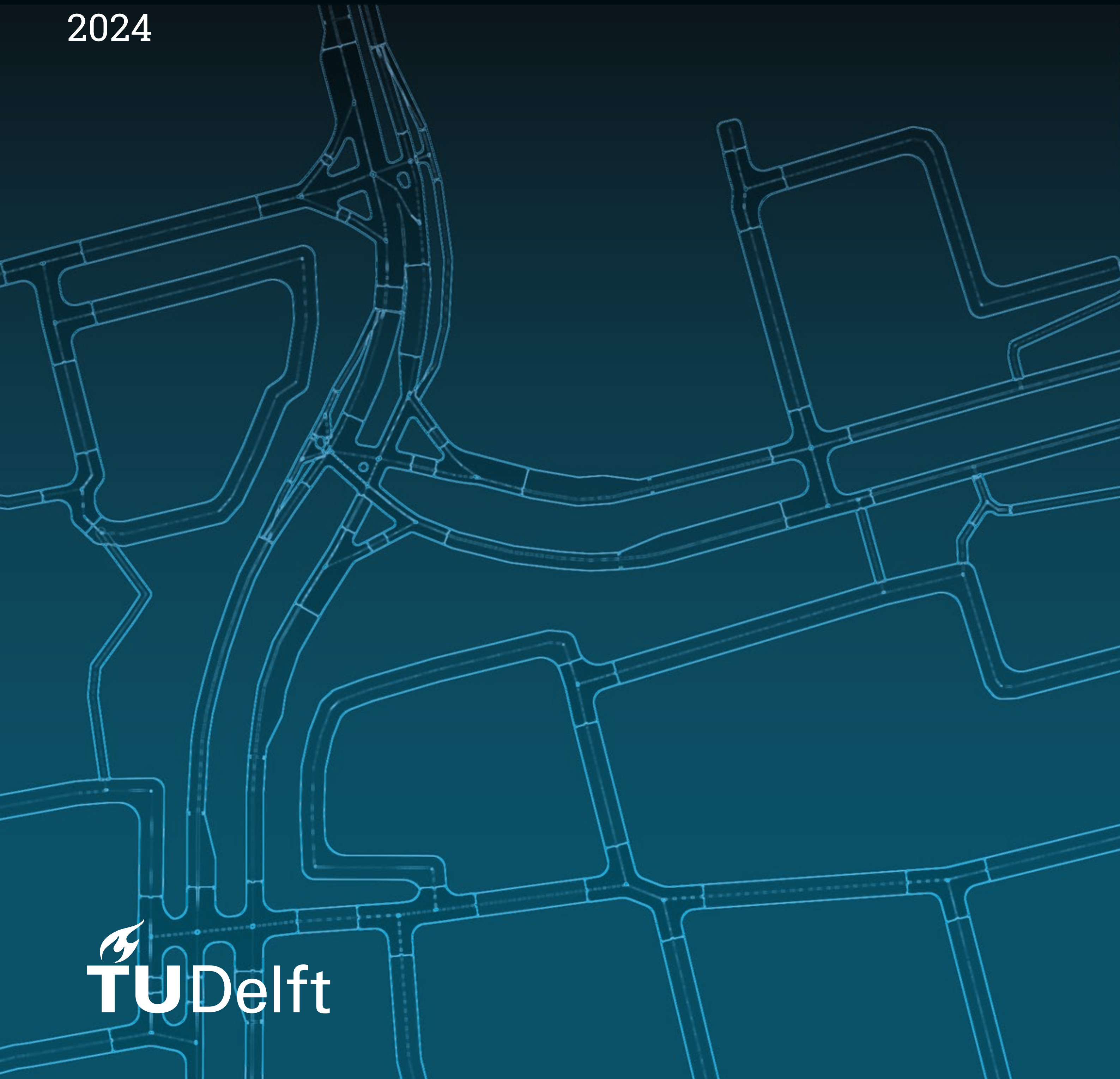
MSc thesis in Geomatics

Reconstructing a high-detailed 2D areal representation of road network based on OSM data

Chengzhi Rao

2024



TU Delft

MSc thesis in Geomatics

Reconstructing a high-detailed 2D areal representation of road network based on OSM data

Chengzhi Rao

October 2024

A thesis submitted to the Delft University of Technology in partial fulfillment of the requirements for the degree of Master of Science in Geomatics

Chengzhi Rao: *Reconstructing a high-detailed 2D areal representation of road network based on OSM data* (2024)

The work in this thesis was carried out in the:

3D geoinformation group
Delft University of Technology

Supervisors:	Dr. Ken Arroyo Ohori
	Dr. Stelios Vitalis
Co-reader:	Dr.ir. Martijn Meijers

Abstract

Road network models are essential for modern urban planning, transportation analysis, and digital map reconstruction. This thesis addresses the development of a comprehensive digital road network model that advances from linear representation to a lane-level and areal polygonal representation, based on OpenStreetMap (OSM) data. While OSM provides rich road network data, challenges remain in reconstructing high-resolution, lane-level networks that are both geometrically precise and topologically consistent.

The objective of this research is to create a refined digital road model that enhances the geometric accuracy and topological integrity of road networks by transitioning from road-level centerlines to lane-level polygons and full areal representations. This model is designed to capture essential aspects and semantic information such as traffic modes (motorized and cycling), intersection complexity, and road connectivity.

To achieve these objectives, the lane network generation methodology integrates OSM data with graph theory and civil engineering principles, focusing on lane topology and connectivity. The approach enhances linear geometry into areal geometry by combining a buffer-based method for road ribbons with a node-based method for intersection geometry. This process generates multiple intermediate results, including a road-level data model, a traffic-level data model, and grouped strokes representing global and local adjacency as graph-like models. Additionally, lane-level networks, lane geometry, and areal representations of roads and intersections are developed. Key innovations include a generalized method for lane centerline and polygon generation, ensuring road-lane correspondence and consistency between low-LoD linear models and high-LoD areal models.

The final product is a digital road network model that accommodates variations in road shapes, realistic intersection geometry, and detailed traffic lane information, enabling highly detailed urban simulations and transportation analyses. While challenges remain regarding lane alignment, complex intersections, and the lack of detailed traffic signal data, this research successfully bridges the gap between OSM's raw linear road centerline data and the lane-level areal representations.

Keywords: OpenStreetMap, Road network model, Lane-level network, Linear representation, Areal representation, Road, Intersection, Topology, Graph, Geometry, Road centerline, Semantics, Traffic modes.

Acknowledgements

First of all, I would like to express my deepest gratitude to my supervisors, Ken Arroyo Ohori and Stelios Vitalis. Their steadfast backing, insightful direction, and willingness to help have been priceless during this endeavour. They provided me with the freedom to explore my research topic, allowing me to experience the joy of discovery and the satisfaction of overcoming unexpected challenges.

I am also sincerely grateful to my co-reader, Martijn Meijers, for his insightful feedback and for helping me identify the blind spots in my thesis. His contributions were instrumental in refining my work and broadening my perspective.

As I complete my studies at TU Delft and the Geomatics program, I am filled with appreciation for the incredible journey of these past two years. Starting from scratch with no experience in data analysis or coding, I have grown immensely. Each day brought new learning experiences, broadening my horizons and pushing me beyond my comfort zone. While knowledge is limitless, the courage to embrace change and adapt to new environments is equally boundless—a quality I will carry forward in life.

A heartfelt thank you goes to my dear friends and sisters, spread across the globe—China, Sweden, the UK, Germany, Italy, and beyond. They have been my confidants, my source of solace, and have made me feel understood and valued. Their presence reminds me that I am never alone, even when navigating the uncertainties of life.

To my family, though separated by distance, I am forever grateful. They are my roots and my source of courage, always offering their support and love from afar.

Lastly, I want to thank myself. I recognize the struggles I faced, and I am proud that I chose to confront them head-on, rather than being held back. Here's to embracing the flow of life and continuing to move forward.

Proceed and navigate. Find my anchor. Go with the flow.

Contents

1. Introduction **1**
1.1. Motivation . . . 1
1.2. Research objectives . . . 5
1.3. Research questions . . . 6
1.4. Research challenges and scope . . . 7
1.4.1. Research challenges . . . 7
1.4.2. Scope of the Research: . . . 7
1.5. Thesis overview . . . 8

2. Background **9**
2.1. Important terminology in transportation domain . . . 9
2.1.1. Terminology of Road . . . 9
2.1.2. Terminology of Lane . . . 9
2.1.3. Terminology of Intersection and Overpass . . . 10
2.2. Road network . . . 11
2.2.1. Road network modelling . . . 11
2.2.2. Graph theory with road network . . . 12
2.3. Road network in OpenStreetMap . . . 14
2.3.1. OpenStreetMap (OSM) road network structure . . . 14
2.3.2. Road representation and modelling in OSM . . . 16
2.4. Important Geometric Characteristics of Road Network . . . 19
2.4.1. 'Strokes' concept in road network . . . 19
2.4.2. Geometric design of highways and streets . . . 19
2.4.3. Turning paths for roadway designs . . . 21
2.5. Summary of road network background . . . 22

3. Related work **25**
3.1. Categories of related road network researches . . . 25
3.2. Stroke-Based Studies . . . 25
3.3. OSM-Driven Studies . . . 27
3.3.1. Generate 2D road network from OSM road centrelines . . . 27
3.3.2. Enhanced 2D road network generation methods by integrating OSM data with other source datasets . . . 30
3.4. Road network 2D geometry generation . . . 31
3.4.1. Road width transition part generation . . . 31
3.4.2. Intersection polygon generation . . . 33

4. Methodology **35**
4.1. Methodological Framework . . . 35
4.1.1. Key stages of methodology . . . 35
4.1.2. Pipeline and workflow diagram . . . 37

4.2. Model Design and Method Design . . . 38
4.2.1. Model Design . . . 39
4.2.2. Method Design . . . 44
4.3. OSM Road Network Initialization and Stroke Generation . . . 45
4.3.1. Data preparation and filtering . . . 46
4.3.2. OSM data pre-processing . . . 47
4.3.3. Road adjacency identification . . . 48
4.3.4. Grouped strokes generation . . . 51
4.3.5. Update road attributes . . . 53
4.4. Lane Network Reconstruction . . . 53
4.4.1. Hierarchical Reconstruction of Lane Centerlines . . . 55
4.4.2. Lane-level Network Connectivity Modification . . . 65
4.4.3. Lane-level Network Connectivity Correction . . . 68
4.5. 2D Road Polygon Reconstruction . . . 70
4.5.1. General Method for Polygon Generation . . . 71
4.5.2. Post-processing: Geometry Errors Detection and Modification . . . 71
4.5.3. Intersection Geometry Generation . . . 75

5. Implementation **77**
5.1. Datasets, software and libraries . . . 77
5.2. OSM Road Network Initialization . . . 79
5.2.1. OSM data pre-processing . . . 79
5.3. Grouped Strokes Generation . . . 79
5.4. Hierarchical Reconstruction . . . 82
5.4.1. Specifics of first-order lane generation . . . 82
5.5. Lane-level Connectivity Modification and Correction . . . 82
5.6. 2D Road Polygon Generation . . . 84

6. Results and analysis **87**
6.1. Results of 2D linear representation . . . 87
6.1.1. Results of pre-processing and grouped stroke generation . . . 87
6.1.2. Results of hierarchical reconstruction . . . 90
6.1.3. Results of lane network modification and correction . . . 97
6.1.4. Results of final lane-level network . . . 102
6.2. Results of 2D areal representation . . . 103
6.2.1. Road polygons of general method . . . 103
6.2.2. Results of post-processing . . . 105

7. Conclusion and Future Work **111**
7.1. Research Overview . . . 111
7.2. Impacts and Contributions . . . 115
7.3. Limitations . . . 117
7.3.1. Dependency on OSM Data Quality . . . 117
7.3.2. No optimization for the raw road centerlines . . . 117
7.3.3. Efficiency Constraints . . . 118
7.4. Future work . . . 118
7.4.1. Test more datasets for more regions . . . 118
7.4.2. C1 continuous curve optimization . . . 118
7.4.3. Automatic reconstruction . . . 119

A. Reproducibility self-assessment **121**
A.1. Marks, and Self-reflection 121

List of Figures

1.1. Levels of Detail (LoDs) of road representation, figure from Labetski, A. and van Gerwen, S. and Tamminga, G. and Ledoux, H. and Stoter, J. [2018]. 2
1.2. Main types of digital road model representation.[1] 3
1.3. Changes of road width, figure from Chatzidiakos [2021]. 4
1.4. Visualization of 3D road centerline of Bern, data from Swisstopo, SwissTLM3D - Swiss Topography Landscape Model. 5
1.5. Model details of 3D City model of Zurich, the blurred road boundary indicates the poor quality of the road model. 5

2.1. Through lane, turning lanes and the lane allows both of through and turning maneuvers. 10
2.2. LoDs of road linear representation, from LoD0.1, the representation can describe road network as road segments (edges) and intersections (nodes), the lines at LoD0.2 represent carriageway centerlines, and at LoD0.3 they are lane centerlines. Figure from Labetski, A. and van Gerwen, S. and Tamminga, G. and Ledoux, H. and Stoter, J. [2018]. 12
2.3. LoDs of road areal representation, LoD1 describes road level, LoD2 distinguishes carriageways, and LoD3 models every lanes. 12
2.4. Example of multi-LoDs road representation, areal representation at LoD2, and linear representation LoD0.3, figure from Boersma [2019] 13
2.5. Linear semantic enrichment (left) and areal semantic enrichment (right) of *TrafficSpace* coloured by function attributes in CityGML. Figures from Beil et al. [2020]. 13
2.6. Linear representations of predecessor/successor relations in proposed CityGML 3.0. Figure from Beil et al. [2020]. 14
2.7. Unalignment of lanes caused by the uncertainty of vector direction. 17
2.8. Uncertainty of the intersection labelling. 18
2.9. Illustration for the difference outcomes by using road centerline to generate road polygons 18
2.10. Road segments and strokes structure 19
2.11. Typical examples of lane balance. Figure from AASHTO [2011] 20
2.12. Coordination of lane balance and basic number of lanes through application of auxiliary lanes. Figure from AASHTO [2011]. 21
2.13. Left: minimum centerline turning radius (CTR), fgure from Taylor. Right: OSM simplification of road curvature. 22

3.1. Angle calculation of ***COINS***. 26
3.2. Stroke generation results. 27
3.3. Screenshot of "osm2street", output the road polygons for a chosen region. . . 28
3.4. **A/Bstreet** road polygon generation method 28
3.5. Screenshot of "Detailed rendering of bicycle lanes and junctions as part of the OSM 'Straßenraumkarte'". 29

3.6. Using the 'area:highway' tag to render detailed intersection. [OpenStreetMap Berlin] . . . 29
3.7. OSM-based researches . . . 30
3.8. Variable buffer for robust roadway width transition . . . 32
3.9. Road connector . . . 32
3.10. From lane connectivity to road polygons (HERE Technologies -Lane Model) . 32
3.11. **A/Bstreet** intersection generation method . . . 33
3.12. StreetGen intersection generation schema . . . 33
3.13. StreetGEN intersection generation details . . . 34
3.14. Before and after performing clip operation to polygons near intersection . . . 34

4.1. Methodology overview . . . 37
4.2. Detailed workflow diagram . . . 38
4.3. Structure and transformation of three levels of network, areal representation, from the original OSM data to outcome as polygons. . . . 40
4.4. Schema of extracting lane information from OSM raw data, then applying them for lane generation. . . . 47
4.5. Interior angle for adjacent conditions detections . . . 49
4.6. Fork labelling based on the interior angles. . . . 52
4.7. Comparison of using the different assumptions for lane network and road polygon generation. . . . 55
4.8. Main factors for the hierarchical generation approach to achieve the 'good lane continuity'. . . . 55
4.9. Road-level and lane-level "main route" and "branches", expanding meanings from "stroke" concept. . . . 56
4.10. Schema of hierarchical roads selection process and decision. . . . 56
4.11. Hierarchical generation result overview. . . . 57
4.12. 2nd-order offset rules: 'through' and bike lane matching. . . . 60
4.13. In 3rd order offsetting process, the matching situation could be ideal or not ideal, i.e. adjacent lanes match partly. . . . 62
4.14. Generation method for 4th-order road, the bi-directional roads. . . . 63
4.15. Special instruction schema to resolve the issues caused by uneven lane distribution in some bi-directional roads. . . . 64
4.16. Modify the lanes of fork-labelled roads to ensure the correct topology and geometry relations with their adjacent lanes in bi-directional roads. . . . 66
4.17. Left: modify slip road lane centerline geometry. Right: modify auxiliary lane centerline geometry. . . . 67
4.18. Modify "cycleway-bike lane" centerline geometry . . . 67
4.19. Lane topology correction . . . 70
4.20. Lane geometry correction . . . 70
4.21. Different Node degree and the corresponding polygons. . . . 73
4.22. Post-processing methods driven by the corrected true node degree. . . . 74
4.23. Schema of dividing intersection geometry from road polygons . . . 76

5.1. Dataset and region. . . . 77
5.2. 1st grouping result . . . 81
5.3. 2nd grouping result . . . 82
5.4. Fork results . . . 83
5.5. Intersection buffer method for geometry results. . . . 84

6.1. OSM road data pre-processing results. 87
6.2. First result of grouped stroke for all roads after pre-processing. 88
6.3. Comparison of the adjustment results of wrongly labelled slip roads. 88
6.4. Results of all the labelled slip roads. 89
6.5. Representative slip roads labels. 90
6.6. Number of lanes in each road after data pre-processing. 91
6.7. First-order and Second-order roads and generated lanes with accurate location. 92
6.8. Third-order roads and generated lanes with accurate location. 94
6.9. Representative results of 3rd-order lanes generation, connecting with other lanes and converging to intersetions. 95
6.10. Fourth-order and other order roads and generated lanes with accurate location. 95
6.11. Representative results of 4th-order lanes generation, connecting with other lanes and converging to intersetions. 96
6.12. Representative results of special cases regeneration, comparison of first generation result and the refined one. 96
6.13. Results of lane modification for the 'one-way' slip roads. 97
6.14. Results of lane modification for the cycleways to align with bike lanes. 98
6.15. Results of lane modification for the slip roads connect to 'two-way' roads. . . . 98
6.16. Results of lane modification for the auxiliary lanes to align with 'through' lanes. 99
6.17. Results of lane correction: Topology aspect. 100
6.18. Representative results of lane correction: Geometry aspect, join gaps and remove tails. 100
6.19. Final result of lane generation. 101
6.20. Multiplex scenario 1 of final result of lane generation. 102
6.21. Multiplex scenario 2 of final result of lane generation. 102
6.22. Two dissolved results: left is merged by road indices, right is merged by the lane types. 103
6.23. Road polygon generation for special case: dual-carriageways with slip roads . 104
6.24. The corrected nodes and their true node degree as N_k for each intersection. . . 105
6.25. Results of all trimmed polygons to remove the overlapping parts as the adjacent pairs. 106
6.26. Results of all trimmed polygons to avoid the errors at endpoints. 107
6.27. Examples of how to trim endpoint of the polygon, the dark green polygons are the correct results. 107
6.28. Results of typical intersection geometry generation. 108
6.29. Results of roundabout geometry generation. 108
6.30. Results of all intersection polygons without overlapping. 109

7.1. From OSM road centerlines to semantically enriched road polygons as areal representation. 114
7.2. Intersection generation method helps to fix the errors of lane geometry. 115
7.3. Smooth the C0 road polyline to C1 continuous curve. Wilkie et al. [2012] . . . 119

List of Tables

4.1. Example of best-matching-slice and generated L_f for third-order roads. 62
6.1. Results of all the order roads and generated lanes. 91
7.1. Uasges of the tags and corresponding values. 112

List of Algorithms

4.1. Road Adjacency ($\boldsymbol{R}_w$) 50

5.1. First-order lane centerlines generation 86

Acronyms

GIS Geographical Information System 1
LoDs Levels of Detail xiii
OSM OpenStreetMap ix
LiDAR Light Detection and Ranging 1

1. Introduction

1.1. Motivation

Roads are a fundamental element of human civilization, forming road networks that are the backbone of modern society (AASHTO [2011]). These physical road networks are complex structured system composed of a large number of road segments and intersections, shaping the layout of cities, linking communities and enabling the movement of people and goods that accommodate diverse forms of transportation (Gu et al. [2024]). A digital road network is a virtual representation of real-world road infrastructure. It encompasses the geometry, topology, and attributes of roads, intersections, and related elements, stored in a digital format. Digital road networks are typically derived from spatial data, such as Geographical Information System (GIS) databases, sensors, and other digital sources to reflect the real-world road infrastructure accurately (Schilling and Kutzner [2020], Geiger et al. [2012]).

Digital models of road networks are detailed representations of these digital road networks, that aim to capture real-world roads in a virtual form that can be analyzed and manipulated computationally. These models are crucial for a wide range of applications, such as navigation systems, urban planning, traffic management, and digital twin development, where a precise understanding of connectivity and spatial relationships is required. The digital representation of road networks often takes a graph-like form, derived from various datasets such as 2D geospatial data, satellite imagery, and 3D elevation data (Dey et al. [2019]; Gao et al. [2021]; Li et al. [2023]). However, the diversity in data sources and modelling techniques leads to inconsistencies in representation, making it difficult to standardize models for diverse applications (Labetski [2017]).

Digital road network models are typically modelled at different (LoDs), including centrelines, carriageways, and lanes (see Figure. 1.1). The choice of representation method and LoDs depends on the specific application's needs (Biljecki et al. [2015]). While linear models represent roads as centerlines, capturing their connectivity and basic structure, they often fail to reflect the full extent of road geometry, such as the width variations or detailed surface features necessary for more advanced analysis. Improving the accuracy and detail of these models through technologies like Light Detection and Ranging (LiDAR) and satellite imagery is a focus of both academic and commercial efforts. However, acquiring detailed, high-resolution data remains challenging, often restricted to certain regions or proprietary sources, despite progress in manual road modelling, it remains labour-intensive and time-consuming.

Despite extensive research in road network modelling, studies and applications can generally be categorized into three main types (illustrations shown as Figure 1.2) each with a distinct focus.

The first category models road networks as 2D linear geometries, where roads and lanes are represented as single centerlines. As the most common type of road network model, it emphasizes accurate topological relationships and is used in digital maps for the purposes of

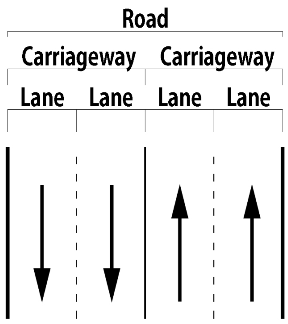


Figure 1.1.: LoDs of road representation, figure from Labetski, A. and van Gerwen, S. and Tamminga, G. and Ledoux, H. and Stoter, J. [2018].

navigation, route planning, traffic flow analysis, etc. (Jiang et al. [2019]; Homayounfar et al. [2019]). High-resolution, graph-like representations are crucial to most navigation services, and with the rise of autonomous vehicles, lane-level navigation systems and high-definition maps have become central to transportation and geospatial research (Xu et al. [2023b]; Yue et al. [2007]).

