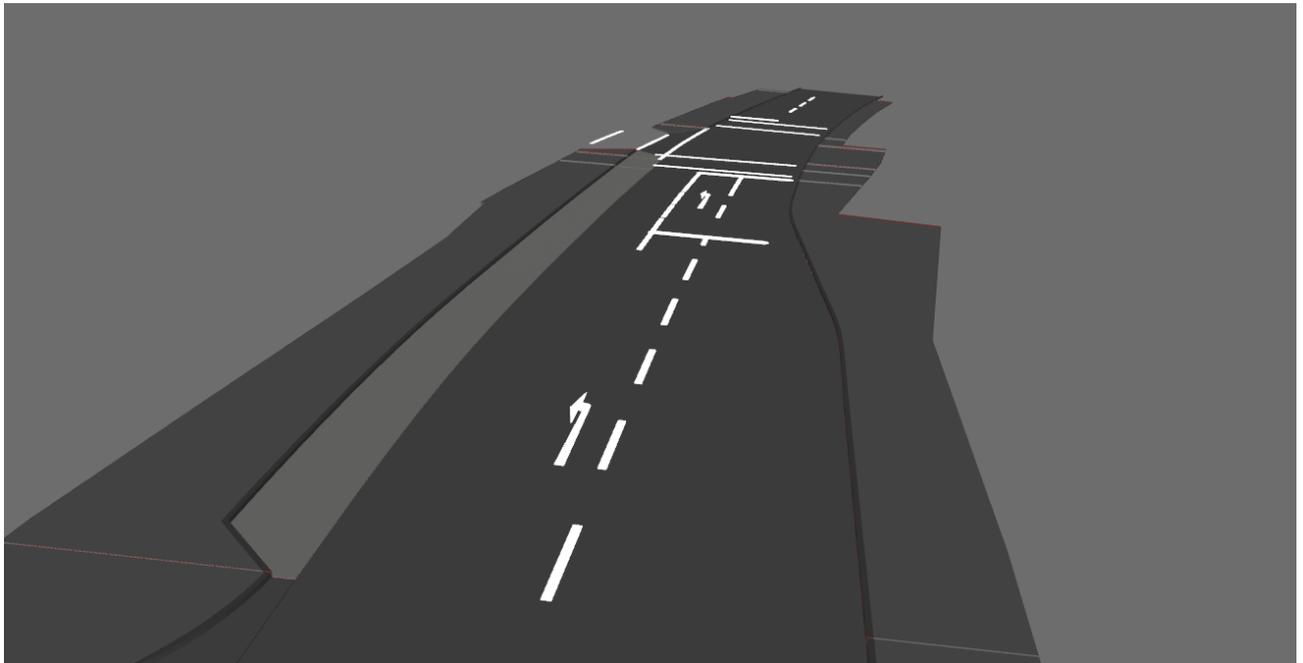


Abbildung 8.14 – Visualisierung des zweiten Testgebiets in FME

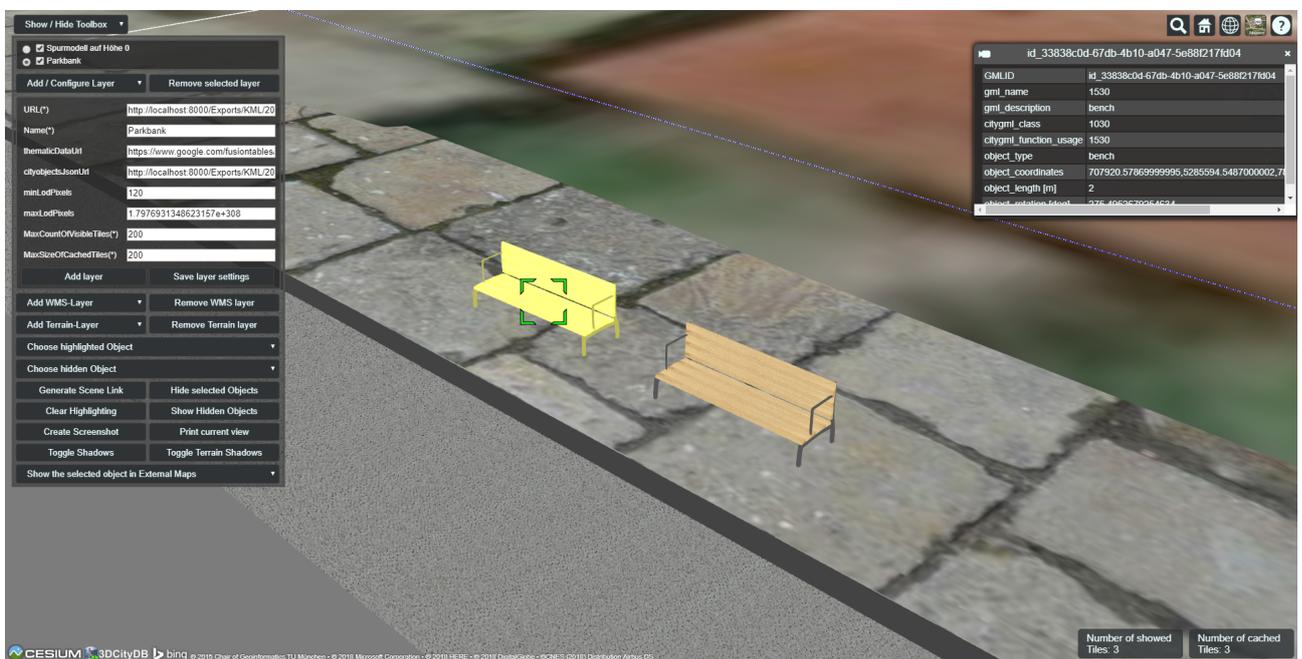


Abbildung 8.15 – Visualisierung eines CityFurniture-Objekts im Webclient

8.3 Kritische Diskussion der Ergebnisse

Die erarbeitete Visualisierungsstrategie aus Kapitel 6 konnte mit den gegebenen Mobile Mapping Daten erfolgreich durchgeführt werden. Die Realisierung erfolgte mittels konkreter Testgebiete. Während das erste Testgebiet in erster Linie der Umsetzung des Konzepts diente, wurde das zweite Testgebiet zur Überprüfung der Übertragbarkeit des Konzepts eingesetzt. Das Ergebnis ist ein mit Attributen angereichertes, semantisches Straßenraummodell mit texturierten Fahrspuren und Objekten.

Das Einlesen des OpenDRIVE-Dokuments in FME ermöglichte die Anreicherung der Referenzlinien mit topologischen Attributen und die Darstellung als LoD0-Network im CityGML-Format.