The second category focuses on areal representations of roads and street spaces. In urban environments, road networks function as vital connectors between other key components, such as buildings, water bodies, and terrain. Areal road models can be generated from linear geometries to approximate road surfaces, or through satellite imagery and data fusion techniques to extract road boundaries and road surface features (Xu et al. [2021]; Zhang et al. [2018]).

The third category involves 3D road network models, a growing trend in geospatial technology. Unlike 2D models, 3D models integrate elevation values to represent roads as three-dimensional entities, whether linear, areal, or volumetric, particularly useful for interchanges, overpasses, and terrain variations. These models enable a more realistic depiction of street spaces and can be integrated with other built environment assets like buildings, helping address real-world challenges in urban management (Beil et al. [2020]). Yet, the transition from 2D to 3D often requires detailed input data, especially for achieving a detailed areal or volumetric 3D road model, which is not always accessible.

2D Areal Representation and Its Importance

The shift from linear to 2D areal representations addresses several limitations inherent in centerline-based models. 2D areal models depict roads as polygons, providing a more accurate description of road widths, boundaries, and surface characteristics by translating linear geometries into polygons, enabling better spatial accuracy. These added details are crucial for applications like navigation, transportation modelling, road safety management, pedestrian dynamics, parking space calculations, urban heat island effects, etc. (Keler et al. [2023], Kenesei [2021]). The precise interaction between roads, intersections, and adjacent urban elements must be considered. For instance, in autonomous vehicle simulations and high-definition mapping, knowing the exact boundary and surface area of roads improves navigation accuracy and safety by offering richer spatial context.

Additionally, 2D areal models play a critical role in applications that require integration with 3D data, such as digital twins and 3D city models. The areal representation ensures that digital roads align seamlessly with other urban elements like buildings, water bodies, and terrain, enabling more realistic simulations and analyses. For example, while linear models might capture elevation along a centerline, they lack the spatial extent to connect accurately with building footprints or other geographic features. Areal models fill this gap, providing the foundational layer needed for creating detailed 3D representations of urban environments.

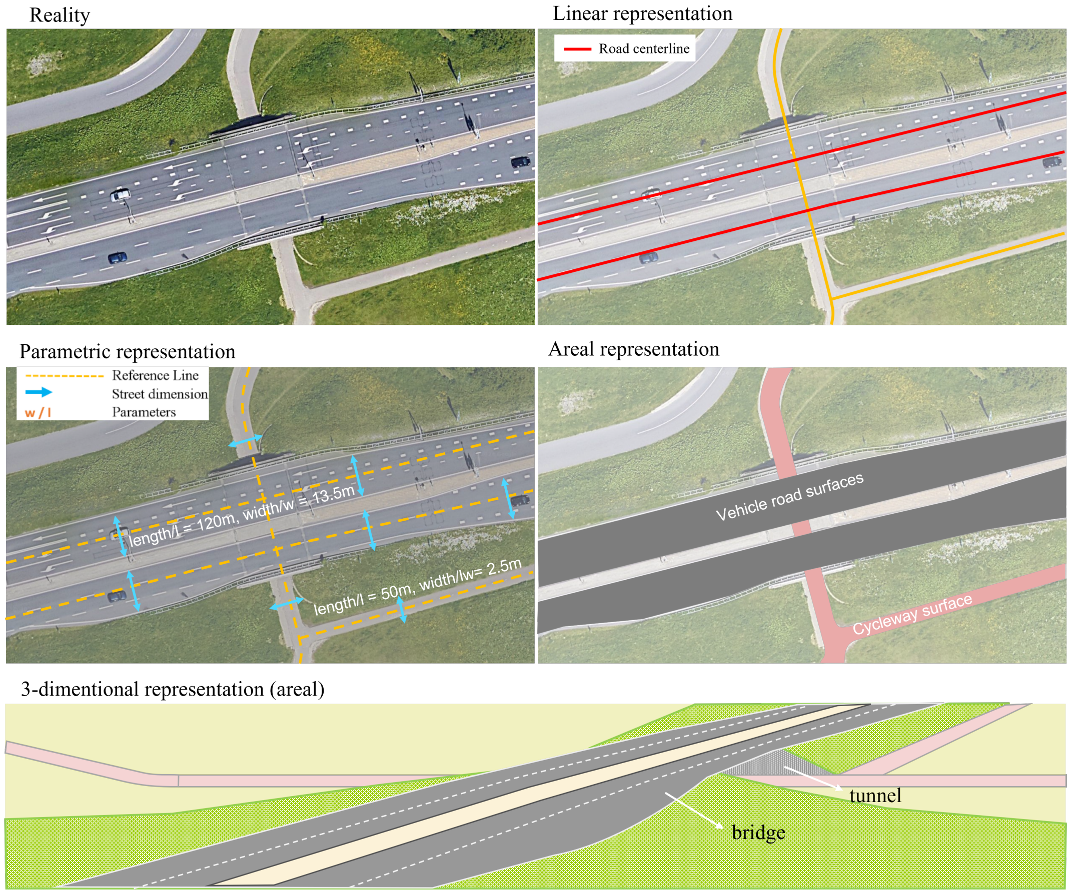


Figure 1.2.: Main types of digital road model representation.[1]

Despite numerous researches, gaps remain between different categories of road network modelling, hindering the continuity and transferability between various representations. Several areas still require improvement:

- **Limited Focus on Geographical Accuracy in Traffic Simulations:** First, traditional traffic flow simulations mainly focus on traffic loads, road properties, and intersection divisions, paying little attention to the precise geographical locations of roads. As long as the topology is accurate, geographical variations in the network produce largely consistent results (Thomson, R. and Richardson, D. [1995]).
- **Lack of Physical Detail in Lane-Level Maps:** Second, lane-level digital maps, while accurate, often fail to capture essential physical characteristics like road width, which

[1] Parametric representation uses parameters rather than geometry objects like other types, thus in this research we only include linear, areal, and 3D representation.

affects numerous modern applications, such as urban management, road maintenance, and realistic simulation environments. Variations in road width and shape, particularly at intersections and lane changes, are critical in representing real-world transportation scenarios (Figure 1.3) (Chatzidiakos [2021]).

- **Inadequate Boundary Detection in Areal Models:** Third, methods for detecting road boundaries using geospatial data frequently lack crucial details like the number of lanes and semantic attributes. These approaches are not based on civil engineering principles (Wang et al. [2014]), limiting their use in navigation systems. Due to data model constraints and format limitations, areal road representations are difficult to integrate with other digital models, such as terrain and buildings, to form a complete digital city model.

- **Reliance on Proprietary High-Resolution Data:** Additionally, high-detail models often rely on data fusion from sources like laser scanning and GPS trajectory data, which are typically sourced from private geospatial providers. This makes such methods impractical for broader applications due to the limited accessibility of high-resolution data.

- **Lack of 2D Areal Representation in 3D Model Integration:** A key challenge in achieving high-detailed 3D road models is the limited availability of accurate 2D areal representations, especially when using open-access data sources. Current 3D road models often rely on linear representations, such as those used in Swisstopo (see Figure 1.4), which include elevation but lack detailed road boundaries. In the start-of-the-art 3D city model, such as the 3D City model of Zurich, the terrain, building and vegetation models are comprehensive, but road representation is still poor, the road model is not the real 3D object with explicit surface and boundary (Figure 1.5).The transition from linear to areal representation is necessary to overcome these integration challenges and to support more precise 3D modelling efforts.

Figure 1.3.: Changes of road width, figure from Chatzidiakos [2021].

In summary, inconsistencies in road network modelling at different levels of detail hinder the development of a comprehensive road network model. The motivations of this study are inspired by the following aspects:

1. Linear representation of digital road network models is commonly used and mostly available as open data compared to other representation types;

2. Areal representation concludes the changes of road surface variation into road model, which is the key stage for comprehensive and detailed digital road modelling. However, the state-of-the-art areal representation as open data is rare.

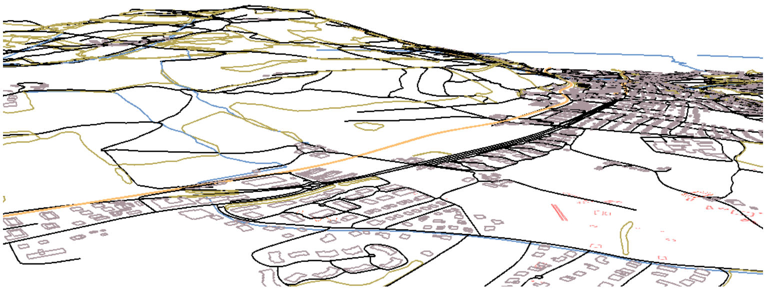


Figure 1.4.: Visualization of 3D road centerline of Bern, data from Swisstopo, SwissTLM3D - Swiss Topography Landscape Model.

Figure 1.5.: Model details of 3D City model of Zurich, the blurred road boundary indicates the poor quality of the road model.

3. The areal transformation highlights the process of polygonal modelling of road surfaces from the linear reference road centerlines, the results of areal representation can include the topological data structure as the main characteristic in linear representation to describe the adjacency and incidence relationships between geometric objects (Vitalis et al. [2019]), and also capture the details of road geometric shapes by their boundaries as a robust representation for road networks.

To address these challenges, to reconstruct the rare areal representation from the open-accessed linear representation, a more cohesive approach is needed to bridge the gaps. Our research aims to develop a procedural approach for reconstructing an accurate 2D linear and areal representation of road networks by only using OSM data. OSM, widely used for its extensive global coverage and crowdsourced map data, offers a practical solution for extracting initial road information. Previous studies have demonstrated successful road extraction from OSM data (Dai et al. [2020]). Building on this work, we aim to generate lane-level road networks from OSM centerlines and further reconstruct a high-detailed 2D areal representation that can be integrated with elevation data to achieve a 3D road network.

1.2. Research objectives

The goal of this research is to:

Reconstruct a high-detailed 2D areal representation of road network from OSM data.

The primary objective of this research is to reconstruct a 2D areal representation of a road network, leveraging OSM road data. The research aims to achieve this overarching goal through a series of specific objectives:

First, the research utilizes OSM road centerlines, one of the most common open-access data sources, to generate high-resolution 2D linear geometry, refining road-level networks to the lane-level, the lane topology provides a realistic description of vehicle movements.

Second, the study aims to enhance the LoDs and enrich the semantic information embedded in the model by extracting relevant data from OSM. This ensures that both the linear and areal representations of the road network not only maintain geometric accuracy but also incorporate detailed semantic attributes.

Third, by considering topology, geometry, and semantics, the research reconstructs a lane-level linear and areal representation of the road network based on the original graph structure, transportation design principles, and semantic enrichment for real-world simulation.

These objectives collectively support the research goal of constructing an intricate 2D road network model that preserves topological relationships and geometric accuracy, leveraging OSM data and real-world scenario principles.

1.3. Research questions

Based on the identified problems and guided by the preliminary research, the main research question is:

"How can we achieve a high-detailed areal representation of road network model with topological and geometric correctness, and enrich its semantic information by only using the OSM datasets?"

Subquestions:

1. *How can the information and attributes contained in OSM road data be utilized to generate and optimize 2D linear and areal representations of road network models?*

2. *Is it possible to enhance the resolution and LoDs of a road network model from the original road-level to a high-detailed lane-level network using only OSM data?*

3. *Can the topology relationships, lane-to-lane adjacency for representing traffic flow movements, be preserved through OSM data processing and detailed generation methods?*

4. *How can we generate 2D road polygons as errorless geometries to accurately capture road shape details, including changes in road width and intersection variations?*

1.4. Research challenges and scope

1.4.1. Research challenges

On the surface it would seem that constructing a high-LoDs road network model from open data would be preferred for facilitating various applications due to a closer resemblance to reality/"the real world" (Labetski [2017]), and that enables users and developers to reuse existing wide-spread datasets and/or extend it with new and innovative functionality (Tamminga and Hoogendoorn [2019]). At the same time there are two major problems that contradict this way of thinking:

1. **No universal method and undefined standards:** Based on the analysis of various road applications and their representation level requirements (Boersma [2019]), the results of the analysis indicated that there is no "one-size-fits-all" solution to road modelling (Vitalis et al. [2022]). Therefore, an important aspect of designing a solution for a road model consists of selecting or preparing data at the appropriate representation level. The final outcome will highly rely on the accuracy, levels of detail, and completeness of the input datasets.

2. **The robustness and resemblance of reconstruction results:** The 2D road network represents real-world entities built according to engineering criteria. Available datasets are simplified digital versions of the real world and construction approaches must address errors in the raw datasets while maintaining model accuracy. It is important to ensure the robustness of road models and minimize errors, such as topological errors and discrepancies with common sense, regardless of the input data quality. Then the levels of detail can be improved during the generation process.

1.4.2. Scope of the Research:

The scope of this research is defined by several key limitations and focus areas. First, road network reconstruction using only open-access data, OSM, presents a significant challenge due to the variability, potential errors and incompleteness of such datasets. To reduce the complexity introduced by multimodal transportation systems, this study focuses exclusively on reconstructing motorized roads and cycleways, excluding pedestrian paths, railways, and other non-motorized networks.

Since the research relies solely on OSM road centerlines, any errors inherent in the raw data will not be addressed. The methods developed will treat OSM data as a reliable and up-to-date representation of real-world conditions.

Furthermore, the research does not attempt to fully reconstruct the complete topology of intricate intersections, particularly in terms of adjacent connections between all turning lanes, these complex variations are difficult to recover using only OSM centerlines. Solving topology errors caused by the OSM data missing or data mapping issues is also out of the scope. However, errors identified in the linear representation can be partially corrected during the areal representation phase.

1.5. Thesis overview

The thesis paper is organized as follows:

- Chapter 2 introduces the key background information, including essential terminology and concepts used in this study. It also discusses the general OSM attributes targeted for exploration and processing, along with fundamental road network analysis concepts and transportation engineering principles. These elements establish the foundational understanding needed for the methods described in later chapters.
- Chapter 3 reviews the relevant literature, beginning with research on road topology analysis and generation. It then explores studies on 2D road geometry generation, geometrical-driven methods, and techniques that emphasize the integration of semantics. The focus here is on studies that have made significant contributions to understanding the characteristics of road networks, with an emphasis on detail improvement, which has been especially influential for this research.
- Chapter 4 outlines the methodology used for road network reconstruction. This chapter presents the full working pipeline, covering model design, data pre-processing, lane-level network generation and correction, as well as road polygon generation for areal representation. Detailed explanations of algorithms, diagrams, and formulas are provided to clarify the approach.
- Chapter 5 explains the implementation of the methods introduced in the previous chapter. This section offers additional insights into the decision-making process and specific steps taken during data processing and model generation.
- Chapter 6 presents the results of the reconstruction and modelling process, including both the linear and areal representations of the road network. Key intermediate products are discussed, and a thorough analysis of the results is provided.
- Chapter 7 gives a brief conclusion together with future work.

2. Background

2.1. Important terminology in transportation domain

2.1.1. Terminology of Road

- **Road**: A transportation infrastructure designed for the movement of motorized or non-motorized traffic, extending from one intersection to another.

- **Cycleway**: A thoroughfare for bicycles, describing infrastructure designed mainly for cyclists.

- **Carriageway**: A carriageway consists of a width of road on which a vehicle is not restricted by any physical barriers or separation to move laterally. Dual carriageways allow for bi-directional traffic in two different flows, while single carriageways either allow traffic flow in one direction, or they allow bi-directional traffic through the same stream. A carriageway generally consists of a number of traffic lanes together with any associated shoulder, but may be a sole lane in width. (Vitalis et al. [2022], Dual-carriageways)

- **One-way street**: A road in which traffic is only allowed to proceed in one direction.

2.1.2. Terminology of Lane

- **Lane:** a division of a road marked off with painted lines and intended to separate single lines of traffic according to speed, direction and purpose, ensuring that vehicles maintain safe distances from each other.

- **Vehicle lane:** A vehicle lane is a designated part of a roadway, typically marked by painted lines, that is meant for a single line of vehicles to travel in a specific direction.

- **Bike lane**: A designated bicycle lane is a portion of the roadway or shoulder designated for the exclusive or preferential use of cyclists.

- **Turning lane**: Turn lanes at intersections are used primarily to separate turning traffic from 'through' traffic, each lane designated for a specific, or set of movements. With turn lanes, vehicles waiting to turn are removed from the 'through' lanes. Turning lane also means the "lane is diverging and running almost parallel to the main route" (Figure 2.1), not a separate road that is disconnected from the main route area of traffic (Harlingen Car Crash Attorney).

- **Through lane**: A through lane is a traffic lane designated for vehicles that proceed straight through an intersection or continue along the main route without turning (Figure 2.1). It is intended for uninterrupted movement and is separate from turn lanes, which accommodate turning vehicles. Through lanes maintain the continuity of traffic flow along the primary direction of travel, minimizing delays caused by vehicles making turning maneuvers.

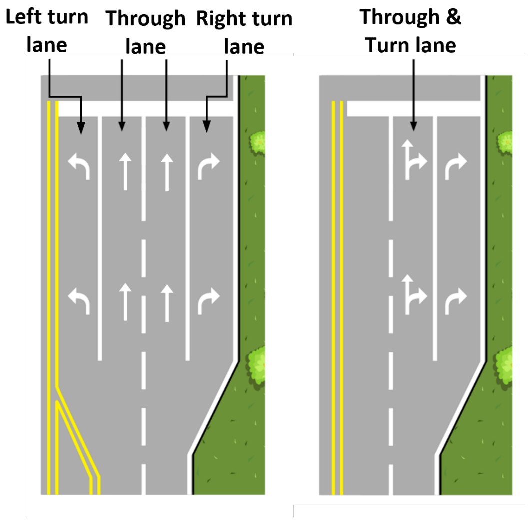


Figure 2.1.: Through lane, turning lanes and the lane allows both of through and turning maneuvers.

2.1.3. Terminology of Intersection and Overpass

- **Intersection**: An intersection or an at-grade junction is a junction where two or more roads converge, diverge, meet or cross at the same height.
- **Entering lanes**: Lanes that allow vehicles to approach and enter an intersection.
- **Exiting lanes**: Lanes that vehicles enter after crossing an intersection, allowing them to continue onto the new roadway. These lanes receive the traffic flowing out of the intersection and guide it away in the appropriate directions.
- **Roundabout**: A rotary and a traffic circle are types of circular intersection or junction in which road traffic is permitted to flow in one direction around a central island, and priority is typically given to traffic already in the junction (Press [1993]).
- **Ramp**: A short road on which vehicles join or leave a main road. Also called as **Slip road**.
- **Off-ramp**: A short road on which vehicles leave a highway or other main road as an exit road diverges from the traffic.
- **On-ramp**: A short road on which vehicles drive on to a highway or other main road as an entering road converges to the traffic.

- **Bridge**: A bridge is a structure built to span a physical obstacle (such as a body of water, road, or railway) without blocking the path underneath.
- **Underpass**: A tunnel is an underground passageway. It is dug through surrounding soil, earth or rock.
- **Overpass**: An overpass is a bridge, road or similar structure that is over another road or railway.
- **Interchange (At-grade)**: An interchange is a road junction that uses grade separations to allow for the movement of traffic between two or more roadways or highways, using a system of interconnecting roadways to permit traffic on at least one of the routes to pass through the junction without interruption from crossing traffic streams.

2.2. Road network

2.2.1. Road network modelling

In the context of road network modelling, roads can be represented both linearly and areally, reflecting different aspects of road geometry and structure. A linear representation models roads as one-dimensional entities, typically as polylines that capture the centerline of the road. This approach is useful for representing the basic path and connectivity of the road network. Conversely, an areal representation captures the width and spatial extent of roads, enabling a more detailed analysis of the road surface.

In the domain of 3D city modelling, the most relevant thematic model in the context of representing transportation space is the CityGML ”Transportation Model” (Gröger et al. [2012]), which is an open data model and XML-based format to represent, store, and exchange semantic 3D city models. Similar to the other city objects, transportation features can be represented in several consecutive LoDs (Gröger et al. [2012]). The linear representation of the road network is LoD0, starting from LoD1 road objects are spatially represented by (multi)surfaces (i.e. areal representation). With the enhancement of LoDs, the model allows more detailed semantic information and segmentation.

The refinement of road LoDs as proposed by Beil and Kolbe [2017] indicates that in LoD0 and LoD1, one road is represented as a single centerline and a polygon shows the entire width of the transportation space. Starting from LoD1, a road can be modelled by multiple areal objects respectively to represent the different types of transportation. The complete road surface is the combination of vehicle carriageways, cycleways and footpaths that become possible in LoD2, each type is an individual *TrafficSpace* object; in LoD3, with the further decomposition of one *TrafficSpace* object, the model allows the representation of each driving lane.

Labetski, A. and van Gerwen, S. and Tamminga, G. and Ledoux, H. and Stoter, J. [2018] propose an improved transportation model in CityGML subsequently, the changes facilitate the ability to describe roads as a multi-LoDs model by combining the linear representation and areal representation. For linear representation, the road network can be represented as LoD0.0 to LoD0.4 (Figure 2.2, and the granularity increases from the single road centerline to the carriageway centerlines, to the individual lane centerlines. Correspondingly, the areal representation can be represented as LoD1-road, LoD2-carriageway, and LoD3-lane representation (see Figure 2.3). The multi-LoDs model, for example, LoD2.3 (Figure 2.4) indicates the areal representation is in LoD2 (carriageway representation), and the linear representation for the road network is in LoD0.3. Combining the semantic information of each *TrafficSpace* object, the enrichment representation of road network and space can be visualized as Figure 2.5.

Apart from the traffic space, other transportation objects such as the area in the physical world that is not directly used for vehicle, bike or pedestrian movement also standardized in the "Transportation Model", for instance, middle lanes, kerbstones, barriers or green space, which called *AuxiliaryTrafficAreas* (Gröger et al. [2012]). This study focuses solely on the specific road space utilized for driving, and cycling, *AuxiliaryTrafficAreas* is out of scope.

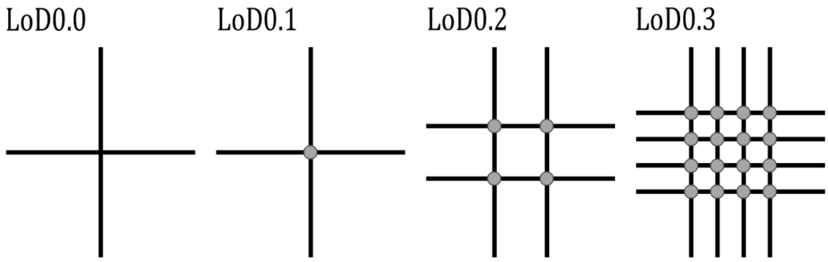


Figure 2.2.: LoDs of road linear representation, from LoD0.1, the representation can describe road network as road segments (edges) and intersections (nodes), the lines at LoD0.2 represent carriageway centerlines, and at LoD0.3 they are lane centerlines. Figure from Labetski, A. and van Gerwen, S. and Tamminga, G. and Ledoux, H. and Stoter, J. [2018].