Mit der Modellierung auf Höhe 0 wurde eine effiziente Darstellung mit kurzer Berechnungszeit umgesetzt. Die in den Grundlagen gegebenen Höhenwerte gehen in diesem Schritt allerdings verloren und müssen für eine dreidimensionale Darstellung über zusätzliche Annahmen ergänzend generiert werden. Die für das erste Testgebiet getroffenen Annahmen konnten für das zweite Gebiet übernommen werden und ermöglichen eine dreidimensionale Darstellung von Gehwegen, Bordsteinen, Park- und Grünflächen. Die Erhöhung von befestigten Seitenstreifen um den festgelegten Wert führte in beiden Testgebieten zu fehlerhaften Darstellungen und in einigen Bereichen zu unerwünschten Stufen im Übergang zwischen benachbarter Spurtypen. Für bisher nicht betrachtete Spurtypen wie Fahrradwege müssen zur Weiterführung des Konzepts zusätzliche Annahmen getroffen werden. Fahrradwege mit direkter Verbindung zur Fahrbahn könnten den Höhenwert der Fahrbahn erhalten. Erhöhte Fahrradwege zwischen Gehwegen und Bordsteinen müssten erhöht dargestellt werden. Zur Implementierung könnten zum Beispiel auch für Fahrradwege Nachbarschaftsbeziehungen abgefragt werden, deren Ergebnisse eine unterschiedliche Erhöhungsart bedeuten würden. Grundsätzlich müssen zur Vermeidung fehlerhafter Darstellungen in anderen Gebieten weiterreichende Annahmen getroffen werden. Ausgeschlossen können Fehler durch diese Modellierungsart allerdings nicht. Diese Fälle werden im Rahmen der Masterarbeit allerdings nicht erprobt und es muss grundsätzlich geprüft werden, ob eine Darstellung auf absoluter Höhe nicht ohnehin mehrere Vorteile mit sich bringen würde. Das Spurmodell auf absoluter Höhe konnte ohne weitere Einstellungen generiert werden.

Die Modellierung des Spurmodells auf tatsächlicher Höhe ist aufgrund der größeren Datenmenge rechenintensiver. Viele in FME implementierte Transformer sind für die Transformation im Zweidimensionalen vorgesehen, wodurch in manchen Fällen eine umständliche Umwandlung der Daten zwischen den Dimensionen erfolgen muss. Die Darstellung der Volumenkörper beruht auf den bei 3D Mapping Solutions generierten 3D-Geometrien. Diese werden im Unternehmen lediglich für Überprüfungszwecke des Spurmodells verwendet und können in manchen Bereichen fehlerbehaftet sein oder Lücken aufweisen. Eine korrekte Texturierung der Geometrieobjekte wird durch die Fehler und Lücken erschwert.

Demnach besitzen beide Darstellungen des Spurmodells Vor- und Nachteile. Nach Anwendungsfall muss entschieden werden, ob eine beispielhafte Darstellung gemessener Straßenraumobjekte auf Höhe 0 zur Präsentation eines Projektes ausreichend ist. Für die Präsentation auf tatsächlicher Höhe müssen die 3D-Geometrien im CAD bestmöglich und fehlerfrei generiert werden.

In dieser Arbeit wurden Kreuzungsbereiche basierend auf dem OpenDRIVE-Standard als einzelne Spuren implementiert. Dies führt an einigen Stellen zu Lücken im Modell. Die Darstellung von Kreuzungsbereichen ist in CityGML nicht genau definiert, wodurch dieser Weg der Modellierung an dieser Stelle gewählt wurde. Für weiterführende Visualisierungen müsste das Problem der lückenhaften Darstellung allerdings gelöst werden.

Beil [2017] schlägt eine Segmentierung des Straßenraums in Abschnitte als eigene Klasse vor, wobei jeder Abschnitt als eigenes *Road*-Objekt mit eindeutiger ID definiert werden könnte. Über eine *parent_id* könnten sämtliche Teilflächen auf das gemeinsame *Road*-Objekt verweisen. Außerdem schlägt er eine Verlinkung der übergeordneten *Road*-Objekte auf alle Teilflächen in Form von XLinks vor. Eine dem Vorschlag ähnliche Segmentierung wird in OpenDRIVE durch die Klassen *Road* und *Junction* definiert. Der Straßenraum wird in Straßen und Kreuzungen eingeteilt, wobei der Kreuzungsbereich beginnt, wenn eine eindeutige Vorgänger- und Nachfolgerbeziehung der Straßenstücke nicht mehr gegeben ist. Die Zuordnung der einzelnen Spuren basiert auf linearer Referenzierung auf übergeordnete Referenzlinien. Auch Kreisverkehre können im OpenDRIVE-Format eindeutig durch die Klasse *JunctionGroup* definiert werden.

Die Kombination beider Modelle wurde in dieser Arbeit nicht erprobt, lässt aber Spielraum für folgende Arbeiten. Das Spurmodell von OpenDRIVE könnte weiterhin als Grundlage zur Implementierung von Kreuzungen in CityGML verwendet werden. Zusätzlich könnten für eine lückenlose Darstellung übergeordnete flächenhafte Kreuzungsbereiche mit eigener ID generiert werden. Die Zuordnung der Teilflächen bzw. Spuren zu den Kreuzungsbereichen könnte nach dem Vorschlag von Beil [2017] in CityGML umgesetzt werden.

Grundsätzlich ist mit den erzeugten Grundlagendaten eine dreidimensionale Modellierung unterschiedlicher Straßenraumobjekte möglich. Die bereitgestellten Daten enthalten zusätzlich ausreichend Attribute zur semantischen Anreicherung des Modells. Das Konzept beinhaltet eine Unterscheidung der Objekte nach Art der Extraktion, wie in Tabelle 7.3 dargestellt, wodurch eine Übertragung auf andere Objekte ermöglicht wird. Die Objekte wurden als explizite 3D-Geometriemodelle in FME implementiert. Die Berechnungszeiten fielen durch die geringe Datenmenge relativ kurz aus. Eine Generierung impliziter Geometriemodelle ist im CityGML-Format möglich und ist vor allem für eine effiziente Darstellung größerer Datenmengen notwendig [Cantzler, 2015].

Die Verarbeitung der Verkehrszeichenmodelle war trotz verwendeter 3D-Modelle aus dem 3D SketchUp-Warehouse sehr zeitintensiv. Die Modelle liegen dort meist in Gruppen vor, wodurch benötigte Verkehrszeichen zuerst extrahiert und mit passenden Attributen angereichert werden mussten. Insgesamt wurde für diese Arbeit ein Verkehrszeichenkatalog mit 62 Verkehrszeichen erzeugt, die mit den Kategorien des VzKat der BASt versehen wurden. In Deutschland wird allerdings zwischen nahezu 600 Verkehrszeichen unterschieden. Für weiterreichende Visualisierungen muss der generierte Katalog mit 3D-Modellen erweitert werden.

Pfosten wurden in dieser Arbeit durch den Bodenpunkt und den höchsten Punkt am Objekt charakterisiert. Der höchste Punkt diente lediglich der Berechnung der Objekthöhe. Die Objekte selbst wurden als gerade angenommen, wodurch schief stehende Pfosten nicht beachtet wurden. Dies führt

in Verbindung mit anderen Objekten wie zum Beispiel der Verknüpfung von Verkehrszeichen und zugehörigen Pfosten zu Darstellungsfehlern.

Die Konvertierung der Daten in das CityGML-Format konnte durch bestehende Writer problemlos in FME umgesetzt werden. Die gesamte Berechnungszeit der einzelnen Modelle mit der Umwandlung in CityGML liegt lediglich zwischen 1-2 Minuten. Das aktuelle CityGML 2.0-Transportmodell lässt lediglich die Geometrien *gml:lod0Network* oder *gml:MultiSurface* zu, wodurch generierte Volumenkörper wie Bordsteine und Gehwege vor der Umwandlung in ihre begrenzenden Randflächen zerlegt werden mussten. Dieses Problem hat auch Beil [2017] dargestellt und schlägt zur volumenhaften Darstellung eine Erweiterung des Modells um die Geometrie *gml:MultiSolid* vor.

Im 3DCityDB Web-Map-Client wurde das generierte Spurmodell auf Höhe 0 mit allen beschriebenen Objekten visualisiert. Allgemein geht jedoch viel Information über den Austausch als DWG-Datei verloren. In den meisten Fällen ist die Generierung in FME nur sehr umständlich umsetzbar und kann zusätzlich zu Darstellungsfehlern führen. Dieser Schritt kann vermieden werden, wenn der gesamte erweiterte Visualisierungsprozess intern bei 3D Mapping Solutions verwendet wird und so ein Zugriff auf alle für das OpenDRIVE-Dokument generierten Daten möglich ist. Grundsätzlich ist die Erweiterung des Erfassungsprozesses bei 3D Mapping Solutions mit wenig Aufwand möglich, da die Softwarepakete der 3DCityDB mit dem Web-Map-Client frei zugänglich sind. Die bei 3D Mapping Solutions erfassten Daten können zur Erweiterung und Verbesserung des Konzepts beitragen.

9 Fazit und Ausblick

Im Rahmen dieser Arbeit konnten durch Mobile Mapping erfasste Spur- und Objektdaten ein virtuelles semantisches CityGML-konformes Straßenraummodell erzeugt werden. Das detaillierte Straßenraummodell liegt texturiert im LoD 3 vor und besteht unter anderem aus Fahrspuren, Grünflächen, Parkflächen, Verkehrszeichen und Bäumen. Anhand konkreter Testgebiete konnte die erarbeitete Strategie zur Modellierung des Straßenraums realisiert und eine detaillierte Visualisierung der MMS-Daten ermöglicht werden.

Mit einer Studie relevanter den Straßenraum betreffender Arbeiten wurden verschiedene Anwendungsbereiche, Methoden der Datenextraktion und Datenquellen im Bereich der Straßenraummodellierung aufgezeigt. Anhand verschiedener Kriterien wie Verfügbarkeit, Vollständigkeit und semantischer Informationsgehalt konnten die Datenquellen beurteilt werden. Daraus geht hervor, dass in den meisten Fällen eine Kombination verschiedener Datenquellen, wie sie bei Mobil Mapping Systemen eingesetzt werden, verwendet werden. Dies bringt eine Verknüpfung verschiedenster Daten mit sich und führt zu hochgenauen und semantisch detailreichen Ergebnissen.

Insbesondere wird in dieser Arbeit vergleichend auf die Modellierungsstandards OpenDRIVE und CityGML eingegangen. Obwohl die XML-basierten Standards grundsätzlich beide für die Modellierung des Straßenraums verwendet werden können, weisen die Datenmodelle große Unterschiede auf. Der Vergleich der beiden Datenmodelle in Inhalt, Struktur, Geometrie, Semantik, Topologie und visueller Erscheinung verdeutlicht die Unterschiede, zeigt aber auch dass eine Abbildung von OpenDRIVE nach CityGML grundsätzlich mit einigen Abwandlungen möglich ist. Grundlegender Unterschied ist die geometrische Beschreibung der Objekte. Während diese in OpenDRIVE mittels linearer Referenzierung und aufwendigen mathematischen Funktionen erfolgt, werden Geometrieobjekte in CityGML flächenhaft mittels absoluten Koordinaten und Primitiven beschrieben. Die in OpenDRIVE erlaubten Geometrien könnten grundsätzlich auch für CityGML implementiert werden, da sie in der ISO 19107 enthalten sind. Semantische Unterschiede können durch die Verwendung generischer Attribute in CityGML ausgeglichen werden. Trotz allem konnte die Übertragbarkeit von einigen Teilen der Datenmodelle dargelegt werden. Die an den Vergleich anschließende semantische Zuordnung einzelner Attributwerte zeigt einen möglichen Weg der semantischen Überführung von OpenDRIVE nach CityGML und dient als Grundlage für die Umwandlung der für das OpenDRIVE-Format aufbereiteten vorliegenden Grundlagendaten.

Die abschließende Visualisierung im auf Cesium basierenden Web-Map-Client der 3DCityDB ermöglicht die Verknüpfung von detaillierten, hochgenauen Straßendaten und einem virtuellen 3D-Gebäudemodell.

Die Straßenraumdaten gewinnen durch die realitätsnahe Präsentation zusätzlich deutlich an Anschaulichkeit. Über die Visualisierung hinausgehend sind mithilfe des Clients Abfragen und somit den Straßenraum betreffende Analysen, wie beispielsweise die Selektion von Straßen mit einem vorgegebenen Tempolimit, möglich. Ein detailliertes, räumlich-semantisches Modell des Straßenraums eröffnet zahlreiche weitere Anwendungen jenseits der Fahrsimulation und Navigation und ist für die Präsentation der Ergebnisse und Kundenakquise von Vorteil.

Das erzeugte Straßenraummodell kann als Grundlage für weiterführende Visualisierungen dienen und durch die semantische Anreicherung weiterer Attribute und die Implementierung anderer Straßenraumobjekte erweitert werden. Durch Ergänzung des erarbeiteten Ansatzes im Erfassungsprozess von 3D Mapping Solutions, ist eine automatische Generierung eines semantischen, dreidimensionalen Straßenraummodells in CityGML und die Darstellung des Modells im Web-Map-Client als zusätzliches Nebenprodukt möglich. Anhand bestehender Projekte könnte die Implementierung weiter erprobt und ausgebaut werden. Für eine effiziente Darstellung größerer Datenmengen wird eine Implementierung impliziter Geometrien vorgeschlagen, welche den Rechenaufwand dreidimensionaler Objekte deutlich reduziert. Die Darstellung der Straßenraumdaten Web-Map-Client erleichtert die Anreicherung mit zusätzlichen 3D-Gebäudemodellen und kann auf diesem Weg eine realitätsnahe Visualisierung eines 3D-Stadtmodells ermöglichen. Der Vergleich der Datenmodelle von OpenDRIVE und CityGML kann auch zukünftig für weiterführende Analysen der beiden Standards und der Weiterentwicklung beider Modelle dienen. Grundsätzlich wurden in dieser Arbeit Möglichkeiten zur Visualisierung von mittels Mobile Mapping aufgenommenen Straßendaten geschaffen.