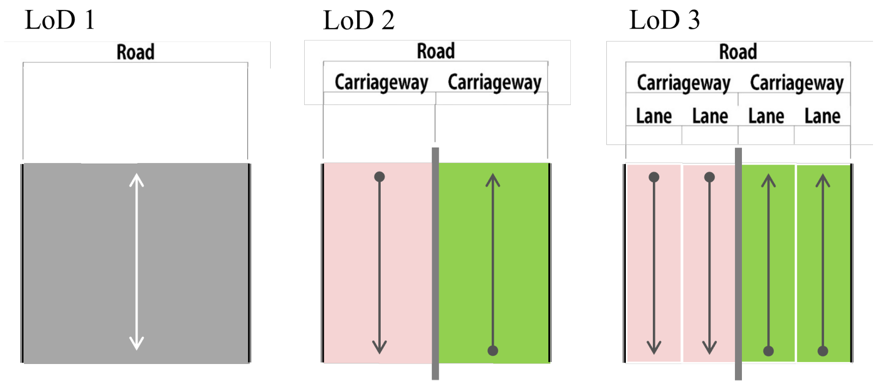


Figure 2.3.: LoDs of road areal representation, LoD1 describes road level, LoD2 distinguishes carriageways, and LoD3 models every lanes.

2.2.2. Graph theory with road network

To enhance the understanding of road connectivity and directionality, graph representations offer a convenient means of handling the topological structure and associated information describing a road network. A road network can be abstracted as a graph-like model, which is an object consisting of two sets called its vertex set and its edge set (Thomson, R. and

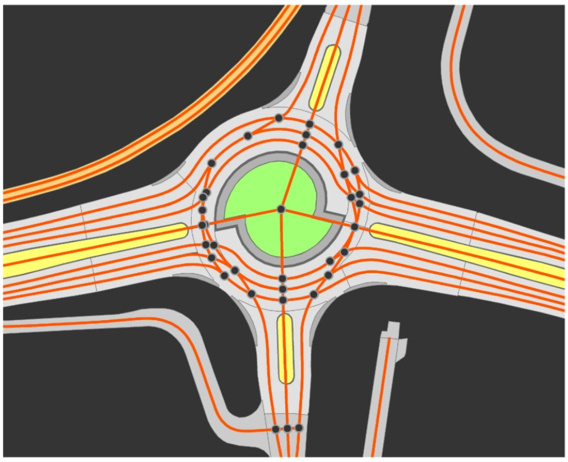


Figure 2.4.: Example of multi-LoDs road representation, areal representation at LoD2, and linear representation LoD0.3, figure from Boersma [2019]

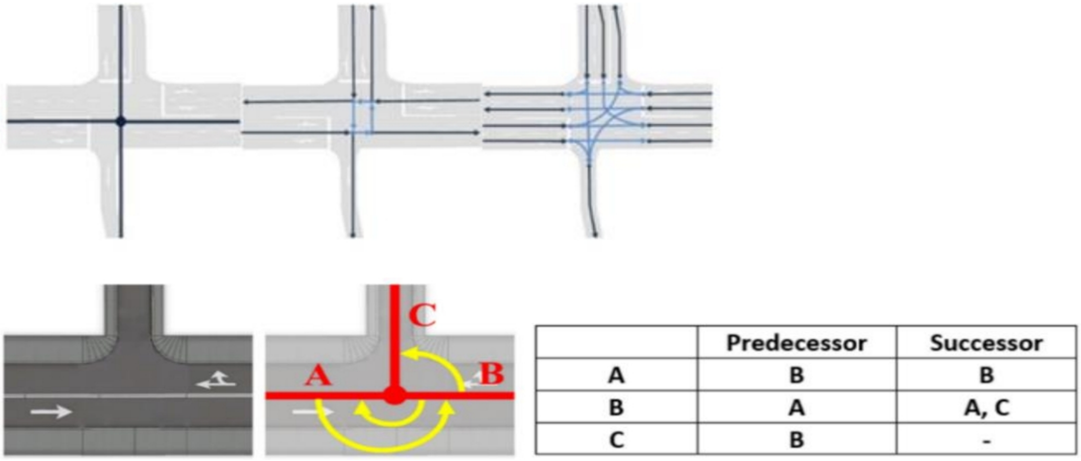
Figure 2.5.: Linear semantic enrichment (left) and areal semantic enrichment (right) of *TrafficSpace* coloured by function attributes in CityGML. Figures from Beil et al. [2020].

Richardson, D. [1995]). Road lines are represented as edges; the start and end vertex of each road line are considered as graph vertices, also known as *node* to distinguish them from the geometry vertex. Each node signifies a junction or terminal point, while edges represent the road segments connecting these nodes.

The node degree, **degree** (or valency) in a road network graph, denoted as k, corresponds to the number of road segments (edges) connected to a node (Trudeau [1993]). For instance, When the degree k of the node is 2, the node is the shared terminal point of two adjacent roads, and high-degree nodes indicate major intersections with multiple connecting roads, which are critical points for traffic flow analysis and intersection reconstruction.

One of the most essential characteristics of a road line is the directionality (i.e. road line is a vector), therefore, when we consider a road network as a graph, especially the travel direction of each edge is vital for further data utilization, the road network should be represented as a "digraph" (Pung et al. [2022]), meaning that edges point in one direction from one node to another node, edges representing one-way streets in road networks. As a network is directed, nodes have two different degrees, the in-degree, which is the number of entering edges, and the out-degree, which is the number of exiting edges (Degree distribution).

In some applications, such as navigation and traffic simulations, with the direction of edges, the predecessor/successor concept is used to describe the more complex linking mechanisms for representing traffic logic, including relations of individual lanes to their adjacent lanes (Beil et al. [2020]), an explicit representation of predecessor/successor relations can be modelled as shown in Figure 2.6. The topology relationships are crucial for understanding traffic movement and routing within the network.



	Predecessor	Successor
A	B	B
B	A	A, C
C	B	-

Figure 2.6.: Linear representations of predecessor/successor relations in proposed CityGML 3.0. Figure from Beil et al. [2020].

2.3. Road network in OpenStreetMap

OSM is a globally crowdsourced, open-access geospatial dataset maintained by a community of volunteers who contribute data about roads, trails, and various points of interest worldwide, OSM allows anyone to edit and update the map, fostering a dynamic and frequently updated resource. The dataset is composed of various elements: nodes (representing points), ways (linear features like roads and area boundaries), and relations (which define how these elements are interconnected) (OSM-elements). Each element can carry tags that provide detailed attributes, such as road types or land use. One of OSM's significant advantages is its openness and global reach, making it a highly accessible and detailed resource that often outpaces commercial and governmental datasets in reflecting real-world changes. The quality of OSM data has improved substantially over time, making it a valuable resource for research and analysis of urban street networks (Barrington-Leigh and Millard-Ball [2017]).

2.3.1. OSM road network structure

OSM road centerline An OSM road centerline in OSM is modelled as a polyline, and is specified by a collection of nodes (points, vertices), the road segments connecting these nodes. Each segment is assumed to be the shortest line on the Earth's surface (negating terrain issues) connecting its two ends (OSM-way). The nodes that define the geometry of the road are listed in the proper sequence, and the direction of the vector indicates the forward travel direction of the road.

OSM road tags Each OSM road centerline typically represents a section of a physical road on the ground road and is tagged with various attributes that describe the road's characteristics. Tags are attached to its basic data structures (its nodes, ways, and relations), a tag consists of two items, a key and a value, and each tag describes a geographic attribute of the feature. These tags are pivotal for understanding transportation complexity as they provide the necessary and auxiliary metadata about road geometry and associated features. Since OpenStreetMap's tagging system allows users to describe features with unlimited attributes, the tagged features of different roads and sources can be quite uneven. To support the road modelling reconstruction, we focus on the most commonly used tags, and most features can be described using only a small number of tags, the usage of these tags is global. The primary feature that is related to the road network in OSM is **Highway**, the selected tags that might contribute to the road model reconstruction are as follows:

1. *highway*=*, this tag specifies the traffic types for the road network as its value, from most to least functionally important for motor vehicle traffic; also labels the link roads, special road types, designated cycleways and paths.

2. *cycleway*=*, this tag in OSM aims to distinguish whether the cycling area is located on the main roadway or lane, and labels the condition as its value. A cycle lane lies within the roadway itself (on-road), whereas a cycle track is separate from the road (off-road). Note that a cycle track may alternatively be drawn as a separate way next to the road which is tagged as *highway=cycleway*.

3. *oneway*=*, this tag describes whether a road permits bi-directional traffic. The value can be stored as three annotations: "yes", "no", and "reversible".

4. *name*=*, the name of the road in the real world.

5. *junction*=*, describes how a specific junction is constituted. The most common value is "roundabout", to describe the tagged closed way as a roundabout in the network. This automatically implies *oneway=yes*, the oneway direction is defined by the sequential ordering of nodes within the road centerline.

6. *lanes*=*, this tag uses an integer as the value, which is used to specify the total number of lanes of a road. It includes several types of traffic lanes suitable for vehicles wider than a motorbike, general-purpose traffic, bus lanes, etc. And it excludes the lanes dedicated and marked for parking, and the narrow lanes that are used for cyclists (tag *cycleway=lane* for those). The contribution rule of roads requires that if the number of lanes changes, it is necessary to split the OSM road into two road centerlines. This should be done when a new lane has started (regardless of width), or a lane has finished disappearing (usually a merge with another lane).

7. *turn*=*, this tag key can be used to specify the indicated turn or merge direction for a way or lane. The key is predominantly used with the lanes suffix *turn:lanes*=* to tag indicated turn markings for individual lanes at intersections or other junctions, corresponding to the road markings on the lane surface in any real situation where a manoeuvre is signed. The order of the *turn:lanes*=* value is represented by starting with the leftmost lane and ending with the rightmost, with each lane-value separated by a "|" (vertical bar). On bi-directional ways, the turn indications for one direction of the road can be specified using the *turn:forward*=* or *turn:backward*=* suffixes. .

8. *bridge*=*, used to describe that a way is on a bridge. *bridge=yes* is a non-specific bridge tag, possibly combined with other *bridge*=* tags, the value of the other tags is used to indicate the type of the bridge.

9. *tunnel*=*, is used for the lower way of some grade-separated crossings. In general, if the lower way is long and surrounded by earth it is almost certainly a tunnel.

Tag key suffixes Due to the complexity of road attributes, when adding details to roads, it is often important to differentiate between the direction of travel or the side of the way, four terms *forward*, *backward*, *left* and *right* are defined in OSM as the additional information of the primary tags. The tag values (or key suffixes) *forward* and *backward* describe a direction along a way, but not a side of the way. The code *forward* means the direction in which the way is drawn in OSM, while backward means the opposite direction. The tag values (or key suffixes) *left* and *right* describe a side of a way, but not a direction along the way. *left* means the left-hand side of the way when looking in the forward direction (as defined above), while *right* means the right-hand side also when looking in the same direction. Note: *:both* as a suffix is used to explicitly say it applies to both sides. Examples: *lanes:backward*=* is the number of lanes in backward direction; *turn:lanes:forward*=* is the turn indications for each lane in the forward direction; *cycleway:left*=* is a cycleway on the left side of the road.

2.3.2. Road representation and modelling in OSM

Linear and areal representation There are different methods for modelling the same real-world road features in OSM. Linear representation as the road network has no geometrical representation in nodes of its (change in) width, but perhaps only a tag *width*=* stating an average width (OSM-way). Therefore with a linear representation a mapper must choose a centerline, for which there can be various options to decide the exact position of the drawn centerline, especially when the feature itself is asymmetrical in its cross-section (options: centre of carriageways vs centre of carriageways + sidewalk vs location of directional markings on carriageways).

Aside from the uncertainties related to road width and centerline accuracy, there are additional challenges with the geometric correspondence in linear representations. For example, when different traffic types intersect, the linear representations often extend continuously over the crossing (i.e., a physical junction) instead of adding a node to represent the intersection and splitting the road segments. This mapping approach can complicate the identification of junctions, changes in road width, and other variations in the physical shape of the road space, as these features are not accurately or clearly reflected in the OSM linear representation alone.

While the areal representation in OSM for a road is tagged in the key *area=yes*, and represented as a polygon, this tag is seldom necessary, thus it is rarely used in road mapping, even though it gives a more accurate description of the actual shape of a road.

Conclusion of OSM road representation

- **Relation of linear center and areal center:** The OSM road centerline can encapsulate various driving conditions, including one-way streets, two-way streets, and even separate cycleways. The centerline is a simplified representation that abstracts the complex geometry of multiple-lane centerlines into a single line, therefore, it is important to note that the road centerline is not always positioned at the geometric center (center axis) of its corresponding 2D road polygon. Additionally, based on the mappers' options encountering the asymmetrical roads, the curvature of the road centerline typically reflects the direction and movement of 'through' traffic, even in cases where the road includes turning lanes or bike lanes. This simplification can be misleading, particularly in scenarios where road width changes and turning movements are significant.

- **Discrepancy of vector direction and travel direction:** The use of *forward* and *backward* tagging rules introduces a level of uncertainty. These tags are employed by mappers to provide additional traffic information, but they can create confusion during data interpretation for routing and analysis. Specifically, when a bi-directional road centerline connects with other roads (whether single-directional or bi-directional), the vector directions may contradict each other. Despite this, the opposite directions are tagged to represent continuous traffic flow in the real world, complicating the accurate depiction of traffic dynamics, see Figure 2.7. In the aggregate, users cannot clarify the connectivity of traffic flow by using only the vector of the bi-directional road centerline.

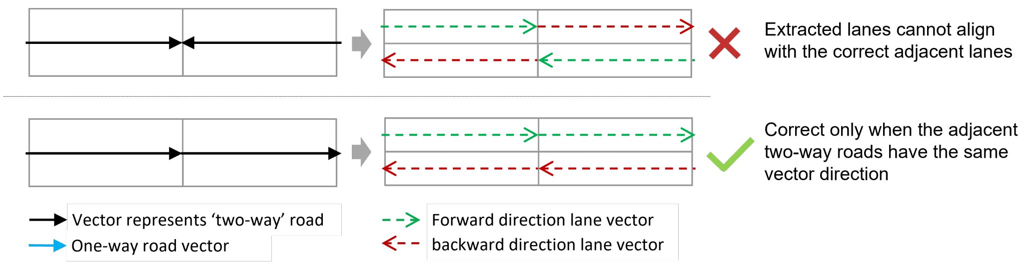


Figure 2.7.: Unalignment of lanes caused by the uncertainty of vector direction.

- **Crossing and missed intersection mapping:** Road segments, particularly in the domain of intersections, should be differentiated from road centerlines. Intersections can be classified based on the number of road segments (intersection legs) involved, but some intersections are represented as road centerlines without a real intersection (i.e. no shared node), the intersections have 0 road segments (intersection legs), as shown in Figure 2.8. This method of modelling crossing traffic flows may lead to uncertainty and inaccurate identification of the intersection area in the physical world.

Difference results of road centerline application for road polygon generation Based on the above understanding of OSM road structure and representation methods, we observe that the unclear relationship between road linear and areal representations, along with the complex conditions of road space and traffic, leads to varying outcomes when using road centerlines in different approaches to generate road polygons. The schema in Figure 2.9 shows the centerline representations and corresponding road polygon's generation results.

1. **Centerline as movement shape representation**: If the entire road segment is treated as a representation of the shape of the movement, all lanes will align parallel to the driving curvature. In cases where the road contains multiple lanes, the road centerline may not coincide with the geometric center of the road polygon, nor with the center of

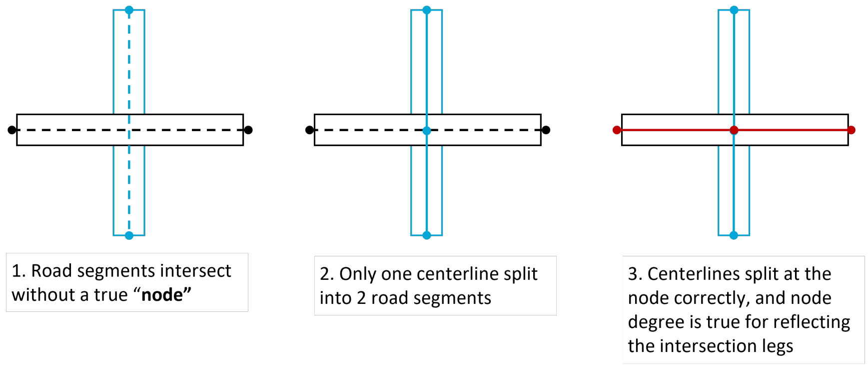


Figure 2.8.: Uncertainty of the intersection labelling.

the polygon's edges. This method better reflects the actual driving paths but requires careful consideration of the centerline's placement within the overall road structure.

2. **Centerline vertices at the center axis of road polygon boundary**: Alternatively, when the start and end vertices of a road centerline are treated as if they are located at the centers of the two short edges of the road polygon, the resulting lane geometries may become deformed. This approach can alter the trajectory of vehicle movements, leading to inaccuracies in representing the actual flow of traffic.

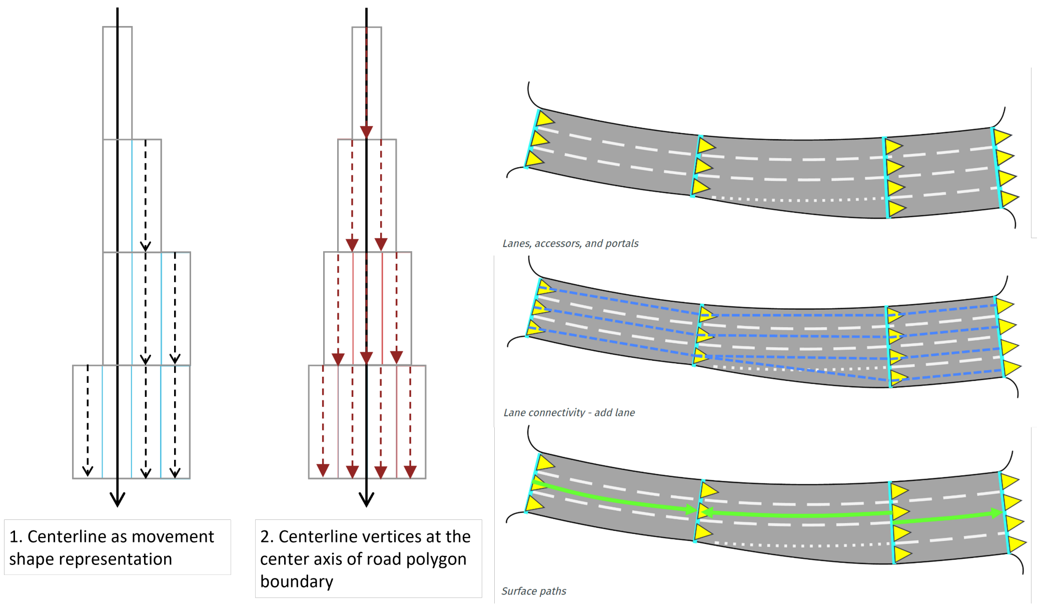


Figure 2.9.: Illustration for the difference outcomes by using road centerline to generate road polygons

2.4. Important Geometric Characteristics of Road Network

2.4.1. 'Strokes' concept in road network

Based on the principle of 'Good Continuity', Thomson and Richardson [1999] proposed a concept termed **"strokes"**. When observing a road network or river network, natural linear elements will be seen which extend through junctions and maintain the topology and geometry connectivity, the aggregation of these elements is the "stroke": "a set of one or more arcs in a non-branching and connected chain". Strokes represent the natural functional units of a network and reveal its hierarchical characteristics (Yang et al. [2011]). Using this concept in the road network analysis, the definition of a stroke is a group of road segments that are continuous at junctions, and they have the same road types and can move through the network with no abrupt change in direction, or they intersect at a small angle (as shown in Figure 2.10). These road segments as strokes are one of the fundamental elements of road network structural analysis, usually representing the high hierarchical roads and main traffic flow in the transportation since the good continuation of them.

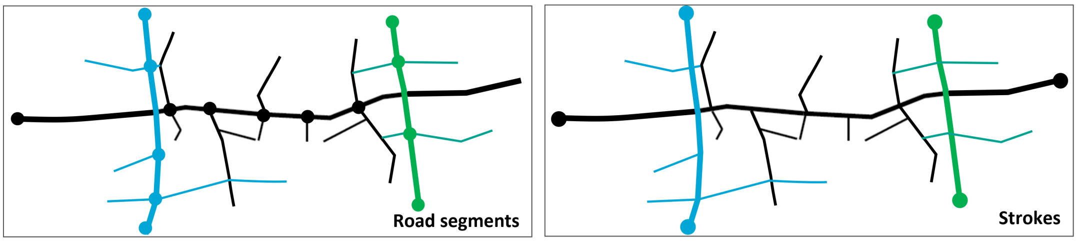


Figure 2.10.: Road segments and strokes structure

Since the **"strokes"** concept was proposed, there are many researchers who have used the *stroke-based* approach to conduct analysis of road networks. Strokes are products of a higher-level aggregation of road segments that can reflect functional importance and perceptual significance, the characteristics it has form the undeniable importance for network analysis, such as road network simplification, spatial pattern recognition, and road network spatial target matching (Yang et al. [2011]; Pung et al. [2022]; Yang et al. [2014]). Utilizing the hierarchical structure and the connectivity determined by topological and geometrical factors, studies work on the multi-scale representation of road networks and road network selection (Li and Zhou [2012]; Benz and Weibel [2014]), which are vital for map generation and geo-related visualization.

2.4.2. Geometric design of highways and streets

In transportation design, geometric design is crucial for ensuring that roads are both efficient and safe. According to the introduction in AASHTO [2011], the number of lanes on a road, particularly on main routes of freeways or streets, is primarily determined by traffic volume. Lane numbers cannot change randomly; they must adhere to specific rules that significantly impact the geometric characteristics of roads, especially in transitional areas such as lane-changing parts and intersections. These design principles are not only regulated in freeway construction but also have a dependency that can be applied to various types of roads. These rules as background to guide us in understanding the various changes in the road network, and facilitate the reconstruction method design.

Basic number of lanes The designation of the basic number of lanes is essential for determining the lane configuration on a road. It is important to maintain consistency in the number of lanes along any arterial route. The basic number of lanes refers to the minimum number of lanes that are designated and sustained over a considerable length of the route, regardless of changes in traffic volume and lane-balance needs. It represents a constant number of lanes assigned to a route, excluding any auxiliary lanes.

Principle of lane balance To ensure efficient traffic flow through and beyond an interchange, it is crucial to maintain a balance between the number of lanes on the freeway and its ramps. The basic number of lanes on the highway, as well as the minimum number of lanes on ramps, are determined by design traffic volumes and capacity analysis. This lane configuration should be consistent over a significant stretch of the freeway and should not be altered between interchanges, simply because there are substantial volumes of traffic entering and leaving the freeway. In essence, continuity in the basic number of lanes must be preserved after the merging or diverging of the interchanges (typical examples as shown in Figure 2.11).

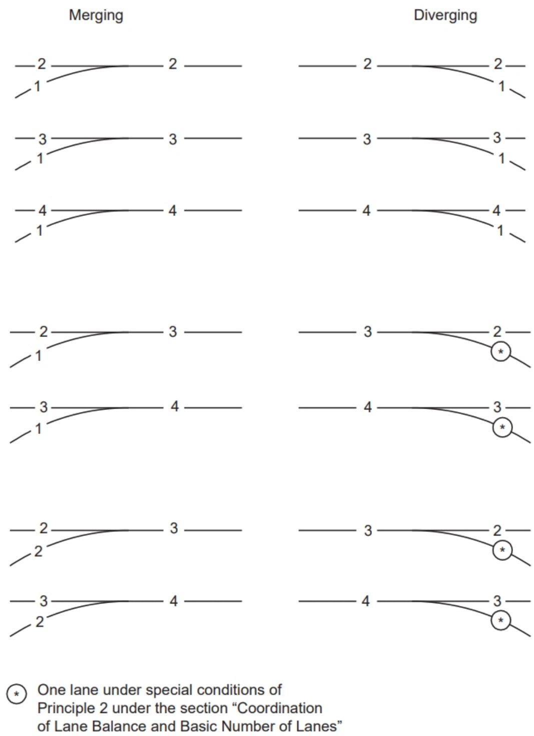


Figure 2.11.: Typical examples of lane balance. Figure from AASHTO [2011]

Auxiliary lanes After introducing the 'basic number of lanes' and 'lane balance', we must also consider the auxiliary lanes that form the transition between the main route and the ramp. An auxiliary lane is a section of the roadway adjacent to the through lanes, designed for purposes such as speed changes, turning, storage for turning, and other functions that support the movement of through traffic. The width of an auxiliary lane should match that of the through lanes. Auxiliary lanes may be added to maintain lane balance, meet capacity requirements, or facilitate turning, speed changes, and the manoeuvring of traffic entering or exiting the main route of a roadway.

In Figure 2.13, several examples illustrate the coordination between lane balance and the basic number of lanes through the use of auxiliary lanes. The consistent number of lanes on the main route—before entering the auxiliary lane and after passing the ramp—highlights the importance of maintaining the minimum basic number of lanes to ensure **route continuity** in traffic flow, regardless of variations in road configuration. Additionally, by integrating the concept of **"strokes"**, where the main traffic route is simplified into a single road segment, each segment represents a partition of the stroke. Therefore, in the reconstruction of road networks, incorporating the rules and correlations related to the minimum number of lanes within a stroke, along with auxiliary lanes and ramps, is essential for ensuring topological and geometric connectivity. These principles reflect the foundational aspects of transportation design in the real world.

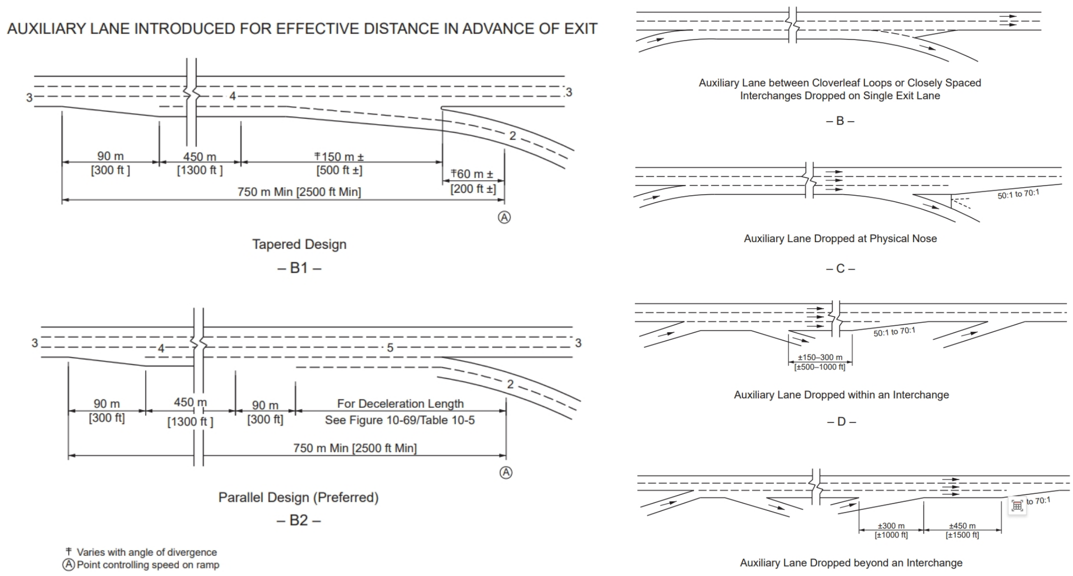


Figure 2.12.: Coordination of lane balance and basic number of lanes through application of auxiliary lanes. Figure from AASHTO [2011].

2.4.3. Turning paths for roadway designs

In roadway design, understanding the turning paths of vehicles is crucial, as these paths define the geometric parameters necessary for safe and efficient road layouts to accommodate different vehicle manoeuvres. Key elements include the minimum centerline turning radius (CTR), wheelbase, outer track width, and the path of the inner rear tyre during turns

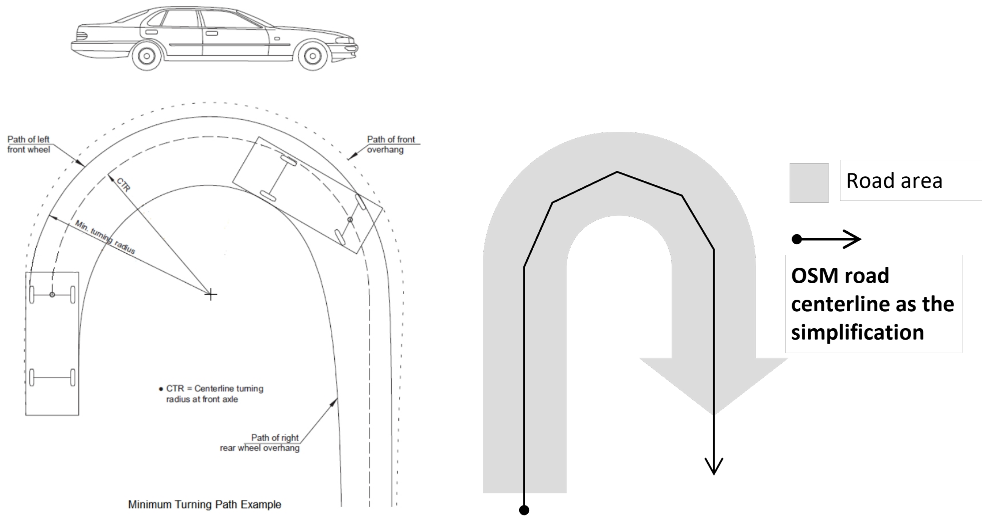


Figure 2.13.: Left: minimum centerline turning radius (CTR), fgure from Taylor. Right: OSM simplification of road curvature.

(Taylor). These dimensions are critical for ensuring that vehicles can navigate curves and corners without encroaching on adjacent lanes or roadway boundaries (Figure **??**).

In the context of OSM road data, which often simplifies real-world roads into linear segments, these turning paths curves and corners as a series of short segments, with the turning radius effectively encoded as the interior angle between these segments. If the interior angle between segments is too small, it may incorrectly suggest a tight turn, when in reality, the segments might represent intersecting roads rather than a continuous turning path. This distinction is crucial during the reconstruction process, as mistaking an intersection for a turn could lead to incorrect assumptions about traffic flow and road connectivity. Therefore, accurately interpreting these angles and considering the geometric design principles of turning paths is essential for producing a road network model that reflects true traffic movement and connectivity.

2.5. Summary of road network background

This **"Background"** chapter highlights key points that influence the design and implementation of methods for road network modelling, which serves as preconditions for further discussion of the study.

The complexities inherent in road networks have been explored throughout this chapter. The variations in traffic demand often lead to significant changes in road geometry, which may not be fully captured by linear representations like road centerlines or segments. The structure of OSM data to store related information is intricate, and without proper data processing, it can obscure the details required for accurately reconstructing these changes.

As a graph-like model, a road network inherently possesses topological features such as driving directions on main routes, road merging, diverging, and intersections. However, to meet the objectives of this study, focusing solely on topological continuity is insufficient. It is equally important to consider geometric relationships and semantic details to develop a high-level LoDs road network model.

Transportation geometric design rules serve as the fundamental for constructing the physical road features and the understanding of reconstructing road networks in a digital model. The mapping of road centerlines can vary due to the asymmetrical shapes of roads, making geometric design rules critical for generating a fidelity model. Establishing correlations between roads, lanes, and road surfaces is essential for accurate procedural reconstruction. Concepts from different hierarchical levels of road network structure, such as road-level strokes combined with lane-level route continuity, should be integrated to achieve a comprehensive model.

3. Related work

3.1. Categories of related road network researches

Based on the differences in data sources, methodology, and research goals, we focus on the related studies of road network model reconstruction and generation that are instructive for our research objectives, thus we summarize these related work into the following categories:

1. **Stroke-based methods**: These methods focus on generating strokes, extracting the main strokes from a road network, or simplifying linear representations of roads hierarchically, to analyse the characteristics of road networks efficiently.
2. **OSM-driven methods**: Leveraging OSM data to generate road networks, this category can be divided into two subcategories:
   a) *Generate 2D road network from OSM road centerlines*: The studies and projects are focused on creating 2D road models and developing visualization software capable of generating digital maps using OSM road datasets.
   b) *2D road network generation by integrating OSM data with additional datasets*: The research focus on enhanced 2D road network generation methods by integrating OSM data with other source datasets can produce improved results with varying levels of detail, tailored to specific purposes. The outcomes differ according to the characteristics of the combined datasets.
3. **Geometry-driven methods**: Focusing on the geometric representation of roads, this category includes:
   a) *2D Areal Road Network Generation*: These studies explore methods for transitioning from linear to areal road representations, addressing intersections, connected road ribbons, and road geometry refinement.
   b) *2.5D and 3D Road Network Models*: Research in this area explores the creation of elevated road models using elevation data, though there are fewer studies compared to 2D road generation.

3.2. Stroke-Based Studies

The primary objective of research on road networks is to transition from individual road centerlines or segments to more abstract, hierarchical representations known as strokes. Strokes are intended to capture the continuity and hierarchical structure of road networks by grouping road segments across junctions and analyzing the network's geometric and topological properties.

Several studies have explored different methods for stroke generation, focusing mainly on two key factors: semantic properties and geometric calculations Figueiredo and Amorim [2004]; Thom [2005]; Thomson [2006]; **?**. Semantic properties such as road names and functions Jiang and Claramunt [2004]; Tomko and Winter [2008], help categorize roads based on their relative importance within the network. However, the geometric approach, particularly the calculation of angles between adjacent road segments, has been emphasized by Tripathy et al. [2020], due to its ability to handle cases where semantic information is incomplete or inconsistent.

In the existing literature, such as the studies Yang et al. [2011]; Tripathy et al. [2020] , the detailed discussion of intersecting angle calculations highlights the intricate changes in movement within road networks and validates the use of geometric characteristics for analyzing "good continuity".

The algorithm of an open-source tool ***COINS*** proposed by Tripathy et al. [2020] only chooses the geometric factor for generating street hierarchies. It uses the OSM data in India, and begins by splitting raw street data into individual line segments, each assigned a unique ID. It then identifies adjacent segments at both ends of each line, calculates their orientations, and determines the interior angles between them. Using these angles, the algorithm selects the best fit for each segment (Figure 3.1). This approach demonstrates that geometric continuity is directly linked to the topological correctness necessary for accurate road network generation and reconstruction.

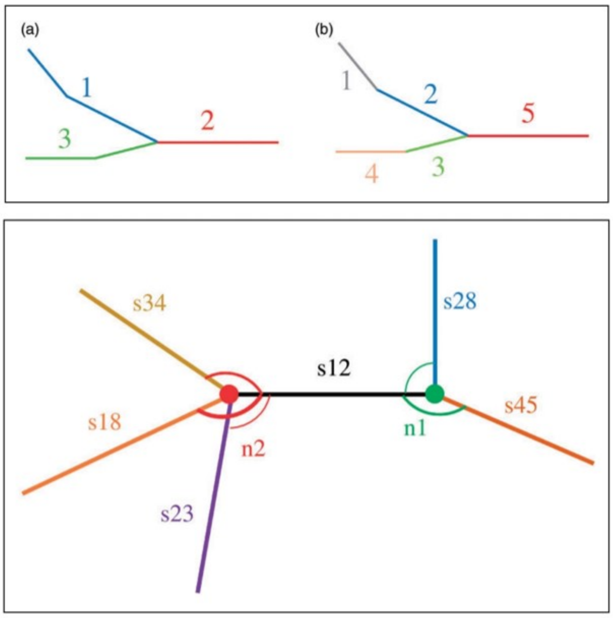
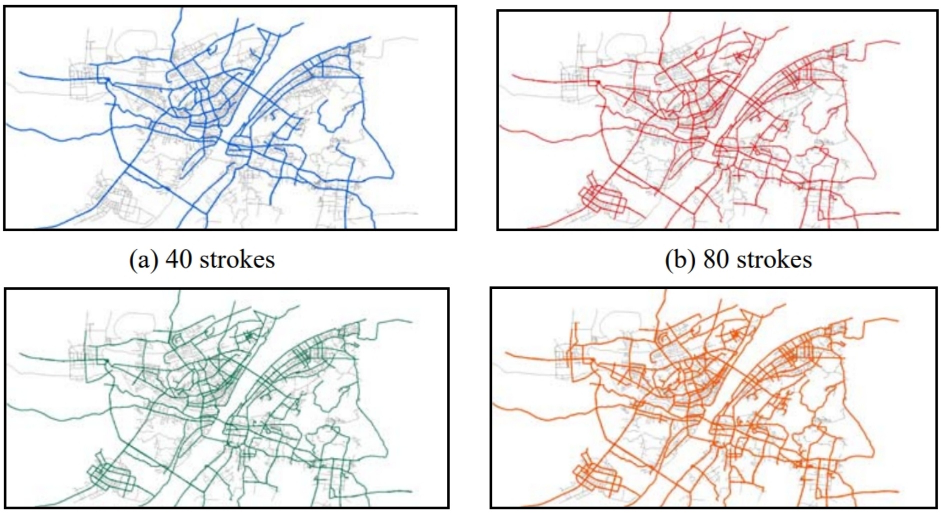


Figure 3.1.: Angle calculation of ***COINS***.

Based on the definition of strokes, all the studies emphasize that the outcomes of stroke generation are global, resulting in a complete hierarchical structure for the entire network. This process typically operates at the road level, focusing on the centerlines of motorized traffic. However, there is a noticeable gap in the literature concerning the application of these methods to cycling networks or other non-motorized forms of transportation.

In terms of road network model generation objective, there are limitations to this **stroke-based** approach, a significant issue is that the stroke hierarchy, while perceptually significant (refer to some stroke generation results in Figure 3.2), may overlook some of the topological relationships within the network. The global nature of stroke generation can obscure the real-world traffic flow directions, as the new hierarchy is based more on perceptual importance than on actual traffic dynamics, potentially leading to an incomplete representation of the road network's topology.



(a) 40 strokes (b) 80 strokes

(a) 120 strokes (b) 160 strokes

Figure 3.2.: Stroke generation results.

3.3. OSM-Driven Studies

3.3.1. Generate 2D road network from OSM road centrelines

Projects like **"abstreet"**A/B Street and **"osm2street"**A/B Street/ osm2streets contribute to the creation of detailed 2D road models and digital maps through OSM road datasets. The **"abstreet"** project serves as a comprehensive transportation planning and traffic simulation software for creating cities friendlier to walking, biking, and public transit, **"osm2street"** provides a simplified street network schema, emphasizes the creation of 2D road models from OSM road centrelines, (Figure 3.3). Figure 3.4 illustrates how to utilize the OSM *tags* and reorganize the relations of the exact lane with its corresponding traffic direction and width. With the correct number of lanes, a complete road polygon is reconstructed. This approach improves the accuracy of lane polygon and road polygon, as well as the semantic information details of traffic types.

Figure 3.3.: Screenshot of ”osm2street”, output the road polygons for a chosen region.

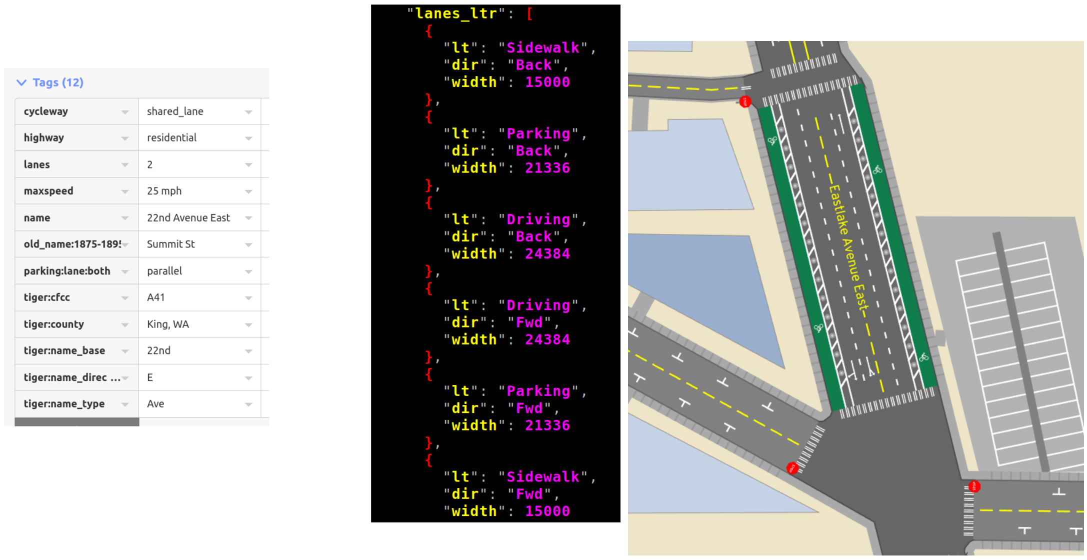


Figure 3.4.: **A/Bstreet** road polygon generation method

The project ”The Neukölln street maps” focused on showcasing how detailed mapping of urban environment and street lane infrastructure – especially for bike and foot traffic – can be done with OSM OpenStreetMap Berlin, emphasizing the importance of detailed attributes in OSM when rendering for specific elements like bicycle lanes (Figure 3.5, 3.6). Furthermore, the paper titled ”OSM-Based Automatic Road Network Geometry Generation in Unity” Yu [2019] presents a method for automatic road network geometry generation using OSM data in Unity, demonstrating the applicability of OSM for 2D road network creation in diverse contexts.

Figure 3.5.: Screenshot of "Detailed rendering of bicycle lanes and junctions as part of the OSM 'Straßenraumkarte'".

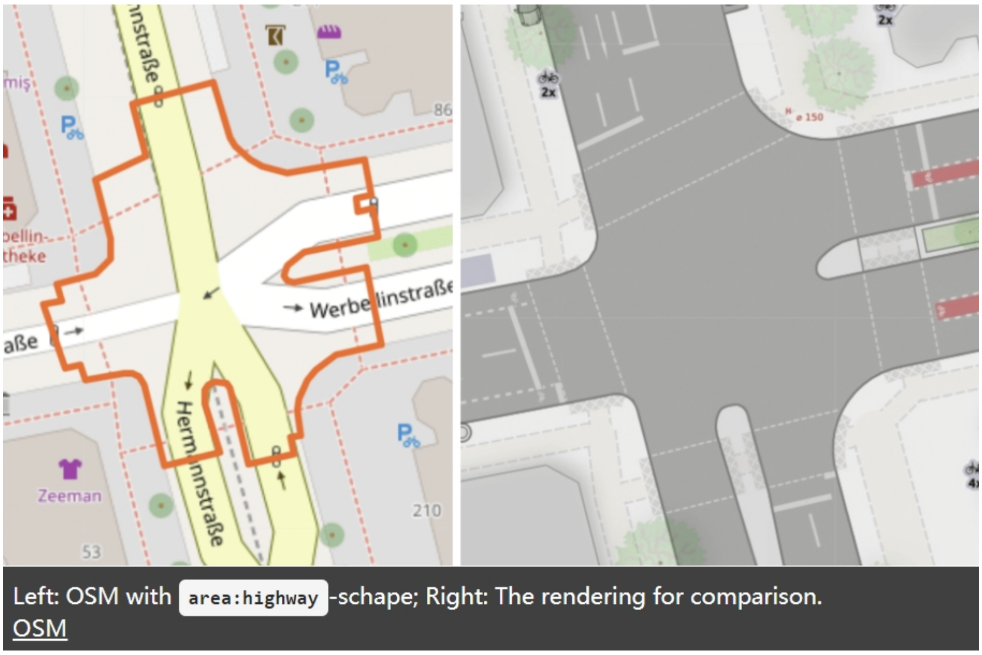
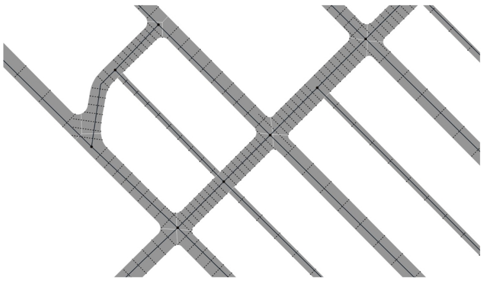


Figure 3.6.: Using the 'area:highway' tag to render detailed intersection. [OpenStreetMap Berlin]

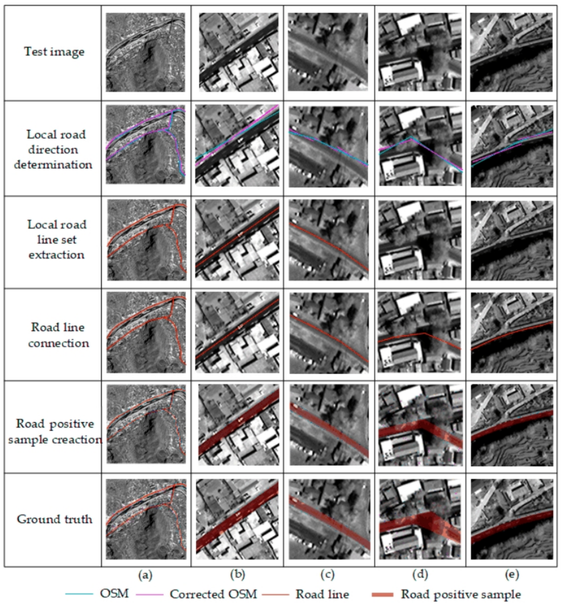
In summary, these projects and studies, while diverse in their specific focuses, converge on the common point of utilizing OSM data for the generation of 2D road networks. Whether developing toolkits, enhancing visualizations, or leveraging OSM for automatic geometry generation, these endeavours collectively underscore the significance of OSM as a foundational data source for 2D road modelling within 3D city environments. The results vary depending on the amount of information available and the method used to extract it for the final outcome.