Danksagung

Mein besonderer Dank gilt Herrn Professor Thomas H. Kolbe für die Betreuung dieser Arbeit und die wertvollen Anregungen und Ratschläge in den Gesprächsrunden. Ich möchte auch Herrn Christof Beil und Frau Caroline Marx für konstruktive Anregungen und Hilfestellungen danken. Ein spezielles Dankeschön vor allem an das gesamte Team von 3D Mapping Solutions, insbesondere an Frau Christiane Radies, für die kontinuierliche Unterstützung während der gesamten Bearbeitungszeit.

Literaturverzeichnis

3D Mapping Solutions GmbH, 2017. URL <http://www.3d-mapping.de/>.

Jörg Albertz. *Einführung in die Fernerkundung: Grundlagen der Interpretation von Luft- und Satellitenbildern*. Wissenschaftliche Buchgesellschaft, Darmstadt, 2001.

Julian Amann. Einbettung von prozeduralem Wissen in Produktdatenmodelle am Beispiel der Richtlinien für die Anlage von Landstraßen mithilfe der Programmiersprache IFCPL. In *Proceedings of the 27th Forum Bauinformatik*, Sep 2015.

Julian Amann, André Borrmann, T. Chipman, E. Lebegue, T. Liebich, und P. Scarponcini. IFC alignment project, process map and use cases (informative), 2015. URL <http://www.buildingsmart-tech.org/downloads/ifc/ifc5-extension-projects/ifc-alignment>.

Emmanuel P. Baltsavias. A comparison between photogrammetry and laser scanning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(2-3):83–94, 1999.

David Barber, Jon Mills, und Sarah Smith-Voysey. Geometric validation of a ground-based mobile laser scanning system. *ISPRS Journal of Photogrammetry and Remote Sensing*, 63(1):128–141, 2008.

BAST. Objektkatalog für das Straßenwesen Version 2.017, 2016. URL <http://www.okstra.de/docs/2017>.

Bayerische Vermessungsverwaltung. Geodaten Online, 2018. URL <https://geoportal.bayern.de/geodatenonline/>.

Christof Beil. Detaillierte Repräsentation des Straßenraums in 3D-Stadtmodellen. Masterarbeit, Technische Universität München, 2017.

Christof Beil und Thomas H. Kolbe. CityGML and the streets of new york – A proposal for detailed street space modelling. In *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 9–16, 2017.

Filip Biljecki, Jantien Stoter, Hugo Ledoux, Sisi Zlatanova, und Arzu Çöltekin. Applications of 3D city models: State of the art review. *ISPRS International Journal of Geo-Information*, 4(4):2842–2889, 2015.

- Claus Brenner. Extraction of features from mobile laser scanning data for future driver assistance systems. In *Advances in GIScience*, pages 25–42. Springer, 2009.
- Barbara Burger. Erzeugung eines semantischen 3D-Stadtmodells der Stadt New York auf der Basis von Open Data – DGM, Straßen und Zonierung. Masterarbeit, Technische Universität München, München, 2015.
- Berit Cantzler. Erzeugung eines semantischen 3D-Stadtmodells der Stadt New York auf der Basis von Open Data – DGM, Gebäude, Flurstücke, Gewässer, Parks und Bäume. Masterarbeit, Technische Universität München, München, 2015.
- Julien Chaplier, Thomas Nguyen That, Marcus Hewatt, und Gilles Gallee. Toward a standard: RoadXML, the road network database format. *Actes INRETS*, pages 211–220, 2010.
- Liang-Chien Chen und Chao-Yuan Lo. 3D road modeling via the integration of large-scale topomaps and airborne LiDAR data. *Journal of the Chinese Institute of Engineers*, 32(6):811–823, 2009.
- Liang-Chien Chen, Chao-Yuan Lo, Yi-Chen Shao, und Tee-Ann Teo. Automatic reconstruction of 3D road models by using 2D road maps and airborne LiDAR data. In *Proceedings of 27th Asian Conference on Remote Sensing (ACRS2006)*, pages 9–13, 2006.
- Simon Clode, Peter J Kootsookos, und Franz Rottensteiner. The automatic extraction of roads from LiDAR data. In *The International Society for Photogrammetry and Remote Sensing's Twentieth Annual Congress*, volume 35, pages 231–236. ISPRS, 2004.
- con terra GmbH. *FME Desktop: Das deutschsprachige Handbuch für Einsteiger und Anwender*. con terra GmbH, 2015.
- Pete Doucette, Peggy Agouris, und Anthony Stefanidis. Automated road extraction from high resolution multispectral imagery. *Photogrammetric Engineering & Remote Sensing*, 70(12):1405–1416, 2004.
- Philippe Ducloux, Julien Chaplier, Guillaume Millet, und Gilles Gallee. RoadXML 2.4.1 – Road network description, XML format specification version: 2.4.1, 2009.
- Marius Dupuis, Martin Strobl, und Hans Grezlikowski. OpenDRIVE 2010 and beyond – Status and future of the de facto standard for the description of road networks. In *Proceedings of the Driving Simulation Conference DSC Europe 2010*, 2010.
- Marius Dupuis, Mohamman Bahram, Hans Grezlikowski, Andreas Richter, Alex Goldberg, Ekkehard Klaerner, Dr. Bernhard Bock, und Laban Kaellgren. OpenDRIVE – Format specification, rev. 1.4. VIRES Simulationstechnologies GmbH, 2015.
- Naser El-Sheimy. An overview of mobile mapping systems. In *FIG Working Week*, pages 16–21, 2005.
- Cameron Ellum und Nasser El-Sheimy. Land-based mobile mapping systems. *Photogrammetric engineering and remote sensing*, 68(1):13–17, 2002.