3.3.2. Enhanced 2D road network generation methods by integrating OSM data with other source datasets

The paper titled "From Road Centrelines to Carriageways—A Reconstruction Algorithm" introduces a methodology for creating carriageways based on OSM's centrelines and open access areal representations (Figure 3.7a), showcasing the effectiveness of this approach for worldwide applicationVitalis et al. [2022]. Another noteworthy work, "An OSM Data-Driven Method for Road-Positive Sample Creation," proposes a novel method for creating accurate road-positive samples for deep learning using OSM data and orthophoto images (Figure 3.7b), incorporating road homogeneity constraints and texture features**?**. In "Map Matching and Lanes Number Estimation with OpenStreetMap," a new method for estimating the number of lanes using low-precision GPS data and OSM is presented[Kasmi et al. [2018]]. Additionally, the MSc. thesis on "Safety-Driven Road Width Estimations from Vector Data" introduces a novel approach for estimating road width using vector data, emphasizing the significance of considering road width for road safety management applicationsChatzidiakos [2021].



(a) OSM centrelines and road width calculation. [Vitalis et al. [2022]]



(b) OSM raw data, corrected OSM and road positive samples. [Dai et al. [2020]]

Figure 3.7.: OSM-based researches

These studies advance 2D road network generation by improving accuracy and detail, addressing specific purposes, and utilizing diverse fusion datasets. The approaches collectively use OSM-based methods and emphasize to how to address the misalignment between OSM and additional datasets and modify the encountered errors. Overall, they demonstrate innovative approaches that integrate OSM data with other datasets to enhance 2D road network generation for improved detail and purpose-specific tailoring.

3.4. Road network 2D geometry generation

This section highlights studies focused on generating 2D polygons as areal representations of road networks. Similar to the OSM-driven methods discussed earlier, most approaches use road centerlines as input data and assign width values to transform them into polygons. To accurately represent road shapes and create error-free digital models, two key generation steps are essential and cannot be overlooked:

1. **Road Boundary Mismatch**: When adjacent roads have different lane counts or widths (e.g., different traffic types), generating proper transition areas is crucial.

2. **Intersection Polygon Generation**: Studies propose various methods to distinguish intersections from regular road segments to ensure that road components and structures are well-represented.

3.4.1. Road width transition part generation

Handling changes in road width requires methods that generate realistic transition polygons. These can be grouped into two approaches. First, road-level modification: This approach emphasizes geometric correctness at the road boundary level. In "STREETGEN: In-Base Procedural-Based Road Generation" (Cura et al. [2015]), the authors introduce a variable buffer to smooth transitions between roads with different widths. It also offers the advantage of being able to control the three most classical transitions (symmetric, left, and right) and the transition length using only the street axis. The variable buffer adjusts the width along the street axis, ensuring more realistic transitions between road segments using circles and trapezoids (Figure. 3.8). Another study Xu et al. [2023a] tackles seamless lane transitions by introducing a "Road Connector" function. In cases where these two roads feature differing lane counts, the function generates a line segment along these roads from the node to which they are both connected. Subsequently, it produces a buffer based on the width of the road (Figure. 3.9). Second, lane-level modification: This method focuses on lane polygons, refining them to correct asymmetrical lane shapes, often due to auxiliary lanes. As the method proposed by HERE Technologies -Lane Model, the method starts with generating matching lane centerlines for consecutive roads. Asymmetric lanes typically found on one side of the road as auxiliary lanes, are offset to form lane polygons. The final road polygons of the consecutive roads often show mismatches in the center axis due to the different lane shapes (Figure. 3.10).

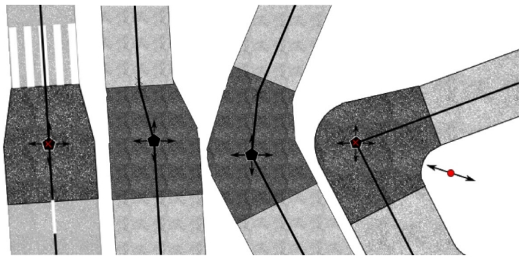


Figure 3.8.: Variable buffer for robust roadway width transition

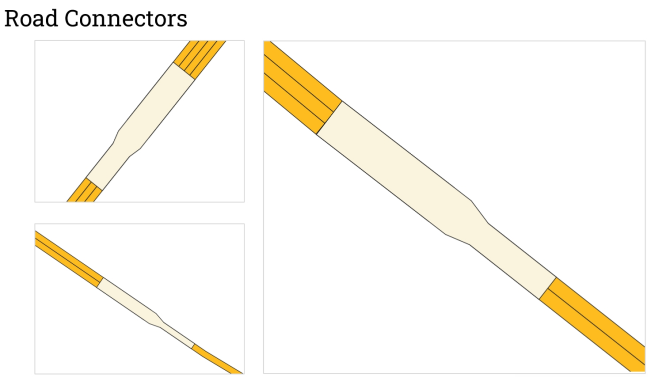


Figure 3.9.: Road connector

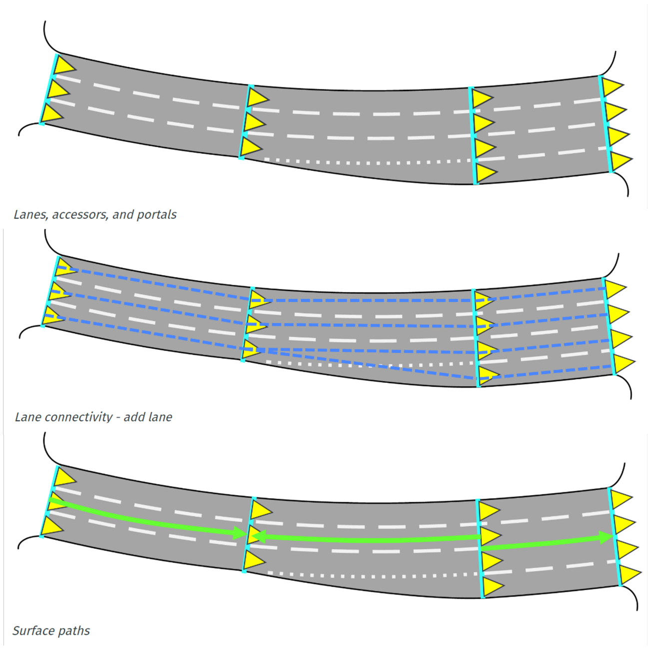


Figure 3.10.: From lane connectivity to road polygons (HERE Technologies -Lane Model)

3.4.2. Intersection polygon generation

Since these studies proposed geometry-driven methods, the intricate traffic flows within the intersection region are not the focus, they aim to generate the intersection boundary and split the intersection polygon and converged road polygons by using the boundary. There are two key steps for the intersection geometry generation: 1) **Address overlapping or collision parts**: Trim the road geometries where roads overlap. 2) **Create boundary arcs**: Define the boundaries of intersections with arcs or curves.

In A/B Street, the approach uses road centerlines from OSM, calculating left and right boundaries to handle overlaps. When roads collide, the method trims them back to avoid overlap, ensuring that roads meet the intersection at perpendicular angles (A/B Street-Geometry). This approach results in intersections without refined turning radii (Figure 3.11).

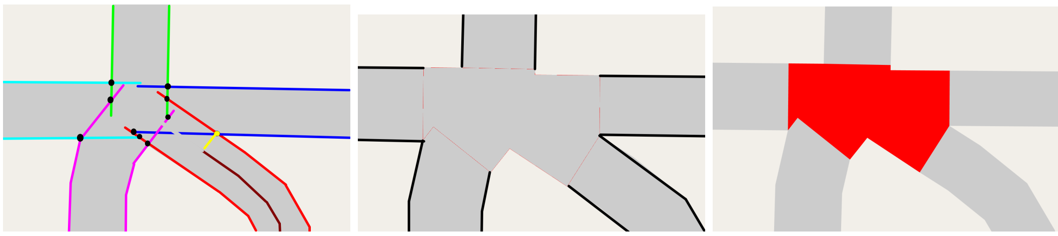


Figure 3.11.: **A/Bstreet** intersection generation method

In StreetGEN (Cura et al. [2018]), the intersection geometry is guided by civil engineering principles, emphasizing smooth corner transitions for vehicle safety. Road turning corners are modelled as arcs, typically circular, to prevent sharp turns (See Figure. 3.12). The method calculates turning radii based on road types and generates arcs for the intersection boundaries using buffer and geometric operations. The process identifies the center of the arc by buffering the road axes and finding the closest points of intersection between the buffers. Surfaces of intersection are generated using the calculated arcs, then use the "ST BuildArea" function (See Fig. 3.13b). Given a set of geometries, it breaks all the geometries into polylines, and then creates the largest possible surface from those polylines.

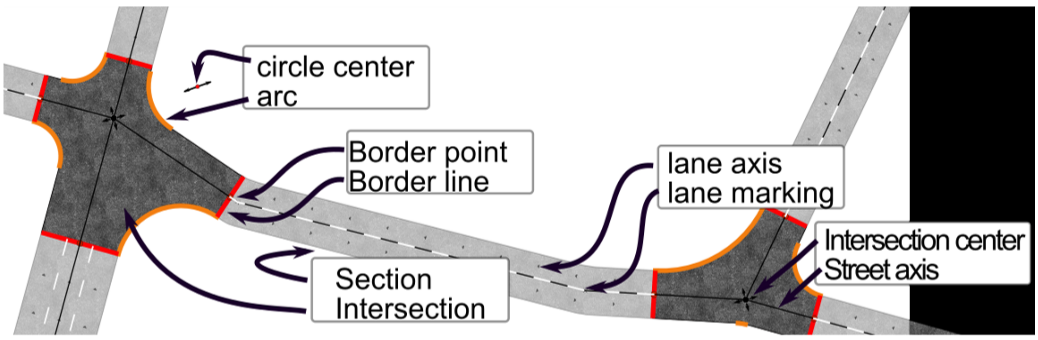
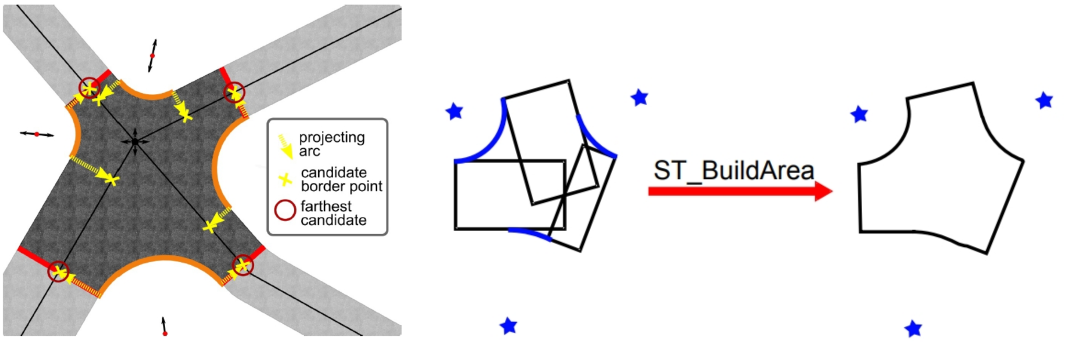


Figure 3.12.: StreetGen intersection generation schema



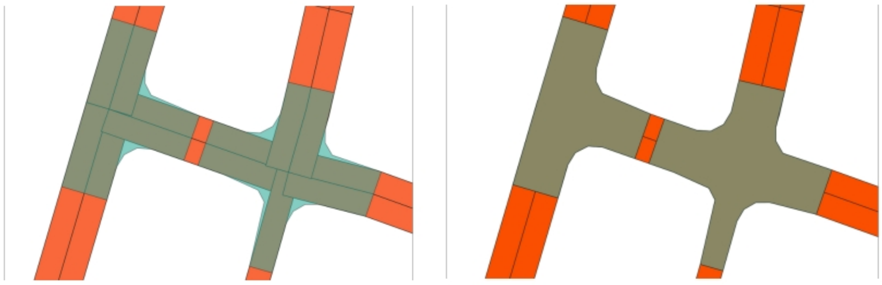
(a) StreetGEN intersection key factors illustration.

(b) Function ST_BuildArea, integrate cutted polygon and arc to build the maximum area possible.

Figure 3.13.: StreetGEN intersection generation details

A third method, proposed by Xu et al. [2023a], generates turning corners using a ”double-buffer” technique. Unlike StreetGEN, this method does not compute an exact turning radius. Instead, it applies a positive buffer followed by a negative buffer at the same distance to smooth out the intersection boundary. The generated result is similar to the result from ”StreetGEN” with the approximate accurate turning arcs (See Fig. 3.14).

Once the intersection boundaries are defined, the final step across these methods is to clip and merge. This process trims unwanted overlaps between regular road geometries and intersection polygons, ensuring error-free road networks.



(a) Before clip

(b) After clip and merge

Figure 3.14.: Before and after performing clip operation to polygons near intersection

4. Methodology

Based on the research objective defined in Chapter 1, background knowledge in Chapter 2, and the previous researches presented in Chapter 3, we propose our methodology in this chapter. Our input data is the road network represented as raw OSM road centerlines, which typically contain unwanted roads and attributes, but are expected to convey the major characteristics in topology, geometry and semantics of the road network. To reconstruct the road network in lane-level details and achieve the 2D areal representation, we leverage the encoded attributes and the existing topology structure to design methods that span from linear representations as polylines at road level to road ribbons and intersection geometries as polygons in areal representation. To improve the topological correctness and the geometrical correctness of the reconstructed road model, we modify the centerline geometry to address the issues and avoid inherent errors. Furthermore, the corrections of the derived lane topology as linear elements and areal elements are applied to optimize the outcomes.

The chapter is organized as follows: Section 4.1 provides an overview of the reconstructed approach; Section 4.2 introduces the different levels of network model sketches to interpret the essential elements of each model, and define the structured rules for the variety of data which need to be stored in the data models. Section 4.3 describes the OSM road data pre-processing approach applied to extract the strokes and create the adjacency relations; Section 4.4 presents the approaches and algorithms we use for generating the primary lane centerline results, as well as the modification and correction methods applied to lane geometry; Following the lane-level road network reconstruction, the method used for upgrading the linear to areal representation as road polygons is introduced in Section 4.5.

4.1. Methodological Framework

This section introduces the key stages of the methodology. Each paragraph in the first subsection describes a specific stage, outlining the goals, reasons of why, tasks, and outcomes of the process, leading to the next phase. A reconstruction pipeline is visually illustrated in Figure 4.1 and a detailed flowchart in Figure 4.2.

4.1.1. Key stages of methodology

1. Initial: Model Design and Method Design

- **Goal:** To establish a clear modelling framework that connects the road, lane, and areal levels of the road network and defines the crucial strategies for their reconstruction.
- **Why:** Establishing a clear modelling framework ensures that each level (road, lane, and areal) is systematically connected, allowing for a seamless transition between these representations. This stage is critical for defining the modelling strategies that will

guide each subsequent step, ensuring that topological consistency, geometric accuracy, and semantic enrichment remain priorities throughout the process.

- **Approaches:**
  1. Analyze the relationships between road-level and lane-level networks and design the overall model accordingly.
  2. Develop sketches for the road-level, traffic-level, lane-level, and areal representations.
  3. Identify key aspects that will guide the methodology: topological consistency, geometric accuracy, and semantic enrichment.
- **Outcomes:** A structured model design and three defined aspects that guide the entire methodology.

2. Preparation: OSM Road Network Initialization and Stroke Generation

- **Goal:** To prepare and clean OSM raw data as a road-level network by extracting relevant attributes, identifying road adjacencies, and generating grouped strokes that will serve as the basis for lane-level and areal representation.
- **Why:** OSM data contains a wealth of tags and embedded information that, if properly extracted and processed, can significantly enhance the representation quality. Extracting attributes like road types, widths, and traffic types is crucial for accurately modelling lane-level details. Without this step, the hidden complexities in OSM data would remain unutilized, limiting the ability to create accurate lane-level networks.
- **Approaches:**
  1. The process involves filtering OSM data, normalizing road geometries, and identifying topological relationships between road segments.
  2. Spatial analysis is applied to detect road adjacencies and intersections, after which road segments are grouped into continuous strokes.
  3. The attributes of these grouped strokes are recalculated and updated to the road networks accordingly.
- **Outcomes:** Cleaned and processed road networks with grouped strokes and updated attributes, ready for lane-level reconstruction and areal polygon generation.

3. Linear Generation: Lane Network Reconstruction

- **Goal:** To generate a lane-level network from the road-level network and grouped road strokes, ensuring topology consistency, geometric accuracy and semantic enrichment.
- **Why:** Even after data preparation, the initial linear representation for a road segment remains as a single road centerline, which does not capture the complexity of multi-lane roads. Reconstructing the lane network is essential for transforming these single-road centerlines into detailed, well-distributed lane centerlines. This step is the backbone of improving the LoDs of the road model, providing the necessary structure for generating accurate lane polygons and a more detailed 2D areal representation.

- **Approaches:**
  1. Lane centerlines are hierarchically reconstructed by extracting lanes from road strokes, considering road width and traffic type.
  2. Connectivity modifications ensure correct lane-to-lane adjacency, followed by topological corrections to fix any connectivity issues, focusing on both local and global traffic flow.
- **Outcomes:** A geometrically and semantically enriched lane-level network with accurate lane connectivity, ready for further processing in areal representation.

4. **Areal Generation: 2D Road Polygon Reconstruction**

- **Goal:** To accurately generate error-free 2D road polygons, capturing road shapes, widths, and intersection variations.
- **Why:** This stage involves the transformation from lane centerlines into accurate road polygons, focusing on identifying and correcting errors that can arise during the areal representation process. Despite high-quality linear data, issues like misaligned polygons and inaccurate road boundaries can occur when generating area-based representations. This step includes targeted methods for detecting such issues and refining the polygons, ensuring that road segments and intersections are accurately represented with complete boundaries.
- **Approaches:**
  1. A general polygon generation method is applied to create road and lane polygons.
  2. Post-processing corrects geometric errors using a node degree-based method to adjust polygon shapes and fix intersection issues.
  3. Intersection geometry is generated by merging road polygons, ensuring smooth transitions and accurate corner curvatures.
- **Outcomes:** A set of geometrically correct, detailed road and intersection polygons as the high-fidelity and high-LoDs 2D areal representation of the input road datasets.

4.1.2. Pipeline and workflow diagram

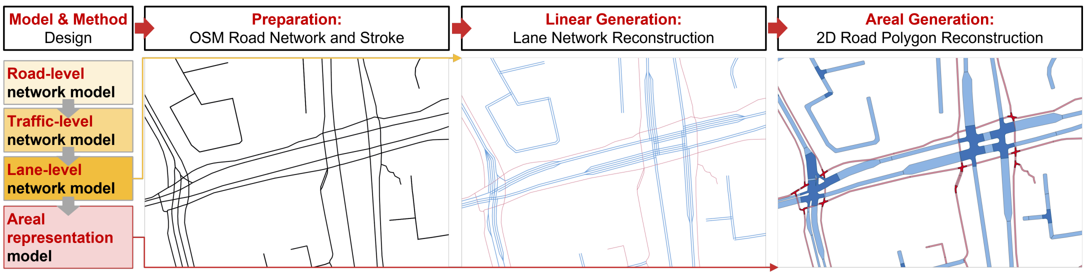


Figure 4.1.: Methodology overview

The pipeline is divided into 3 phases:

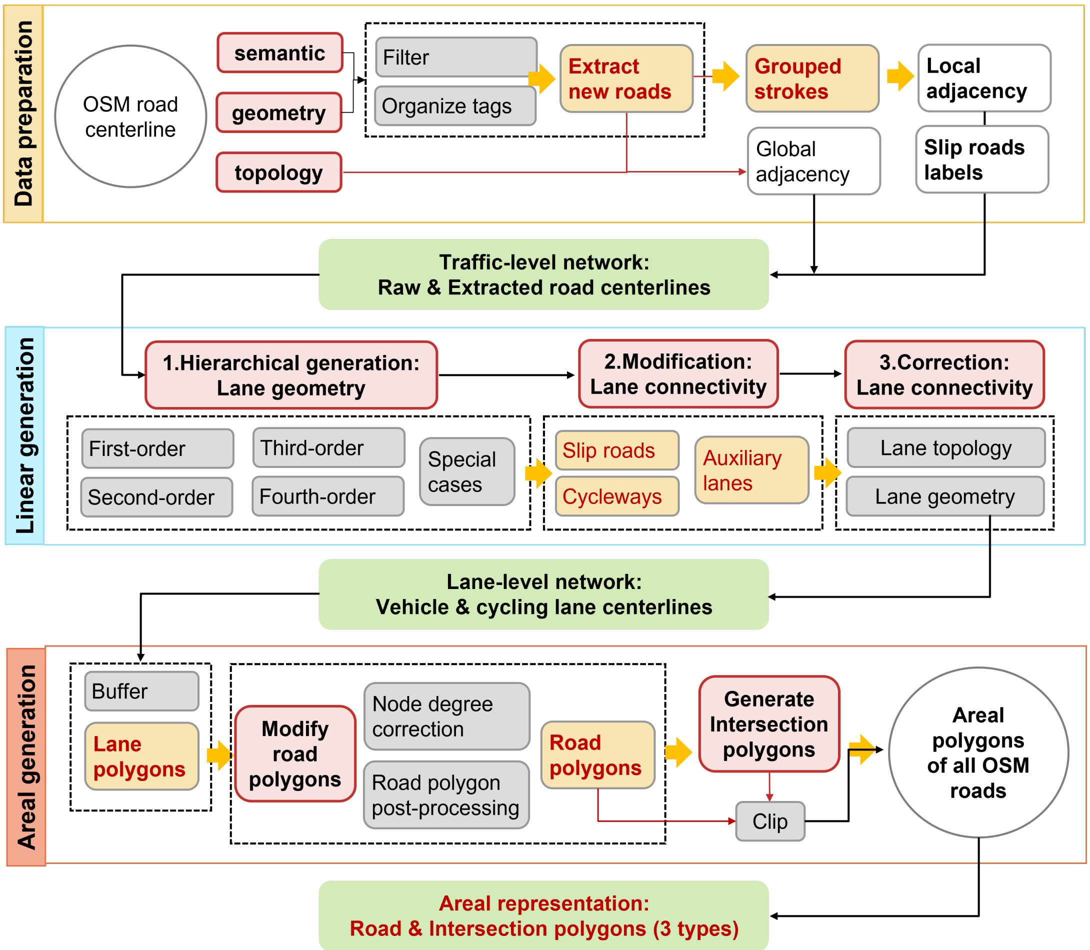


Figure 4.2.: Detailed workflow diagram

1. **Data Preparation:** Focuses on cleaning and structuring OSM data to form a road-level network with accurate attributes.

2. **2D Linear Generation:** Builds lane-level networks, emphasizing precise lane connections.

3. **2D Areal Generation:** Transforms linear representations into detailed areal models, capturing the full scope of road geometries.

4.2. Model Design and Method Design

According to the previous background knowledge and analysis conclusion, the modelling decision and design are regarded as the start of methodology, and also the guide for the further processing structure.

4.2.1. Model Design

This step establishes the modelling framework that guides the entire methodology, focusing on the relationship between road-level and lane-level networks. It ensures that each level—from road to lane to areal representation—is accurately connected, preserving the geometry, topology, and semantics required for high-resolution 2D areal models. The model data structure, relations and transformation between different levels, as well as the key elements are shown as Figure 4.3.

Relationships between road and lane networks To achieve a high-resolution 2D areal representation of the road model, the improved road geometry should originate from the lane level, capturing detailed geometric information for each lane as well as the topological relationships between roads and lanes. The road-level network provides a simplified overview, capturing intersections and road connectivity, while the lane-level network offers a more detailed view that accurately represents individual lanes and their geometric alignment. Together, these levels ensure that the generated 2D areal model retains both structural integrity and detailed spatial accuracy. Consequently, it is crucial to extract both the lane-level data and their corresponding topological relationships from the raw data, and clearly define how the road and lane networks interrelate.

Road-level network: The road-level network is modelled as a weighted directed graph (**DiGraph**), denoted as $\boldsymbol{G_r}$, where nodes represent traffic intersections and edges represent road segments with directional information. The traffic attributes are stored as edge properties, and the geometric correspondence characteristic describes the connectivity between roads. This simplified representation mirrors real-world conditions: if multiple edges converge at a single node, it indicates that the traffic areas of the corresponding roads meet at the same physical location.

Lane-level network: The lane-level network is modeled as a **MultiDiGraph**, denoted as $\boldsymbol{G_l}$. Here, edges represent lane segments with directional information, while nodes capture the connectivity between adjacent lanes. We introduce the key concept for evaluating the completeness and correctness of the lane-level network topology, **lane connectivity**, which directly influences the geometrical alignment of lanes (**lane alignment**). **Lane connectivity** is assessed based on two criteria:

- Consistency of Connectivity: The connectivity between roads and lanes must align topologically. If two road segments are connected at the road level, at least one pair of lanes from the corresponding roads must also exhibit a one-to-one connection. A failure to maintain this lane-level correspondence suggests that the lane topology does not accurately reflect the original road network.

- Integrity of Connectivity: Integrity spans across topology, geometry, and semantics. Accurate geometrical details of lane centerlines are essential for correct topological connections between lanes and roads. Additionally, lane edges in $\boldsymbol{G_l}$ must inherit and enrich attributes from the road-level network to ensure the model supports high LoDs in road reconstruction.

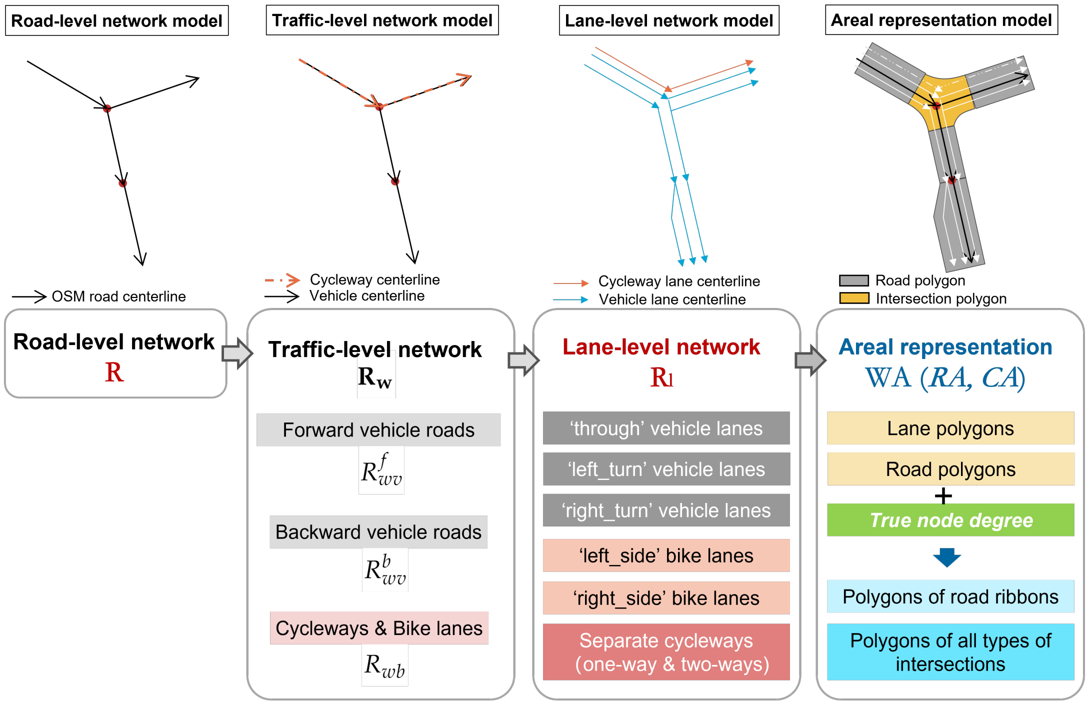


Figure 4.3.: Structure and transformation of three levels of network, areal representation, from the original OSM data to outcome as polygons.

Road-level network model sketch Without loss of generality, we consider the road network $\boldsymbol{W}$ as a set of roads $\boldsymbol{R}$ and a set of intersection $\boldsymbol{C}$, which is the mathematical abstraction of a physical road, OSM raw road centerlines. The road network is expressed as follows:

$$\begin{aligned} &\mathbf{W} = (\mathbf{R}, \mathbf{C}) \\ &\mathbf{R} = \{\mathbf{r}_i\}_{i=1}^{n} \qquad \mathbf{C} = \{\mathbf{c}_j\}_{j=1}^{m} \end{aligned} \tag{4.1}$$

where n and m are the numbers of the roads and intersections, respectively; r_i is the ith road in the set of road of the whole network, c_j is the jth intersection in the set of intersection of the whole network.

To represent the road in the road-level model, we define it as follows:

$$\begin{aligned} &\mathbf{r} = (I_r, V_r, VS_r, VE_r, L_n, L_r, L_f, Q_r) \\ &\mathbf{Q_r} = (M_r, H_r, T_r, D_r) \end{aligned} \tag{4.2}$$

The road $\boldsymbol{r}$ is the mathematical abstraction of a physical road (technically, it's a single road centerline that is encoded in OSM as a road) consisting of one or more traffic lanes and semantic attributes. I_r is the unique identifier for each road, which can either be traversed from the OSM raw data or defined internally. V_r, VS_r, VE_r are the set of vertices, the start vertex and end vertex of the road centerline, respectively, each vertex in V_r contains the latitude(x) and longitude(y) coordinates as its physical location, the order of the vertices follows the travelling direction and curvature of the road. L_n is the total number of lanes within a road; L_r is the set of traffic lanes in the road. L_f is the set that describes the offset distances of traffic lanes relative to the road centerline geometry. This is important for reconstructing the spatial configuration of lanes around the centerline.

$\boldsymbol{Q}_r$ is the attribute collection that can be extracted from OSM raw data (i.e., OSM *tags*)for each road to represent the semantic information:

- M_r is the movement direction of traffic flow in the road, since one single OSM road centerline can represent a one-way street, and also a two-way street; refer to the *'oneway'* tag;
- H_r is the traffic type of the road, refer to the *'highway'* tag;
- T_r is the turning attribute of the road, implying the changing direction of road or lanes, refer to the series of *'turn'* tags;
- D_r is the indicator of road's junction type or 3D relation, implying the interchange or 3D intersection transportation space, refer to the *'junction'*, *'bridge'*, *'tunnel'* tags.

The intersection $\boldsymbol{c}$ uses the geometric and topological information of a physical intersection to represent in the road network model, which we defined as follows:

$$\mathbf{c} = (O_c, N_k, J_r, JS_r, JE_r, A_c)$$

$$\mathbf{A}_c = \begin{array}{c} \\ p_1 \\ p_2 \\ \vdots \\ p_n \end{array} \begin{array}{c} \begin{array}{cccc} q_1 & q_2 & \cdots & q_n \end{array} \\ \begin{bmatrix} 1 & 1 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \end{array} \tag{4.3}$$

where O_c represents the geographic location of the intersection, which coincides with the coordinate of the node representing the intersection. In a 2D linear representation, O_c serves as the shared point where roads meet. For each intersection, N_k represents the number of "*intersection legs*" or connected roads at an intersection, a crucial feature from graph theory. In this model, N_k simplifies the concept by not distinguishing between entering or exiting traffic as a directed graph needed and instead represents the total node degree. J_r, JS_r, JE_r are the representation of the *intersection legs*, they are the set of all the joint roads, entering roads and the exiting roads of the intersection, respectively.

To describe the topology connection relationship of roads of each intersection, we can build an adjacent matrix $\boldsymbol{A}_c$ to indicate the connectivity of road segment geometry. For example, A_{ij} equals 1, which means p_i and q_j are connected, the road segments share the same endpoint, otherwise disconnected.

Traffic-level network model sketch The traffic-level network serves as a mid-level model that represents and differentiates between various types of traffic flows within road ribbons. Its purpose is to bridge the broader road-level network and the detailed lane-level network by adding additional road centerlines based on raw data ($\boldsymbol{R}$), distinguishing between different traffic modes. The key traffic modes reflected in this model are twofold: 1) Road direction (e.g., *forward* and *backward*), and 2) Driving types (e.g., *motorized driving* and *cycling*).

This mid-level model is denoted as $\boldsymbol{R}_w$, represented as a MultiDiGraph to resolve discrepancies between the direction of travel and vector direction (as mentioned in Section 2.3.2). The distinction between motorized driving and cycling is evident in both the topological connections—e.g., between vehicle-vehicle lanes, cycleway-cycleway, or cycleway-bike lanes—and the spatial distribution of the transportation space allocated to each traffic type.

In this model, we only highlight the representation of roads, $\boldsymbol{R}_w$. Roads are represented by a single centerline geometry, similar to $\boldsymbol{R}$, but with enriched details to describe multi-lane conditions and traffic modes:

$$
\begin{aligned}
\mathbf{R_w} &= \{\mathbf{r}_i\}_{i=1}^{j} \\
\mathbf{R_w} &= (R_{wv}^{f}, R_{wv}^{b}, R_{wb})
\end{aligned}
\tag{4.4}
$$

where j is the new number of roads after extraction, and it may differ with the numbers of the roads n in $\boldsymbol{R}$. $\boldsymbol{R}_w$ can also be represented as the set of *forward* direction motorized roads $\boldsymbol{R}_{wv}^{f}$, *backward* direction motorized roads $\boldsymbol{R}_{wv}^{b}$, and the set of all the cycling roads $\boldsymbol{R}_{wb}$, each subset maintains the topology correctness to facilitate the lane generation.

To emphasize the specific characteristics of roads in the mid-level model, the road element $\boldsymbol{r}_w$ is tailored based on the road-level model ($\boldsymbol{r}$) and redefined as follows:

$$
\begin{aligned}
\mathbf{r_w} &= (I_r, I_w, V_w, VS_w, VE_w, L_{nw}, L_w, Q_w) \\
\mathbf{Q_w} &= (M_w, H_w, T_w, D_w)
\end{aligned}
\tag{4.5}
$$

I_r as the raw road index, retained to maintain the connection between $\boldsymbol{r}$ and the mid-level representation ($\boldsymbol{r}_w$). A new index I_w is for identifying the new bike lanes and new '*backward*' roads. V_w, VS_w, VE_w are the set of vertices, start vertex, and end vertex of the new road centerline, respectively, ordered by travel direction. L_{nw} is the total number of lanes for a given traffic flow, indicating lane counts in one direction; and L_w is the set of lanes associated with the traffic flow. T_w depicts the tuning attributes of each vehicle lane in L_w; for the extracted bike lanes, T_w indicates whether they are positioned to the left, right, or on both sides of vehicle lanes. M_w and H_w may may differ from the original attributes in $\boldsymbol{R}$ according to the extracted mid-level road types.

The traffic-level model extracts new road types and differentiates them by direction and mode (e.g., cycling versus motorized driving). Roads such as one-way streets or dedicated cycleways retain their original attributes after the division of $\boldsymbol{R}$ into subsets, ensuring that the mid-level roads in $\boldsymbol{R}_w$ are ready for further lane generation and detailed lane-level modelling.

This intermediate network thus serves as the foundation for generating high-fidelity lane geometry and for linking road-level and lane-level networks, ensuring that both directional and traffic-type distinctions are incorporated into the final network structure.

Lane-level network model sketch The lane-level road network is designed to provide enhanced geometric and topological detail compared to the road-level network, capturing individual lane attributes and relationships, denoted as $\boldsymbol{R}_l$. This finer-grained network is essential for modeling high-resolution traffic and spatial dynamics. Since the lane information in OSM raw data is embedded within various tags, reconstructing the lane-level network requires extracting and organizing this hidden information while preserving the road-lane correspondence. Based on the OSM data characteristics and the model design requirements, we define the lane-level network for two main goals:

1. To represent the lane geometry, lane topology and semantic information, which facilitates the reconstruction of physical roads and intersections at the lane-level;

2. To complete the relations between the road and its lanes, the relations include 1) the geometric position between the raw road centerline and the derived lane(s), 2) topological connections of adjacent lanes and adjacent roads, 3) as well as the semantic information of each lane and their parent road.

Lane topology includes two parts: connectivity at turns and among parallel lanes Zhu and Li [2008]. Whether using the road-level model to reconstruct or the lane-level model, the final results in geometry representation should contain the road geometry and intersection geometry. Differentiate from the road-level and traffic-level network model (Equation 4.2, 4.5), we define one road in the lane-level model as one lane or multiple lanes; the geometry, topology and attributes of the lane(s) are mostly derived and inherited from its parent road $\boldsymbol{r}_w$. Similarly, the set of joint roads of an intersection $\boldsymbol{c}$ can be replaced as a set of joint lanes in the lane-level model.

$$\begin{aligned}
&\mathbf{r_l} = \mathbf{L_r} = \{\mathbf{l}_i\}_{i=1}^{a} \\
&\mathbf{l} = (I_r, I_w, V_l, VS_l, VE_l, Q_l) \\
&\mathbf{Q_l} = (M_l, H_l, T_l, D_l, W_l, FD_l)
\end{aligned} \tag{4.6}$$

The $\boldsymbol{r}_l$ is the set of its lanes. To describe the details of a lane, we define the lane keeps the raw index of its prior road I_r as the road-lane mapping reference, and I_w inherits from the traffic-level road centerline to distinguish the lanes which have different attributes with the raw road r_i.

In the attributes of a lane, $\boldsymbol{Q}_l$ inherits $\boldsymbol{r}_w$ attributes. Need to notice that, T_r and T_w indicate turning conditions of all lanes as a list, but in lane-level, T_l only represents the exact turning label of the lane. Two added attributes W_l and FD_l depict the width of the lane and the relative offset distance to the road centerline, respectively. Similar to the road geometry modelling method, V_l, VS_l, and VE_l represent the set of vertices, the start vertex and end vertex of each lane centerline, respectively; those values will differ with the road centerline geometry as they show the deviation and relative location of lanes.

When we define the intersection $\boldsymbol{c}_l$ in lane-level model, replacing the J_r, JS_r and JE_r as J_l, JS_l and JE_l; N_l represents the total number of lanes joint in the intersection; the adjacent matrix $\boldsymbol{AL}_c$ of lane-level topology connection is more complex than the $\boldsymbol{A}_c$ since lane connectivity in an intersection encompasses the travel direction of traffic flows and serve as a direct graph.

$$\begin{aligned}
&\mathbf{c_l} = (O_c, N_l, J_l, JS_l, JE_l, AL_c) \\
&\mathbf{J_l} = (r_1, r_2, ..., r_n) \\
&\quad = (\{\mathbf{l}_i\}_{i=1}^{a}, \{\mathbf{l}_j\}_{j=1}^{b}, ..., \{\mathbf{l}_q\}_{q=1}^{c}) \\
&\mathbf{AL}_c = \begin{array}{c} \\ p_1^1 \\ p_2^1 \\ p_1^2 \\ p_2^2 \\ \vdots \\ p_1^n \\ p_2^n \end{array}
\begin{array}{c} \begin{array}{ccccccc} q_1^1 & q_2^1 & q_1^2 & q_2^2 & \cdots & q_1^n & q_2^n \end{array} \\
\begin{bmatrix}
1 & 1 & 1 & 1 & \cdots & 0 & 0 \\
1 & 1 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 1 & 1 & \cdots & 0 & 0 \\
0 & 1 & 1 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 1 & 1 & \cdots & 1 & 0 \\
0 & 0 & 1 & 1 & \cdots & 1 & 0
\end{bmatrix} \end{array}
\end{aligned} \tag{4.7}$$

Areal representation model sketch Areal representation is derived from the linear geometry of the lane-level network model, in road level are the road polygons aggregated by the lane polygons. This involves generating lane polygons and dissolving them based on road indices I_r, which reflect the affiliation between roads and their respective lanes. Based on the road network model sketches, we can design an areal representation model where all elements are represented by 2D polygons.

This model, denoted as $\boldsymbol{WA}$, represents the linear road network $\boldsymbol{W}$ through a set of road polygons $\boldsymbol{RA}$ and intersection polygons $\boldsymbol{CA}$. The areal model allows for more detailed visualization of the road network by converting linear elements into complete 2D polygons, capturing the geometric and semantic attributes of the road and intersection areas:

$$\begin{aligned}&\mathbf{WA} = (\mathbf{RA}, \mathbf{CA})\\&\mathbf{RA} = \{\mathbf{r_A}_i\}_{i=1}^{n} \qquad \mathbf{C} = \{\mathbf{c_A}_j\}_{j=1}^{m}\end{aligned} \tag{4.8}$$

where n and m are the numbers of the road polygons and intersection polygons, respectively.

The areal model of the road network is defined as follows, using attributes inherited from the linear road network:

$$\begin{aligned}&\mathbf{r_A} = (I_r, P_r, L_n, L_P, W_r, HA_r, D_r)\\&\mathbf{c_A} = (I_c, O_c, N_c, P_c, R_c, R_P, H_c)\end{aligned} \tag{4.9}$$

For a road polygon, I_r (road index) ensures that each road polygon corresponds to its original road in the linear network; P_r (road polygon) Represents the full geometry of the road surface, accurately capturing the road's shape; L_n and L_P specify the number of lanes and their corresponding polygon geometries; W_r is the total width of road surface, calculate by the L_n and the corresponding FD_l; semantic information for distinguishing road types is still needed, HA_r derived from the original road type (H_r), this semantic information classifies the road as either 'vehicle', 'cycleway', or 'mixed', based on the lanes within the road polygon and their respective traffic modes; D_r is the inherited attributes from OSM road data for marking the 3D relationships of roads.

For an intersection, some attributes from node are still vital, the O_c and N_c represent the original node position and the node degree of this intersection, respectively; P_c is the polygon for describing the compact boundary of one intersection, representing the area where different roads meet and turning movements occur; R_c and R_P store the joint road information as a set of roads indices and the corresponding polygons P_r; additionally, semantic information used to classify intersections based on the traffic modes they serve, denoted as H_c.

4.2.2. Method Design

Primary aspects of reconstruction methods The methodology employed in this study is structured around three primary aspects: topology-driven, geometry-driven, and semantic-driven methods. Combining these methodological approaches, which prioritize different fundamental considerations, optimizes the final results by leveraging the advantages of their juxtaposition and affiliation.

Topology-driven method By prioritizing topological connection, this method highlights building the correct relationships between the road network and lane network, maintaining the consistency and integrity of connectivity from the road network to the derived lane network. The method ensures a robust understanding of the structural road and lane network, as well as interdependency among the components.

Geometry-driven method Prioritized to achieve accurate measurements and spatial configurations. Geometry-driven methods ensure the the completeness of geometric correspondence, and precise representation of shapes, sizes, and physical characteristics, enhancing the accuracy of the reconstructed models.

Semantic-driven method Integrated to assign meaningful labels and interpretations to the reconstructed model. Semantic-driven methods enhance the context and significance of the reconstructed models, improving their interpretability and practical application by representing the various lane types and multi-functional road areas.

4.3. OSM Road Network Initialization and Stroke Generation

Introduction The purpose of this step is to prepare the OSM road data for generating a lane-level representation that is both robust and accurate. The goal is to transform raw, unstructured data into a standardized and structured format that supports high-resolution lane models. To achieve this, the following sub-sections are implemented, each serving a specific function:

- **Data Preparation and Filtering** (4.3.1): This step focuses on filtering and extracting the relevant road data from OSM. The data preparation is crucial because OSM contains a wide range of tags, not all of which are useful for lane-level modelling. This step ensures that the input data aligns with the requirements for further lane-level modelling, excluding unrelated elements like railways and bus stops.
- **OSM Data Pre-processing** (4.3.2): After data extraction, this step aims to correct and enrich the attributes of the road-level graph to ensure topological completeness. This is necessary because raw OSM data often lacks details like lane counts and lane-changing rules. By refining these attributes, this step lays the foundation for building a complete traffic flow representation that will later be used to model lane geometries accurately.
- **Road Adjacency Identification** (4.3.3): This step identifies how road segments relate to each other in terms of connectivity, which is essential for defining turning relationships and maintaining topological consistency across the road network. Calculating interior angles between adjacent segments helps ensure that connected roads are identified correctly, facilitating accurate lane alignment and intersection modelling.
- **Grouped Strokes Generation** (4.3.4): This process involves generating grouped strokes for better continuity of road and lane centerlines. Unlike standard stroke generation methods that focus on broader road network analysis, this tailored approach emphasizes local continuity between lanes, ensuring that lane transitions and intersections are handled correctly. It aims to create sub-strokes that preserve the continuity needed for accurate lane-level reconstruction.

- **Updating Road Attributes** (4.3.5): The final sub-step updates the attributes of the road segments with the refined details obtained from previous steps, such as adjacency relationships and grouped strokes. This update ensures that the road network is ready for subsequent lane-level and areal representation phases, supporting high-LoDs required for accurate modelling.