- Gunnar Gräfe, Wilhelm Caspary, Hans Heister, Jochen Klemm, und Manfred Sever. The road data acquisition system MoSES—determination and accuracy of trajectory data gained with the applanix POS/LV. In *Proceedings, The Third International Mobile Mapping Symposium*, pages 3–5, 2001.
- Gerhard Gröger, Thomas H. Kolbe, Claus Nagel, und Karl-Heinz Häfele. OGC city geography markup language CityGML encoding standard, OGC 12-01, 2012.
- Hans-Christoph Gruler, Erik Stubkjaer, Peter Axelsson, und Lars Wikstrom. OGC land and infrastructure conceptual model standard (LandInfra), OGC 16-07, 2016.
- Haiyan Guan, Jonathan Li, Yongtao Yu, Cheng Wang, Michael Chapman, und Bisheng Yang. Using mobile laser scanning data for automated extraction of road markings. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87:93–107, 2014.
- Alexander Haring. *Die Orientierung von Laserscanner- und Bilddaten bei der fahrzeuggestützten Objekterfassung*. Dissertation, Technischen Universität Wien, 2007.
- Carsten Hatger. On the use of airborne laser scanning data to verify and enrich road network features. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 36, pages 138–143, 2005.
- Carsten Hatger und Claus Brenner. Extraction of road geometry parameters from laser scanning and existing databases. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 34(3/W13):225–230, 2003.
- Jochen Hettwer. Objektkatalog für das Straßen- und Verkehrswesen – OKSTRA. In Buhmann/Pietsch/Heins, editor, *Digital Design in Landscape Architecture*, pages 234–239. Wichmann Verlag, Heidelberg, 2008.
- Xiangyun Hu, Chao V. Tao, und Yong Hu. Automatic road extraction from dense urban area by integrated processing of high resolution imagery and LiDAR data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 35:B3, 2004.
- ISO 14825. Intelligente Transportsysteme - Geographische Dateien (GDF) - GDF5.0 (ISO 14825:2011). Normenausschuss Automobiltechnik, 2011.
- Anttoni Jaakkola, Juha Hyyppä, Antero Kukko, Xiaowei Yu, Harri Kaartinen, Matti Lehtomäki, und Yi Lin. A low-cost multi-sensoral mobile mapping system and its feasibility for tree measurements. *ISPRS journal of Photogrammetry and Remote Sensing*, 65(6):514–522, 2010.
- Yang Ji. *Durchgängige Trassen- und Brückenplanung auf Basis eines integrierten parametrischen 3D-Infrastrukturbauproduktmodells*. Dissertation, Technische Universität München, 2014.
- Yang Ji, Katharina Lukas, Mathias Obergrießer, und André Borrmann. Entwicklung integrierter 3D-Trassenproduktmodelle für die Bauablaufsimulation. In *Tagungsband des 20. Forum Bauinformatik, Dresden*, 2008.

- Harri Kaartinen, Juha Hyyppä, Antero Kukko, Anttoni Jaakkola, und Hannu Hyyppä. Benchmarking the performance of mobile laser scanning systems using a permanent test field. *Sensors*, 12(9): 12814–12835, 2012.
- Thomas H. Kolbe, Zhihang Yao, Claus Nagel, Richard Redweik, Philipp Willkomm, György Hurda, Arda Müftüoğlu, und Felix Kunde. 3D city database for CityGML version 3.3.0 documentation 2016. 2016. URL www.3dcitydb.org.
- Pankaj Kumar, Paul Lewis, und Conor P. McElhinney. Parameteric analysis for automated extraction of road edges from mobile laser scanning data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(2):215, 2015.
- LDBV. Die amtlichen Geobasisdaten der Bayerischen Vermessungsverwaltung. *Landesamt für Digitalisierung, Breitband und Vermessung*, 2018a.
- Breitband und Vermessung LDBV, Landesamt für Digitalisierung. Produkte des LDBVs, 2018b. URL <https://www.ldbv.bayern.de/produkte/>.
- Mathias Lemmens. Mobile laser scanning point clouds – Status and prospects of automatic 3D mapping of road objects. *GIM International*, 2017.
- Jesse Levinson, Michael Montemerlo, und Sebastian Thrun. Map-based precision vehicle localization in urban environments. *Robotics Science and Systems*, 2007.
- Ilka May. *Fortführung und Erweiterung von GDF (Geographic Data File) für Autonavigationssysteme*. Dissertation, Institut für Geographie der Technischen Universität Berlin, 2002.
- Joel C. McCall und Mohan M. Trivedi. Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation. *IEEE Transactions on Intelligent Transportation Systems*, 7(1): 20–37, 2006.
- Rohini Narwade und Vijaya Musande. Automatic road extraction from airborne LiDAR: A review. *Int. Journal Engineering Research and Applications*, 4(12):54–62, 2014.
- Stephan Nebiker, Kay Kamer, und Hannes Eugster. Integrierte Georeferenzierung von luft- und bodengestützten Stereobilddaten im Straßenbereich – Strategien und Ergebnisse. *DGPF Tagungsband 22 / 2013, Dreiländertagung DGPF, OVG, SGPF*, 2013.
- Pascal Neis, Dennis Zielstra, Alexander Zipf, und Alexander Strunck. Empirische Untersuchungen zur Datenqualität von OpenStreetMap-Erfahrungen aus zwei Jahren Betrieb mehrerer OSM-Online-Dienste. *Angewandte Geoinformatik*, 2010, 2010.
- Julia Offer. Automatisierte Generierung eines Stadtmodells für Straßenbahnsimulatoren. Masterarbeit, Technische Universität München, 2016.
- OKFD. Offene Daten. *Open Knowledge Foundation Germany Version:2016*, 2016. URL <https://okfn.de/themen/offene-daten/>.

- OpenStreetMap. OpenStreetMap – Deutschland, 2018a. URL <https://www.openstreetmap.de/>.
- OpenStreetMap. OpenStreetMapWiki, 2018b. URL <https://wiki.openstreetmap.org/wiki/>.
- Gordon Petrie. An introduction to the technology: Mobile mapping systems. *Geoinformatics*, 13(1):32, 2010.
- Shi Pu, Martin Rutzinger, George Vosselman, und Sander Oude Elberink. Recognizing basic structures from mobile laser scanning data for road inventory studies. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(6):S28–S39, 2011.
- Xiaozhi Qu, Bahman Soheilian, und Nicolas Paparoditis. Vehicle localization using mono-camera and geo-referenced traffic signs. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, pages 605–610, June 2015.
- Roland Ruhdorfer. Kopplung von Verkehrssimulation und semantischen 3D-Stadtmodellen in CityGML. Masterarbeit, Technische Universität München, 2017.
- Paul Scarponcini. LandInfra executive summary, OGC project document 16-000, 2016.
- Markus Schreiber, Carsten Knoepfel, und Uwe Franke. LaneLoc: Lane marking based localization using highly accurate maps. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 449–454, June 2013.
- Klaus-Peter Schwarz und Naser El-Sheimy. Kinematic multi-sensor systems for close range digital imaging. *International Archives of Photogrammetry and Remote Sensing*, 1996.
- Klaus-Peter Schwarz und Naser El-Sheimy. Digital mobile mapping systems – State of the art and future trends. In *Advances in Mobile Mapping Technology*, pages 19–34. CRC Press, 2007.
- Orvar Segerström. Automating 3D graphics generation using GIS data – Terrain and road reproduction. Masterarbeit, Umea University, Department of Computing Science, 2015.
- G. Siegle, J. Geisler, F. Laubenstein, H. H. Nagel, und G. Struck. Autonomous driving on a road network. In *Proceedings of the Intelligent Vehicles '92 Symposium*, pages 403–408, Jun 1992.
- Ildiko Suveg und George Vosselman. Reconstruction of 3D building models from aerial images and maps. *ISPRS Journal of Photogrammetry and remote sensing*, 58(3-4):202–224, 2004.
- C Vincent Tao und Jonathan Li. *Advances in mobile mapping technology*, volume 4. CRC Press, 2007.
- Chao V. Tao. Mobile mapping technology for road network data acquisition. *Journal of Geospatial Engineering*, 2(2):1–14, 2000.
- Chao V. Tao, Michael A. Chapman, und Bruce A. Chaplin. Automated processing of mobile mapping image sequences. *ISPRS Journal of Photogrammetry and Remote Sensing*, 55(5-6):330–346, 2001.