4.3.1. Data preparation and filtering

In the start phase of data pre-processing, the targeted elements as the input data for the study should be clarified to exclude unrelated elements from numerous transportation data in OSM. As described in Section 2.3, roads are just a subset of OSM data elements, road segments are represented as linear elements among others. Thus, we need to set the *'highway'* feature of **way** element as the target. In this way, the other components of transportation such as railway and infrastructure, area of bus stops and squares would be excluded first.

To extract only the roads we aim to reconstruct, the motorized roads and cycleways, we filter the original data so that we keep only those lines that have a specific value of the *highway*=* tag indicating that it is some form of a motorway or cycleway. According to the filtering goal, the values that indicate a walking route, an auxiliary service route (parking lots paths, building internal paths), or roads under construction are filtered, such as "pedestrian", "path", "footway", "step", "service", "escape", "construction", proposed". The following values of *highway*=* tag were kept:

- primary
- secondary
- tertiary
- motorway
- trunk
- primary_link
- secondary_link
- tertiary_link
- motorway_link
- trunk_link
- residential
- living_street
- unclassified
- cycleway

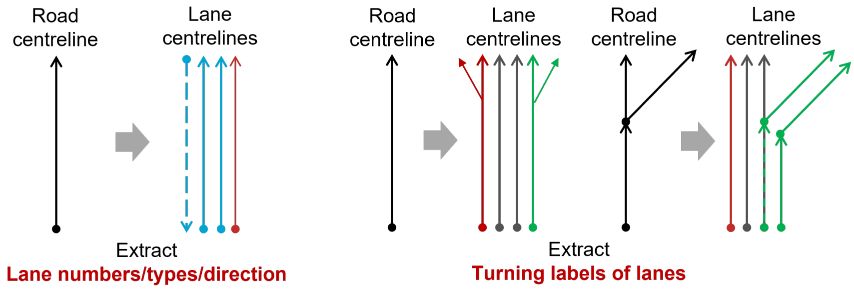


Figure 4.4.: Schema of extracting lane information from OSM raw data, then applying them for lane generation.

4.3.2. OSM data pre-processing

The OSM raw data can be regarded as $\boldsymbol{G}_r$, a *digraph*, the set of edges are the set of raw road centerlines of the road-level network. This phase enhances the attributes of the raw road data to support accurate modelling. Key actions consist of:

- Defining directed edges: Aim to extract all the directed edges that represent the complete traffic flow movements in $\boldsymbol{W}$, not only the vehicle driving routes but also cycling routes, to represent comprehensive traffic flows.
- Attribute enhancement: Augment the dataset with those necessitated to facilitate detailed traffic flow representation.

As the road-level model sketch described in Equation 4.1, 4.2, in the initial $\boldsymbol{R}$, number of roads n is the number of raw road centerlines after filtering; geometric related attributes V_r, VS_r, and VE_r already exist for each raw road centerline; and the attributes that described more detailed information such as number of lanes, and lane changing (Figure 4.4) are not sufficient for the data model, therefore we need to complete the attributes in this step.

Thus, the OSM raw data pre-processing phase encompasses steps as follows:

1. **Extract hidden traffic flows:**

   a) **Extract backward carriageways.** Some bi-directional vectors (i.e. road centerlines) are two-way streets as we analyzed in Section 2.3.2, since one of the principles of the reconstruction method is to prioritize the topology of $\boldsymbol{G}_r$, the *backward* traffic flows are needed for $\boldsymbol{W}$ and $\boldsymbol{R}$. The method is checking the *oneway=** tag, the road in which the value of this key is not *"yes"* can be identified as a two-way street. After the extraction, a new road centerline r_i^{back} will be added to $\boldsymbol{R}$ and $\boldsymbol{R}_w$. As the *backward* direction vector, r_i^{back} has the same vertices in V_r^{back} as V_r but in the opposite order. VS_r^{back} is the original VE_r, and VSE_r^{back} is the original VS_r. The *backward* roads will be added to $\boldsymbol{R}_w$ as a new road $\boldsymbol{r}_w$.

   b) **Extract the bike lanes.** To reconstruct cycling routes and cycling space, only using the *highway=cycleway* elements is not sufficient. Based on the introduction of *cycleway=** tag in Section 2.3.1, we know that completing the cycling area located on the main roadway as bike lanes is necessary to recover the topology connection of cycling traffic, and also can be regarded as an important step for lane centerline extraction and semantic enrichment. The extraction method is to traverse all roads *cycleway=** tag, when values are *cycleway=lane*, *cycleway:left=lane*, *cycleway:right=lane* or *cycleway:both=lane*, that means bike lane(s) should be extracted and added to $\boldsymbol{R}_w$. Reverse the road geometry if the extracted bike lane has the opposite direction of the raw road centerline.

2. **Complete and correct the attributes:** After the extraction, $\boldsymbol{R}_w$ is built, the attributes correction is for each road $\boldsymbol{r}_w$.

   a) **Number of lanes.** Not each road has value in *lanes=** tag, but it is the main factor in reconstructing the lane centerlines, lane polygons and road polygons. Number of lanes L_{nw} can be derived from two tags: 1) *'lanes'* tags; 2) convert the value of *'turn:lanes'* to the number of lanes, the number equals the amount of "|" (vertical bar) plus one. For one-way streets, i.e. the *oneway=yes* motorized roads, directly use the value of this key as L_{nw}, or use the conversion number of *'turn:lanes'*. For two-way streets, suffixes *:forward* and *:backward* are needed, for the road in the subset $\boldsymbol{R}^f_{wv}$, using *'lanes:forward'* or *'turn:lanes:forward'* as the keys; and vice versa. The sum of L^f_{nw} and L^b_{nw} is the total number of lanes L_n of $\boldsymbol{r}$.

   b) **Turing lanes.** The values of *'turn:lanes'*, *'turn:lanes:forward'* and *'turn:lanes:backward'* correspond to the lane markings for drivers in the real world. Besides the auxiliary lanes and ramps which have labelled values in the raw data, this step is used for all the empty value lanes, the "turning" attributes of them are "*through*". According to the suffixes *:forward* and *:backward* to assign the correct turning set to the corresponding $\boldsymbol{r}_w$. The final set of turning values T_r for each road should align with the number of lanes L_n. In other words, each motorized lane should have its turning value.

   c) **Check the road types.** Specicially, H_w of the extracted bike lane should use *highway=cycleway* rather than the H_r. And M_w of all the motorized road in $\boldsymbol{R}_w$ is *'oneway=yes'*.

4.3.3. Road adjacency identification

As a graph-like data, the road network has its unique topology structure that can mostly revealed by the adjacency of each road (edge). This step identifies the topology connection relation for each edge, essential for accurate lane alignment and intersection modelling. Key actions involve:

- Calculating interior angles: Determine the angles between adjacent segments to verify their connectivity.
- Creating road adjacency connections: Use calculated angles to accurately connect road segments, ensuring topological integrity throughout the entire study on three different levels.

4.3.3.1. Interior angle calculation and comparison The method to identify adjacency for each edge (i.e. road or lane) could be generic and can apply to different data or various scenarios. However, unlike the general method used for a graph or a digraph, apart from the direction, the interior angle between two road neighbours is another vital factor that needs to be considered according to the relation of turning path and interior segments has been introduced in Section 2.4.3 (Figure 4.5). To detect the real adjacent edges, after traversing the neighbour edges based on the direction (start vertex and end vertex of each road/lane centerline), we need to calculate the bearing angle of the start segment, denote as θ^s, and the bearing angle of the end segment, denote as θ^e, for the road centerline. The interior angle between two connected edges e_1 and e_2 can be converted by the difference between θ^e of e_1 and θ^s of e_2. e_1 is the predecessor of e_2.

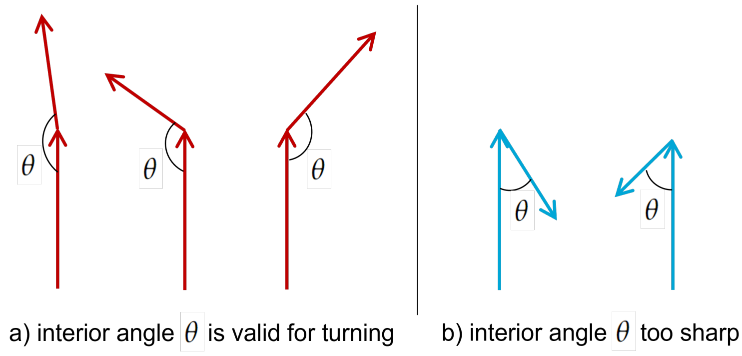


Figure 4.5.: Interior angle for adjacent conditions detections

1. **Bearing angles of each road.** First, the calculation of bearing angles employs the coordinates of segment vertices, where x_1, x_3 are the start vertices' x-coordinates of e_1 and e_2, x_2, x_4 are the end vertices' x-coordinates of e_1 and e_2, and so on. Secondly, considering the changeable direction of roads, we further convert the bearing angle to a new value. The standard arctangent result (angle θ^s, θ^e) relative to the positive x-axis and measured counterclockwise; however, in this method, a new bearing angle θ_t for each arctangent result is instead measured relative to the positive y-axis, with the angle increasing in the clockwise direction. The transformation value can give the exact interior angles between two segments, this method of determining the interior angles is inspired by ***COINS***'s algorithm Tripathy et al. [2020]. The complete calculation is adjusted as follows, replacing the original θ^s, θ^e by its corresponding θ_t:

$$\theta^s = \arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \times \frac{180^\circ}{\pi} \quad \theta^e = \arctan\left(\frac{y_4 - y_3}{x_4 - x_3}\right) \times \frac{180^\circ}{\pi} \tag{4.10}$$

$$\theta_t = \begin{cases} 90^\circ - \theta, & \text{if } \theta \leq 90^\circ, \\ 360^\circ - (\theta - 90^\circ), & \text{if } \theta > 90^\circ. \end{cases} \tag{4.11}$$

After the calculation, we update the starting bearing and end bearing angles as new attributes to the road $\boldsymbol{r_w}$ as:

$$\mathbf{r_w} = (I_r, I_w, V_w, VS_w, VE_w, \theta^s, \theta^e, L_{nw}, L_w, Q_w) \tag{4.12}$$

2. **Valid interior angle to determine adjacency.** To determine whether a turning path is real, compare the difference between the bearing angles of consecutive segments,

e_1 and e_2. If any of these conditions are satisfied, the adjacency between roads is considered valid since it aligns with the turning requirement:

$$\text{Turn is } \mathit{True} \text{ if} = \begin{cases} 0^\circ < (\theta_2^s - \theta_1^e) < 90^\circ & \text{or} \\ 0^\circ < (360^\circ - \theta_1^e + \theta_2^s) < 90^\circ & \text{or} \\ 0^\circ < (\theta_1^e - \theta_2^s) < 90^\circ & \text{or} \\ 0^\circ < (360^\circ - \theta_2^s + \theta_1^e) < 90^\circ & \text{is true.} \end{cases} \tag{4.13}$$

where θ_2^s is the start bearing angle of the current segment e_2; θ_1^e is the end bearing angle of the previous segment e_1, thus, the a valid interior angle will determine e_1 as the predecessor of the current segment e_2.

4.3.3.2. Creating road adjacency connections To accurately establish adjacency connections between road centerlines, we use the traffic-level data $\boldsymbol{R_w}$ rather than the original road-level $\boldsymbol{R}$. Conducting the following algorithm which details the process for determining these relationships, ensuring that only segments with valid turning angles, correct geometric alignment and the same traffic mode are considered as successors or predecessors. The following algorithm 4.1 is for motorized roads by adding the constraints as one of the conditions.

Algorithm 4.1: ROAD ADJACENCY ($\boldsymbol{R_w}$)

Input: A road network R_w, a road index I_w, the start and end vertex of road centerline r geometry VS_w and VE_w, a road highway type H_w, and start and end bearing angles θ^s, θ^e, a equation Eq. (4.13)

Output: Pre_w and Suc_w: the set of predecessors and successors of each road r_w, store to R_w as new attributes

for $r_i \in R_w$ **do**
    for $r_j \in R_w$ **do**
        if $I_i \neq I_j$ *and* $H_i \neq$ *'cycleway' and* $H_j \neq$ *'cycleway'* **then**
            if $VE_i = VS_j$ **then**
                Inter_angle $\leftarrow$ *Eq.*(4.13)Valid_interior_angle(θ_i^e, θ_j^s);
                if *Inter_angle is* ***True*** **then**
                    Suc_i.append(r_j);
            if $VS_i = VE_j$ **then**
                Inter_angle $\leftarrow$ *Eq.*(4.13)Valid_interior_angle(θ_i^s, θ_j^e);
                if *Inter_angle is* ***True*** **then**
                    Pre_i.append(r_j);
return R_w

4.3.4. Grouped strokes generation

From 'Global strokes' to 'Grouped strokes' As we create the adjacency relations for each road segment in road and traffic level, this following process tailors the stroke generation method to enhance the adjacency relations by prioritizing the 'Good Continuity' characteristic in road networks based on the stroke generation as introduced in Section 2.4.1 and 3.2. To achieve the goals of reconstructing the high-LoDs model with **lane connectivity**, we adjust the general stroke generation method from *global* 'Good Road Continuity' strokes to *local* 'Good Lane Continuity' sub-strokes, so-called *grouped strokes*. Key actions include:

- Creating grouped strokes: Generate localized strokes that prioritize lane connectivity over broader network continuity.
- Adjusting strokes of slip roads for continuity: Refine strokes to maintain continuity between lanes, critical for accurate lane reconstruction in lane level network.

Combined with the definition and extraction of traffic-level network $\boldsymbol{R}_w$, denote the set of strokes $\boldsymbol{S}$, and a subset of *grouped stroke* as $\boldsymbol{s}$, the relation of $\boldsymbol{R}_w$, $\boldsymbol{S}$ and road $\boldsymbol{r}_w$ can be described as follows:

$$\begin{aligned} \mathbf{S} &= \mathbf{R_w} = \{\mathbf{s}_i\}_{i=1}^{q} \\ \mathbf{s} &= \{\mathbf{r}_w\}_{i=1}^{p} \end{aligned} \tag{4.14}$$

where q is the total number of grouped strokes of the road network $\boldsymbol{W}$, each $\boldsymbol{r}_w$ in $\boldsymbol{R}_w$ is an element of one grouped stroke $\boldsymbol{s}_i$; p is the number of roads in each grouped strokes.

4.3.4.1. Generation method The method of generation contains two phases: initial grouping and adjustment.

1. **Initial grouping:** Inspired by the global strokes generation method in Section 3.2, the grouped strokes generation chooses two decisive factors as well: 1) Road type, tagged in *highway*=* key; 2) Road name, tagged in *name*=* key. The first-factor, road type, contributes to distinguishing the hierarchical traffic modes. For example, the level of traffic degrades from '*primary*' to '*secondary*'; or differentiates cycling traffic *highway=cycleway* with other motorized types. Road name is semantic information, that helps to divide strokes further based on the intersections (road name may change when encountering a major intersection, this change is commonly seen in urban scenarios).

2. **Adjustment:** The initial groups show the efficiency of the chosen factors. However, according to the functions of grouped strokes, all the roads $\boldsymbol{r}_w$ that belong to the same group as a subset $\boldsymbol{s}_q$ must have the same directional traffic mode. In other words, when a driver or a cyclist moves from the start node of the grouped stroke, it can arrive at the end node(s) of this grouped stroke (subset $\boldsymbol{s}_q$ can be regarded as a continuous line consists of all $\boldsymbol{r}_w$ centerlines). Therefore, the slip roads or ramps as we found in Figure 5.2 should be reassigned to their correct traffic flow and labelled as a different group number. After the adjustment, the adjacency connections of each road $\boldsymbol{r}_w$, i.e. their correct *local* predecessors and successors can be identified within their grouped stroke $\boldsymbol{s}$. Detailed adjustment implementation is described in Section 5.3.

4.3.4.2. Slip roads labelling In the phase of grouped strokes adjustment, the main effort is to fix the wrongly assigned group values of the slip roads in road bifurcations. This step is beneficial to facilitate the recovery of the turning mechanism within the road network.

Based on the understanding of slip roads and ramps, the prerequisite of each diverging movement must be the correct connection: the road has a predecessor which allows or guides the driver to start turning; the predecessor could be an auxiliary lane, or a lane with the marking specified as a turn lane. For the merging conditions, the prerequisites of identifying are not as necessary as diverging one; the direction of the merging road (i.e. on-ramp) and its successor should be consistent to guarantee the topology correctness as the edges of a directed graph.

Since the connection requirements of the predecessors are more strict than the successors, we prioritize the labelling of off-ramp roads/forks, they can be further split into three types: left, right and through forks. A typical case of left, through and right fork road labelling is illustrated in Figure 4.6. Similar to the interior angle calculation in Section 4.3.3, a meticulous interior angle calculation is needed to label them based on the turning direction. Besides the geometric correspondence, the suffixes of *highway=** values also help with the forks labelling: the suffix as *'_link'* indicates a ramp road, for instance, *'highway=primary_link'*.

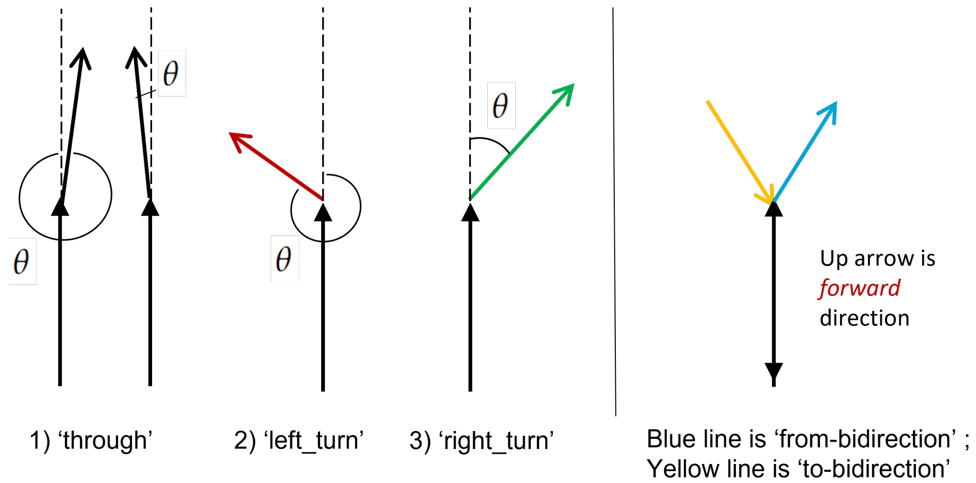


Figure 4.6.: Fork labelling based on the interior angles.

Additionally, on the road configuration, some bi-directional roads have road bifurcations at the endpoint of the segments. In some cases, there is no *'through'* road/lane in those road bifurcations, thus the forks do not link to the turning movements; only represent that two one-way roads with different directions join to the same two-way road. As the description of the traffic-level network $\boldsymbol{R}_w$, we know that both *forward* and *backward* road segments exist; the fork-like roads we aim to specify can be assigned as 1) *'from_bi-direction'*, which is the successor of the *backward* road segment; and 2) *'to_bi-direction'*, which is the predecessor of the *forward* road segment. The illustration of those conditions is shown in Figure 4.6.

The detailed labelling methods are shown in Section 5.3.

4.3.5. Update road attributes

After finishing road adjacency identification, grouped stroke generation and fork road labelling, we store the results of each sub-step as new elements in $\boldsymbol{r}_w$ and semantic attributes in Q_w. The updated road centerline $\boldsymbol{r}_w$ comprise the inherited and derived attributes as shown in Eq. (4.15). The traffic-level road network $\boldsymbol{R}_w$ is fully generated and will be used in further processing to improve the road network to lane-level.

$$\begin{aligned} \mathbf{r_w} &= (I_r, I_w, V_w, VS_w, VE_w, \theta^s, \theta^e, Pre_w, Suc_w, L_{nw}, L_w, Q_w) \\ \mathbf{Q_w} &= (M_w, H_w, T_w, D_w, GS_w, F_w) \end{aligned} \tag{4.15}$$

where elements Pre_w and Suc_w represent the set of the predecessor(s) and successor(s) of $\boldsymbol{r}_w$; GS_w and F_w are the updated attributes, representing the group index (integer) of strokes and labelled fork types (string), respectively. Each $\boldsymbol{r}_w$ has its GS_w, while only the fork roads have valid F_w values to distinguish them from the normal road segments.

4.4. Lane Network Reconstruction

Introduction This section outlines the process for generating a lane-level network, building upon the previously established road-level and traffic-level networks. The core objective is to derive lane geometries from the central road centerline for each road partition. This process is driven by three key considerations:

1. **Semantic Considerations:** The distinction between traffic modes, previously established in the traffic-level network, will be extended to the lane-level network. This affects the validation of connectivity and adjacency across traffic modes, as well as lane topology and geometry. Particular attention is given to ensuring that lane adjustments maintain the correctness of both topology and geometry.
2. **Topological Considerations:** The lane-level network's topology, referred to as **lane-connectivity** in Section 4.2.1, represents continuous traffic flows and accounts for changes in complex scenarios. Ensuring correct and logical connections between lanes is crucial for accurately modeling traffic behavior.
3. **Geometric Considerations:** Although the road network is represented linearly, correct geometric correspondence remains crucial, mirroring the topological relationships. In this linear representation, all nodes in the Digraph/MultiDiGraph are the start or end vertices of the centerlines. The focus here is on:
    a) Emphasizing the unidirectional nature of road centerlines, which reflects the maneuvering behavior of traffic as it enters or exits the roadway.
    b) Recognizing that while the road centerline may not always represent the true center of a road polygon, the derived lane centerlines can be regarded as the true **centers of lane polygons**. These lane centerlines will directly inform the generation of lane polygons.

To integrate these considerations, the following procedural methods are employed:

- **Hierarchical Reconstruction** (4.4.1):

- *Goal & Why*: The primary aim is to create a lane-level network with "Good Lane Continuity," ensuring a smooth and consistent representation of lane flows. Traditional approaches treat the road centerline as the center axis of the road polygon boundary, which may not accurately represent the lane positions in the real world. Our method redefines this by using the road centerline as the '***reference line***, it can help to clarify the 'through' lanes location. With the hierarchy selection of all the road segments, it allows better identification of the accurate lane positions. The comparison of the traditional method and our new method is visualized in Figure 4.7. This adjustment is crucial for ensuring the functional alignment of lanes within complex road structures.

- *Method*: Based on principles from transportation design, emphasizing the "Good Continuity" of local strokes, as detailed in Section 2.4.2. This phase involves determining the positions of lane centerlines relative to the original road centerlines in a hierarchical and stepwise manner. we use two main factors—grouped strokes and the base number of lanes (as shown in Figures 4.8)—to guide the hierarchical placement of lane centerlines relative to the original road centerline. The approach is systematic, breaking down the process into manageable steps that ensure lane centerlines are positioned accurately while maintaining lane continuity.