- Stefan Wanasky. Prototypische Objekte bei 3D-Stadtmodellen am Beispiel des Standards CityGML. Masterarbeit, Technische Universität München, Eidgenössische Technische Hochschule Zürich, 2011.
- Tao Wu und Ananth Ranganathan. Vehicle localization using road markings. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 1185–1190, June 2013.
- Manohar Yadav, Arshad Husain, Ajai K. Singh, und Bharat Lohani. Pole-shaped object detection using mobile LiDAR data in rural road environments. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2:11, 2015.
- Bisheng Yang, Lina Fang, Qingquan Li, und Jonathan Li. Automated extraction of road markings from mobile LiDAR point clouds. *Photogrammetric Engineering & Remote Sensing*, 78(4):331–338, 2012.
- Bisheng Yang, Lina Fang, und Jonathan Li. Semi-automated extraction and delineation of 3D roads of street scene from mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 79:80–93, 2013.
- Bisheng Yang, Zhen Dong, Yuan Liu, Fuxun Liang, und Yongjun Wang. Computing multiple aggregation levels and contextual features for road facilities recognition using mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 126:180–194, 2017.
- Zhihang Yao und Thomas H. Kolbe. Cloud-basierter Webclient für kollaboratives Arbeiten auf semantischen 3D-Stadtmodellen. In Eckhardt Seyfert, Eberhard Gülch, Christian Heipke, Jochen Schiewe, und Monika Sester, editors, *Geoinformationen Öffnen das Tor zur Welt, 34. Wissenschaftlich-Technische Jahrestagung der DGPF*, Publikationen der Deutschen Gesellschaft für Photogrammetrie, Fernerkundung, Geoinformation e.V., Hamburg, 2014.
- Hiroki Yokoyama, Hiroaki Date, Satoshi Kanai, und Hiroshi Takeda. Detection and classification of pole-like objects from mobile laser scanning data of urban environments. *International Journal of Cad/cam*, 13(2):31–40, 2013.
- Chunsun Zhang. Towards an operational system for automated updating of road databases by integration of imagery and geodata. *ISPRS Journal of Photogrammetry and Remote Sensing*, 58(3):166 – 186, 2004. ISSN 0924-2716.
- Liang Zhou und George Vosselman. Mapping curbstones in airborne and mobile laser scanning data. *International Journal of Applied Earth Observation and Geoinformation*, 18:293–304, 2012.

Abbildungsverzeichnis

1.1	Überblick über den Datenauswertungs- und Modellierungsprozess bei 3D Mapping Solutions (Bildquellen: 3D Mapping Solutions)	4
1.2	Visualisierung des semantischen 3D-Straßenraummodells der Stadt New York [Beil, 2017]	5
2.1	Aufbau eines Straßenraummodells bestehend aus vier Ebenen [Ruhdorfer, 2017] . . .	9
3.1	Track- und Inertial-Koordinatensystem in OpenDRIVE [Dupuis et al., 2015, S.13] . . .	23
3.2	Grundprinzip des Straßenverlaufs in OpenDRIVE [Dupuis et al., 2015, S.18]	24
3.3	Fahrspuren in OpenDRIVE [Dupuis et al., 2015, S.20]	25
3.4	Straßenabschnitte in OpenDRIVE [Dupuis et al., 2015, S.21]	26
3.5	Querschnitte mit Überhöhung (links) und Querneigung (rechts) in OpenDRIVE [Dupuis et al., 2015, S.23]	27
3.6	Querprofile in OpenDRIVE [Dupuis et al., 2015, S.24]	27
3.7	Straßenverbindungsmöglichkeiten in OpenDRIVE [Dupuis et al., 2015, S.25]	28
3.8	UML-Diagramm des CityGML-Kernmodell [Gröger et al., 2012, S.50]	30
3.9	Die fünf LoD in CityGML (Graphik: IGG Uni Bonn) [Gröger et al., 2012, S.11]	31
3.10	Repräsentationen der Klasse TransportationComplex - Von links nach rechts: Road, Track, Rail, Square (Graphik: Rheinmetall Defence Electronics) [Gröger et al., 2012, S.124]	32
3.11	UML-Diagramm des CityGML-Transportmodell [Gröger et al., 2012, S.123]	33
3.12	Unterschiedliche LoD des TransportationComplex (Graphik: Rheinmetall Defence Electronics) [Gröger et al., 2012, S.126]	33
3.13	UML-Diagramm eines weiterentwickelten CityGML-Transportmodells [Beil, 2017 S. 52]	34
3.14	Repräsentation eines Straßenstücks nach vorgeschlagenem LoD-Konzept [Beil, 2017 S. 53]	35
3.15	UML-Diagramm der CityFurniture in CityGML [Gröger et al., 2012, S.138]	36
3.16	Rekursive Aggregation von Objekten und Geometrien in CityGML (Graphik: IGG Uni Bonn) [Gröger et al., 2012, S.27]	37
4.1	Aufbau des Mobile Mapping Systems MoSES (Bildquelle: 3D Mapping Solutions) . . .	43
4.2	Erfassungsprozess bei 3D Mapping Solutions	46
4.3	Darstellung der 2D Schraffuren (links) und 3D Geometrien in der Punktwolke (rechts) (Bildquellen: 3D Mapping Solutions)	48

5.1	Darstellung des OpenDRIVE-Schemas 1.4 als UML-Diagramm	53
5.2	Darstellung des UML-Diagramms des CityGML-Kernmoduls [Gröger et al., 2012, S. 51]	54
5.3	Flächenhafte Beschreibung eines Straßenstücks durch Geometrieobjekte	56
5.4	OpenDRIVE: Darstellung der Spurgeometrie durch Breitereinträge	58
5.5	Beschreibung des rechten Fahrbahnrandes eines Straßenstücks durch OpenDRIVE (links) und CityGML (rechts)	59
5.6	Beschreibung eines Straßenstücks mit Unregelmäßigkeiten am rechten Fahrbahnrand durch OpenDRIVE (links) und CityGML (rechts)	59
5.7	Beschreibung des rechten Fahrbahnrandes eines Straßenstücks mit polynomialer Begrenzung durch OpenDRIVE (links) und CityGML (rechts)	60
5.8	Beschreibung von kreisförmigen oder zylindrischen Objekten (links), Beschreibung von Parkbuchten (rechts) ([Dupuis et al., 2015], S.66)	60
5.9	Beschreibung von Polygonen im Track-Koordinatensystem (links) und im lokalen Koordinatensystem (rechts) ([Dupuis et al., 2015], S.68/69)	61
5.10	Beschreibung eines Straßenstücks durch Geometrie und Topologie	61
5.11	Beschreibung eines Straßenstücks durch Geometrie, Topologie und Semantik	62
5.12	Darstellung der Klasse Road des OpenDRIVE-Schemas 1.4 als UML-Diagramm	63
5.13	Verweis zwischen Verkehrszeichen	64
5.14	Beschreibung eines Straßenstücks durch Geometrie, Topologie, Semantik und visuelle Erscheinung	65
6.1	Visualisierung der Daten durch Umwandlung des OpenDRIVE-Formats in ein Computergrafikformat	70
6.2	Visualisierung der Daten durch Konvertierung des OpenDRIVE-Formats in das CityGML-Format	71
6.3	Visualisierung durch Konvertierung der Zwischenprodukte in das CityGML-Format	72
6.4	Visualisierung der MMS-Daten über die Konvertierung in das CityGML-Format und Darstellung durch KML im Web-Map-Client der 3DCityDB	73
7.1	Ablaufdiagramm der Datenbearbeitung mit verwendeter Software	75
7.2	Beispielhafte Ebenenstruktur in der DWG-Datei (links) und in FME (rechts)	78
7.3	Modellierung dreidimensionaler Randstein- und Gehwegflächen	80
7.4	Beschreibung möglicher Attributwerte einer Fahrspur	80
7.5	Darstellung einer Parkfläche vor der Generierung eines Zusatzattributs (links) und als zusammenhängende Fläche (rechts)	81
7.6	Erzeugung des Objektmittelpunkts aus gemessenem Bodenpunkt	83
7.7	Erzeugung von 3D-Modellen durch Verschiebung und Skalierung	84
7.8	Erzeugung von 3D-Modellen durch Verschiebung, Skalierung und Orientierung	85
7.9	Kategorisierung der Collada-Dateien	85
7.10	Berechnung der Objektorientierung durch die Mittellinie	86

8.1	Objektextraktion in der Punktwolke (Straßenmarkierungen, Straßenlaterne, Schilderpfosten, Bäume)	92
8.2	Verschneidung flächenhafter Markierungen mit der Fahrbahnoberfläche	94
8.3	Darstellung der Randsteine auf Höhe 0 (links) und auf absoluter Höhe (rechts)	94
8.4	Texturiertes Spurmodell auf Höhe 0 mit Referenzlinie (links)	94
8.5	Darstellung von Bäumen und Pfosten	95
8.6	Darstellung von Parkbänken	95
8.7	Darstellung von von flächenhaften Objekten auf der Fahrbahn	95
8.8	Darstellung von Straßenlaternen mit Ausleger (links) und ohne Ausleger (rechts)	96
8.9	Darstellung von Verteilerkästen (links) und Blumenkästen (rechts)	96
8.10	Darstellung von Verkehrsschildern (links, mitte) und Ampeln (rechts)	97
8.11	Visualisierung der <i>Transportation</i> -Objekte im Webclient	97
8.12	Visualisierung der <i>CityFurniture</i> -Objekte im Webclient	98
8.13	Visualisierung der <i>Vegetation</i> -Objekte im Webclient	98
8.14	Visualisierung des zweiten Testgebiets in FME	101
8.15	Visualisierung eines CityFurniture-Objekts im Webclient	101

Tabellenverzeichnis

2.1	Mögliche Kriterien zur Beurteilung genannter Datenquellen (Teil 1)	17
2.2	Mögliche Kriterien zur Beurteilung genannter Datenquellen (Teil 2)	18
4.1	Mögliche Spurtypen in OpenDRIVE	47
4.2	Mögliche Objekttypen in OpenDRIVE	48
5.1	Mögliche LoDs und Geometrien zur Beschreibung des Straßenraums in CityGML	57
5.2	OpenDRIVE-Geometriertypen in CityGML und ISO 19107	57
5.3	Darstellung der OpenDRIVE-Klassen mit vergleichbaren CityGML-Klassen	66
5.4	Betrachtung möglicher OpenDRIVE-Spurtypen mit passenden CityGML-Attributen	68
7.1	Generierte Attribute der Referenzlinie	79
7.2	Generierte Spurattribute	82
7.3	Modellierung der Objekte nach Art der Extraktion	82
7.4	Generierte Objektattribute	87
7.5	Attribute der Referenzlinien	87
7.6	Attribute des Spurmodells	88
7.7	Attribute der CityFurniture-Objekte	88
7.8	Attribute der Vegetation-Objekte	89
8.1	Extrahierte Spurtypen von Testgebiet 1	92
8.2	Extrahierte Objekte von Testgebiet 1	93
8.3	Extrahierte Spurtypen von Testgebiet 2	99
8.4	Extrahierte Objekte von Testgebiet 2	100
A.1	Attributtabelle: Pfosten	123
A.2	Attributtabelle: Baum	124
A.3	Attributtabelle: Parkbank	124
A.4	Attributtabelle: Straßenlaterne	125
B.1	Liste aller modellierten Ampeln und Verkehrsschilder mit <i>maincategory_subcategory</i>	127
C.1	Quellen von verwendeten 3D-Modelle und Texturen	130

Anhang

A Generierte Attribute in Tabellen

Im Folgenden werden die Attributtabelle der modellierten Straßenraumobjekte aufgelistet. Diese gelten analog für beide Testgebiete.

Tabelle A.1 – Attributtabelle: Pfosten

Attributname	Wert	Beschreibung
gml_id	id_xx	Eindeutige ID des Objekts
gml_parent_id	fme_gen_xx	Eindeutige ID der Elternklasse
gml_name	1270	Codename des Objekts
gml_description	pole	Beschreibung des Objekts
citygml_feature_role	cityObjectMember	Objektrolle
citygml_lod_name	lod2Geometry	Geometrie
citygml_level_of_detail0	2	Detaillierungsgrad
citygml_target_uri	http://www.opengis.net/citygml/cityfurniture/2.0	
citygml_class	1030	others
citygml_function	1270	pole
citygml_usage	1270	pole
xml_type	xml_surface	XML-Geometrie
opendrive_object_type	pole	OpenDRIVE Objekttyp
opendrive_object_name	pole	OpenDRIVE Objektname
object_coordinates	x,y,z [m]	Objektkoordinaten
object_height	xx [m]	Objekthöhe
object_diameter	xx [m]	Objektdurchmesser
object_radius	xx [m]	Objektradius

Tabelle A.2 – Attributtabelle: Baum

Attributname	Wert	Beschreibung
gml_id	id_xx	Eindeutige ID des Objekts
gml_parent_id	fme_gen_xx	Eindeutige ID der Elternklasse
gml_name	1070	Codename des Objekts
gml_description	unknown	Beschreibung des Objekts
citygml_feature_role	cityObjectMember	Objektrolle
citygml_lod_name	lod2Geometry	Geometrie
citygml_level_of_detail0	2	Detaillierungsgrad
citygml_target_uri	http://www.opengis.net/citygml/vegetation/2.0	
citygml_class	1070	Laubbaum
citygml_function	1070	Laubbaum
citygml_usage	1070	Laubbaum
xml_type	xml_aggregate	XML-Geometrie
opendrive_object_type	tree	OpenDRIVE Objekttyp
opendrive_object_name	tree	OpenDRIVE Objektname
object_coordinates	x,y,z [m]	Objektkoordinaten
object_height	xx [m]	Objekthöhe
object_diameter	xx [m]	Stammdurchmesser
object_radius	xx [m]	Stammradius

Tabelle A.3 – Attributtabelle: Parkbank

Attributname	Wert	Beschreibung
gml_id	id_xx	Eindeutige ID des Objekts
gml_parent_id	fme_gen_xx	Eindeutige ID der Elternklasse
gml_name	1530	Codename des Objekts
gml_description	bench	Beschreibung des Objekts
citygml_feature_role	cityObjectMember	Objektrolle
citygml_lod_name	lod2Geometry	Geometrie
citygml_level_of_detail0	2	Detaillierungsgrad
citygml_target_uri	http://www.opengis.net/citygml/cityfurniture/2.0	
citygml_class	1030	others
citygml_function	1530	bench
citygml_usage	1530	bench
xml_type	xml_aggregate	XML-Geometrie
opendrive_object_type	obstacle	OpenDRIVE Objekttyp
opendrive_object_name	bench	OpenDRIVE Objektname
object_coordinates	x,y,z [m]	Objektkoordinaten (Mittelpunkt)
object_length	xx [m]	Objektlänge

Tabelle A.4 – Attributtabelle: Straßenlaterne

Attributname	Wert	Beschreibung
<code>gml_id</code>	<code>id_xx</code>	Eindeutige ID des Objekts
<code>gml_parent_id</code>	<code>fme_gen_xx</code>	Eindeutige ID der Elternklasse
<code>gml_name</code>	1150	Codename des Objekts
<code>gml_description</code>	streetlamp, latern or candelabra	Beschreibung des Objekts
<code>citygml_feature_role</code>	cityObjectMember	Objektrolle
<code>citygml_lod_name</code>	lod2Geometry	Geometrie
<code>citygml_level_of_detail0</code>	2	Detaillierungsgrad
<code>citygml_target_uri</code>	http://www.opengis.net/citygml/cityfurniture/2.0	
<code>citygml_class</code>	1020	security
<code>citygml_function</code>	1150	streetlamp, latern or candelabra
<code>citygml_usage</code>	1150	streetlamp, latern or candelabra
<code>xml_type</code>	xml_aggregate	XML-Geometrie
<code>opendrive_object_type</code>	pole	OpenDRIVE Objekttyp
<code>opendrive_object_name</code>	streetLamp	OpenDRIVE Objektname
<code>object_coordinates</code>	x,y,z [m]	Objektkoordinaten
<code>object_height</code>	xx [m]	Objekthöhe
<code>object_diameter</code>	xx [m]	Objektdurchmesser
<code>object_radius</code>	xx [m]	Objektradius
<code>object_rotation</code>	xx [deg]	Objektorientierung

B Liste der modellierten Verkehrsschilder und Ampeln

Tabelle B.1 – Liste aller modellierten Ampeln und Verkehrsschilder mit *maincategory_subcategory*

Ampel_1.000.001	Vorschriftzeichen_205	Vorschriftzeichen_283-30
Ampel_1.000.002	Vorschriftzeichen_206	Vorschriftzeichen_286-10
Gefahrzeichen_101-11	Vorschriftzeichen_208	Vorschriftzeichen_286-20
Gefahrzeichen_131	Vorschriftzeichen_209	Vorschriftzeichen_286-21
Richtzeichen_306	Vorschriftzeichen_209-10	Vorschriftzeichen_286-31
Richtzeichen_308	Vorschriftzeichen_220-10	Zusatzzeichen_1000-10
Richtzeichen_314	Vorschriftzeichen_220-20	Zusatzzeichen_1002-10
Richtzeichen_314-10	Vorschriftzeichen_222	Zusatzzeichen_1002-12
Richtzeichen_314-20	Vorschriftzeichen_224	Zusatzzeichen_1002-21
Richtzeichen_314-30	Vorschriftzeichen_224-40	Zusatzzeichen_1002-24
Richtzeichen_314-50	Vorschriftzeichen_229-10	Zusatzzeichen_1004-30
Richtzeichen_386	Vorschriftzeichen_240	Zusatzzeichen_1007-37
Richtzeichen_418-10	Vorschriftzeichen_250	Zusatzzeichen_1007-62
Richtzeichen_418-20	Vorschriftzeichen_253	Zusatzzeichen_1010-51
Richtzeichen_419-10	Vorschriftzeichen_262	Zusatzzeichen_1026-35
Richtzeichen_419-20	Vorschriftzeichen_264	Zusatzzeichen_1040-31
Richtzeichen_432-20	Vorschriftzeichen_267	Zusatzzeichen_1042-32
Richtzeichen_434-50	Vorschriftzeichen_274-20	Zusatzzeichen_1042-33
Richtzeichen_434-51	Vorschriftzeichen_283-10	Zusatzzeichen_1044-10
Richtzeichen_437	Vorschriftzeichen_283-11	Zusatzzeichen_1053-31
Verkehrseinrichtungen_605-10	Vorschriftzeichen_283-20	Zusatzzeichen_1053-34
Verkehrseinrichtungen_605-20		

C Quellen von verwendeten 3D-Modelle und Texturen

Tabelle C.1 – Quellen von verwendeten 3D-Modelle und Texturen

Liste verwendeter SketchUp-Modelle

Objektart	Link	Zugriff
Pfosten	https://3dwarehouse.sketchup.com/model/16d9a6cc60e47009d3dd3331d9d62fd5/Pole	22.05.2018
Ampel- und Schilderpfosten	https://3dwarehouse.sketchup.com/user/0987206586494547490458031/Philipp?nav=models	22.05.2018
Straßenlaterne	https://3dwarehouse.sketchup.com/model/7260ad08488a3156f78e32cea3be35af/Street-Light	22.05.2018
Parkbank	https://3dwarehouse.sketchup.com/model/a887d194b1b35b5f1463fd04979966ca/Street-King-Luminaire	22.05.2018
Baum	https://3dwarehouse.sketchup.com/model/f411f2ed-f5c5-4d1e-b644-b602412b235/Marshalls-Escotet-Kiwi	22.05.2018
Verkehrszeichen	https://3dwarehouse.sketchup.com/user/0987206586494547490458031/Philipp?nav=models	22.05.2018
Ampeln	https://3dwarehouse.sketchup.com/model/c81147539892d853feddd24b42735e2a4/Deutsche-Vorgeschriebene-Fahrtrichtungszeichen	22.05.2018
	https://3dwarehouse.sketchup.com/model/1e16f1d12e1fafb5ea33d1e82atdde21/Niederlande-verkeerslichten	22.05.2018

Liste verwendeter Texturen

Textur	Link	Zugriff
Asphalt	https://texturelib.com/texture/?path=/Textures/road/bare%20asphalt/road_bare_asphalt.0038	22.05.2018
Pflasterung	https://texturelib.com/texture/?path=/Textures/brick/pavement/brick_pavement.0126	22.05.2018
Pflasterung	http://texturelib.com/texture/?path=/Textures/brick/pavement/brick_pavement.0131	22.05.2018
Gras	http://texturelib.com/texture/?path=/Textures/grass/grass/grass.0130	22.05.2018