- **Lane-level Network Connectivity Modification** (4.4.2):

  - *Goal & Why*: The objective here is to refine the lane geometries to address real-world traffic conditions, ensuring that the lane network accommodates transitions smoothly. After the initial hierarchical generation, certain common situations, like the diverging or merging of lanes, the presence of auxiliary lanes and the cycleway-bike lane connection cases, need special handling to maintain realistic continuity in the lane-level network. Addressing these scenarios is vital for creating a lane-level road network with topology and geometry correctness.

  - *Method*: This stage involves making precise adjustments to the previously derived lane geometries to incorporate transition areas. These modifications are based on the adjacency of targeted lanes, and their geometry as the referred geometry. In each step of this phase, we adapt the lane geometry to ensure lane-to-lane alignment.

- **Lane-level Network Connectivity Correction** (4.4.3):

  - *Goal & Why*: The goal of this phase is to correct errors and alignments in the generated lane-level network, ensuring both topological consistency and geometric accuracy. The use of grouped strokes as a foundation in the hierarchical approach may introduce alignment errors, particularly at intersections where stroke continuity is critical. Correcting these issues is essential to validate the generated lane network and to prepare it for the final areal representation stage.

  - *Method*: The corrections are divided into two key components: topology correction, which validates the continuous flow of lanes throughout the network, and geometry correction, which ensures that solely geometric errors would not exist and further affect the areal representation generation.

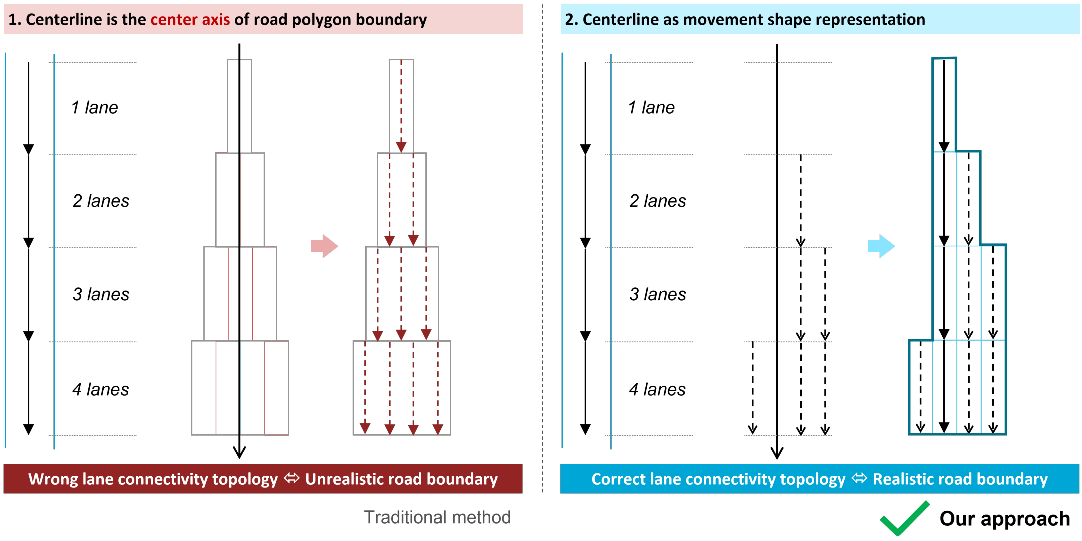


Figure 4.7.: Comparison of using the different assumptions for lane network and road polygon generation.

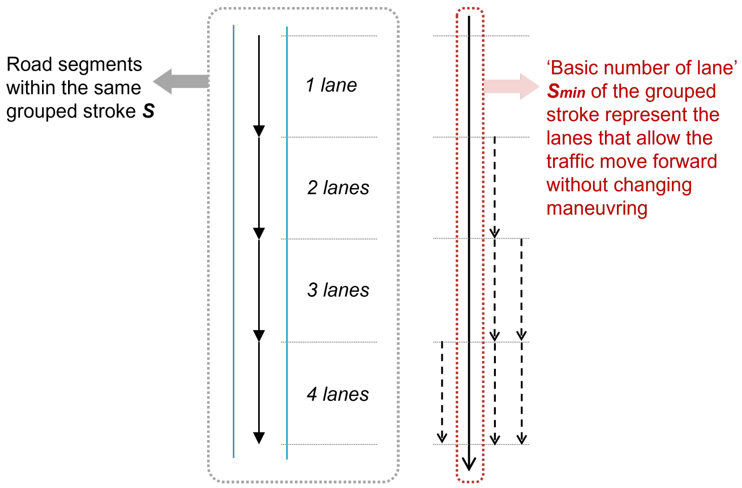


Figure 4.8.: Main factors for the hierarchical generation approach to achieve the 'good lane continuity'.

4.4.1. Hierarchical Reconstruction of Lane Centerlines

The objective of this step is to generate lane centerlines from each road centerline hierarchically. As previously discussed, the road centerline serves as a representation of the driving movement's trajectory. From a macroscopic perspective, a grouped stroke can represent a continuous traffic flow, with the exception of fork roads. At a more granular level, auxiliary lanes within road partitions can be considered as *branches* diverging from the main route (as illustrated in Figure 4.9). Consequently, the curvature of the centerline geometry is a reliable indicator of the driving trajectory for each lane, given that all lanes within a unidirectional road are generally parallel to one another, and hence, parallel to the original road centerline, and the hierarchical order selection results are the raw road indices I_r. The hierarchical process of road generation is depicted in Figure 4.10. Figure 4.11 is the compared illustration of the traffic-level network $\boldsymbol{R}_w$ and lane-level network $\boldsymbol{R}_l$.

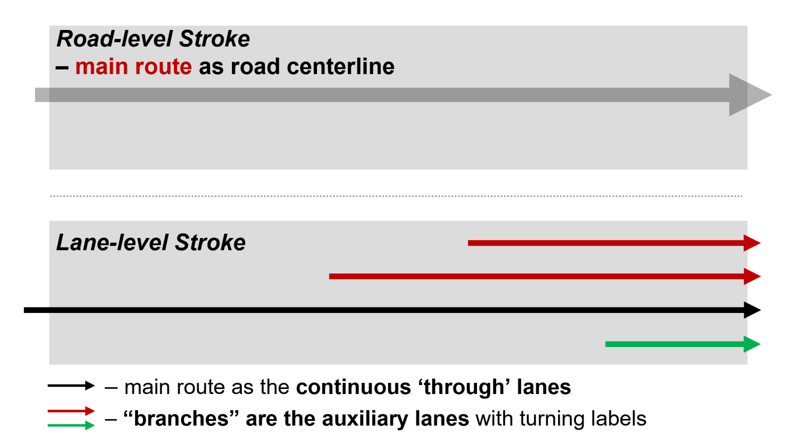


Figure 4.9.: Road-level and lane-level "main route" and "branches", expanding meanings from "stroke" concept.

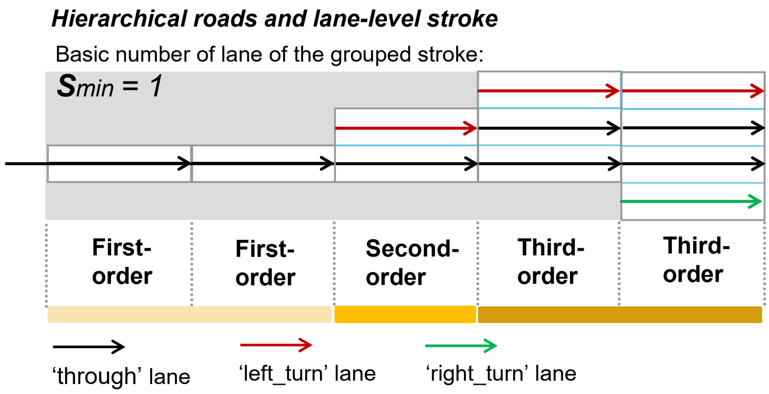


Figure 4.10.: Schema of hierarchical roads selection process and decision.

Note that the hierarchical reconstruction method is implemented after the grouped strokes $\boldsymbol{S}$ have been generated. When discussing the predecessors and successors of each road in the following descriptions, these terms refer specifically to the *local* connection relationships within their respective grouped stroke. There are four hierarchical levels in total. The selection process for the first three hierarchical levels focuses exclusively on one-way roads at the road level, represented by M_r, where the *'oneway'* attribute of road $\boldsymbol{r}$ is set to *'yes'*, particularly for motorized roads. All bi-directional roads at the road level will be processed in the last, fourth order.

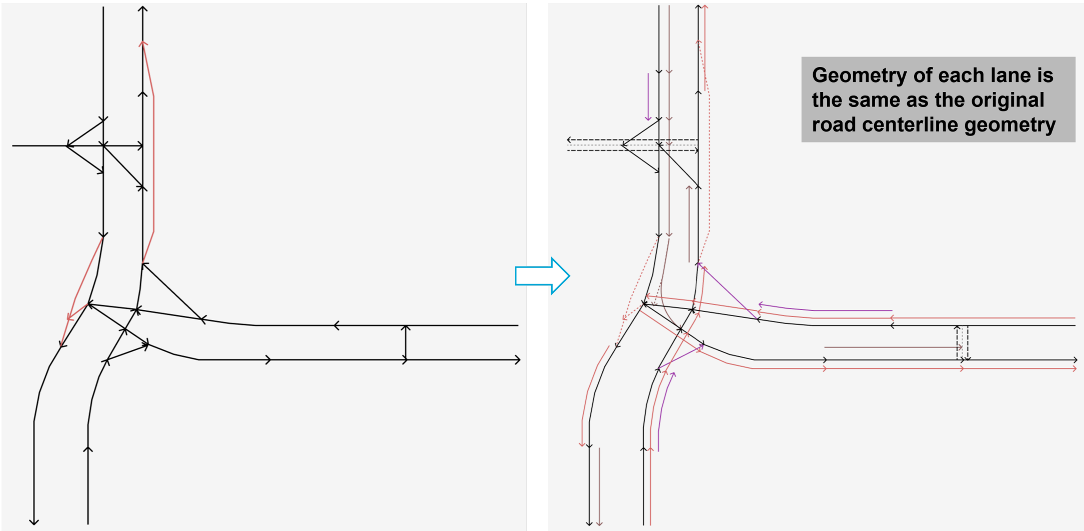


Figure 4.11.: Hierarchical generation result overview.

The outcomes of each hierarchical order of lane generation are as follows:

1. **Selected road indices:** For each level of hierarchical selection, a set of raw road indices (I_r) is identified as the group of roads selected for lane generation. This set determines which roads will undergo lane generation in that phase.

2. **Relative offset distance FD_l and the set of distances L_f:** For motorized roads and bike lanes in $\boldsymbol{R_w}$ sharing the same I_r, relative offset distances (FD_l) must be calculated from the raw road centerline geometry. These offsets ensure that lanes are positioned accurately relative to the centerline. The complete set of offset distances, L_f, is a list that orders all FD_l values from left to right, aligned with the vector direction of the raw road centerline.

3. **Lane centerlines:** Using the L_f values, the individual lanes ($\boldsymbol{l}$) are generated, forming the elements necessary for the lane-level network structure.

4.4.1.1. First-Order Roads

To effectively harness the geometric characteristics and ensure lane connectivity, the primary task is to identify road centerlines that accurately represent the **true center** of their corresponding road polygons. This identification is crucial, as these centerlines serve as the foundation for generating adjacent lanes in compliance with transportation regulations and lane alignment standards. Specifically, within each grouped stroke, the first-order roads—those that act as the **'offset reference'** for all lanes—are characterized by having the **minimum** number of vehicle lanes within the grouped stroke $\boldsymbol{s}$, with all lanes designated as 'through' lanes.

Before delving into the first-order road identification, we need to obtain the **minimum** number of vehicle lanes among the grouped stroke $\boldsymbol{s}$, denoted as s_{min}. Then, denote these roads as r_{h1} to indicate their generation as first-order roads. More specifically, the road segments that meet this criterion must satisfy the following conditions:

1. The road is motorized, i.e., the *highway* value H_w is not '*cycleway*';

2. All lanes in L_r (the set of lanes in the road-level network) have the same orientation, which allows the filtering of *forward* and *backward* $\boldsymbol{r}_w$ segments extracted from two-way roads;

3. For roads not labeled as forks (i.e., F_w is not labeled as *'left_fork'* or *'right_fork'*):

   a) **Rule 1**: All vehicle lanes of the road $\boldsymbol{r}_w$ must be 'through' lanes, meaning that all values in T_w are 'through';

   b) **Rule 2**: The number of vehicle lanes in L_w among all the roads satisfying Rule 1, must be the **minimum** number s_{min}. This rule is specifically designed to avoid scenarios where incorrect *turn:lanes* tag values mistakenly classify auxiliary lanes for turning as 'through' in T_w.

4. For slip roads labelled as forks:

   a) **Case 1**: If the road contains multiple vehicle lanes in L_w, all lanes must have identical turning values in T_w, such as 'right|right';

   b) **Case 2**: If the road has only a single vehicle lane, it is also selected, as it represents the main route of the ramp flow rather than the primary traffic flow within the grouped stroke.

Note that cycleways can be either one-way or bi-directional, but there is no strict **'lane alignment'** requirement for cycleways of different widths, as the traffic regulations are less stringent compared to motorized roads. Therefore, the hierarchical order is established solely for motorized roads. Regarding cycling paths, including cycleways and extracted bike lanes within the traffic-level network $\boldsymbol{R}_w$, the processing methods are as follows:

1. All separated cycleways (i.e., those with a H_r value of *'highway=cycleway'* in the road-level network $\boldsymbol{R}$, and with identical road indices I_r and I_w) can be treated as first-order motorized roads.

2. All bike lanes (i.e., those with a H_w value of *'highway=cycleway'* in the traffic-level network $\boldsymbol{R}_w$, and with differing road indices I_r and I_w) should be aligned parallel to the vehicle lanes. The exact positioning of the bike lane is determined by the hierarchical order of the corresponding first-order road r_{h1} to which it is affiliated.

   The generation process for bike lanes as first-order lanes involves the following steps:

   a) Utilize all I_r values from first-order roads to identify bike lanes that share the same I_r in $\boldsymbol{R}_w$;

   b) After generating all vehicle lanes, the precise location of the bike lane(s) is determined based on the attribute T_w. For example, if a bike lane is tagged with *'cycleway:left=lane'*, it will be positioned on the left side of the left-most vehicle lane.

After selecting all the first-order roads (including motorized roads and cycleways), lane width becomes the critical factor in determining the relative positions of the extracted lanes with respect to the raw centerlines, the centerline geometry of r_{h1} is positioned at the center of all vehicle lanes. A uniform width of 3.25 meters is assigned to all vehicle lanes. For separated cycleways, a width of 2.0 meters is applied to one-way cycleways, while bi-directional cycleways are given a width of 2.5 meters. Bike lanes are assigned a width of 1.25 meters per lane. These width values are then updated as a lane-level network attribute, denoted as W_l. According to the L_nw and W_l of each lane in L_w, we can obtain the lane geometries and the relative offset distances, store those new elements and attributes to l, Q_l, and L_f in r as the set of all lanes' offset distances. The detailed processing implementation is described in Section 5.4.1.

4.4.1.2. Second-Order Roads Once the 'through' lanes and affiliated bike lanes in the grouped strokes are generated, the selection and generation of second-order roads, denoted as r_{h2}, become more straightforward compared to the identification of first-order roads. Similar to the criteria for first-order roads, second-order roads are unidirectional motorized roads. After satisfying these prerequisites, the grouped strokes $\boldsymbol{S}$ are traversed to select the remaining roads within each grouped stroke $\boldsymbol{s}$, excluding all r_{h1} roads. The selection of r_{h2} is governed by the following criterion:

- The number of 'through' vehicle lanes in L_r (a set of lanes of the road r_{h2} at road level) equals the corresponding s_{min} value (the minimum 'through' lanes within the grouped stroke).

Although the hierarchical order selections are primarily based on vehicle lanes, the lane connectivity consideration must also extend to bike lane connections, as the affiliated bike lanes are generated during the first-order lane generation step. Therefore, for the r_{h2} roads, the offset distances of the 'through' lanes can be directly inherited from r_{h1}, since the number of 'through' lanes in all r_{h1} and r_{h2} roads equals the corresponding **minimum** s_{min}. If bike lanes were established during the first-order generation, the offset distances of the bike lanes on r_{h2} roads should remain consistent with those on the same side as the first-order bike lanes (see Figure 4.12). This approach ensures continuity in cycling movements, avoiding abrupt turns, and is particularly effective in regions that prioritize bicycle traffic, such as the Netherlands.

This step is designed to generate all vehicle lanes and affiliated bike lanes using a single road centerline geometry. The generation of second-order lanes involves two key steps:

1. **Adjust the set of turning labels T_r:** The set T_r is formed by combining two sets: one representing all vehicle lanes and the other representing the extracted bike lanes, both linked by the shared road index I_r. To correct errors in the raw data, such as incorrect turning labels (such as 'left|right|through'), it is necessary to refine the updated L_r to ensure the correct turning order and appropriate vehicle-bike lane arrangement:

    - Turning order: Treating T_r as a list, all 'through' lanes should be positioned in the center of the list. Left-turning lanes must be indexed before 'through' lanes, and right-turning lanes must follow 'through' lanes. A corrected example of T_r would be 'left|through|right'.

    - Bike lane arrangement: Prioritizing bike lanes requires them to be positioned directly adjacent to the 'through' lanes and between the 'through' lanes and any turning lanes. For example, a correct arrangement might be 'left|through|right-bike lane|right', or vice versa, depending on the bike lane configuration.

2. **Inherit offset values:** For each grouped stroke, the second-order roads will directly use the offset values of all 'through' vehicle lanes from FD_l. Consequently, parts of the offset values in L_f for r_{h2}, i.e. the FD_l of 'through' lanes in r_{h2}, are already determined.

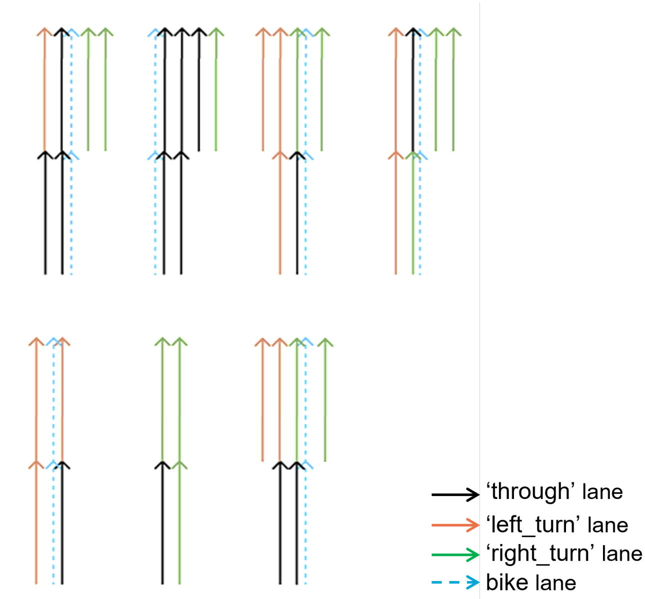


Figure 4.12.: 2nd-order offset rules: 'through' and bike lane matching.

3. **Generate bike and turning lanes:** Since the extracted bike lanes are located close to these 'through' lanes after the adjustment of T_r, the bike lanes can be generated afterwards. r_{h2} contain lanes with specific turning values T_l, following OSM pre-processing, the turning values are well-defined. The lane(s) that has a 'left' value in T_l should be positioned on the left side of the left-most lane (could be vehicle lane or *left_bike* lane) after inheriting the offset values, the offset value is determined by lane width L_w and the FD_l of referred left-most lane. Similarly, 'right' lanes should be generated using the same method, adjusting the offset direction to the right side. Store all offset distances FD_l in the lane network $\boldsymbol{r}_l$, and complete the L_f for all r_{h2} roads.

4.4.1.3. Third-Order Roads

After generating the first and second-order roads, some remaining roads have yet to be processed for lane generation. According to the selection criteria for the first and second-order roads, the remaining one-way roads (excluding bi-directional roads) are classified as third-order roads, denoted as r_{h3}, and they correspond to the following condition:

The number of 'through' lanes does not equal the **minimum** s_{min}. Unlike the first and second-order roads where 'through' lanes are directly matched to maintain **lane connectivity**, third-order roads involve resolving alignment issues between different turning labels on adjacent roads and lanes. For example, this includes addressing alignment between adjacent roads where one has T_w labelled as 'through|through', while the other has 'left|through|right', or dealing with matching bike lane configurations between connected roads.

Sequential Processing of r_{h3} Roads: Lane generation for r_{h3} roads follows a stepwise approach. It begins with targeting an r_{h3} road that has a predecessor belonging to the first or second-order roads, which already have completed lane generation and a full set of lane offset distances, L_f. If the successor of the targeted r_{h3} is also marked as third-order and still requires processing, the method continues sequentially through the grouped stroke until all r_{h3} roads have been traversed.

Lane-Matching Algorithm: To resolve mismatches between turning labels and achieve proper lane alignment in third-order roads, we introduce a **Lane-Matching** algorithm. The goal of this algorithm is to find the `best-matching-slice` between the turning labels T_r of two adjacent roads, $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$. Here, $\boldsymbol{r}_1$ is the predecessor of $\boldsymbol{r}_2$.

The `best-matching-slice` concept is similar to list slicing in programming, where a specific subset of lanes (the "slice") is selected to match the lanes between the two roads. Here we list the matching criteria and several factors which need to be considered, the illustrations of matching situations are shown in Figure 4.13.

1. **Number of lanes (L_n):** The number of lanes in both $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ determines how many potential shifts or alignments need to be evaluated. Each shift represents a different alignment option for the lanes, and the goal is to find the one with the best match.
2. **Turning labels (T_r) in $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$:** The turning labels T_r must be checked for correct order, as described in the second-order road phase. An ideal `best-matching-slice` is one where the turning labels T_l in $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ match lane by lane. The more lanes that match, the better the slice. If no lanes match, the matching score is 0.
3. **Bike lanes and their turning labels (T_r):** If bike lanes exist on both $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$, the `best-matching-slice` must ensure the bike lanes are aligned. If only one road has bike lanes, the bike lane's T_r should be excluded when comparing turning labels.
4. **Relative positioning:** In cases where multiple `best-matching-slice` options exist (i.e., more than one slice has the same number of matching lanes), or when no slice matches at all, relative positioning is used to resolve the ambiguity:
   - *Criterion 1:* Prefer the slice where the center of all lanes in $\boldsymbol{r}_2$ has the smallest offset from the center of all lanes in $\boldsymbol{r}_1$. This minimizes the geometric shift between the two roads and keeps the alignment smoother.
   - *Criterion 2:* If Criterion 1 does not resolve the ambiguity, prioritize right-side lane alignment. In right-driving regions, aligning right-side lanes (especially 'through' lanes) helps ensure a more reasonable traffic flow.

   By applying these criteria, the chosen `best-matching-slice` ensures proper lane alignment and maintains smooth transitions in traffic flow and road geometry.

The **Lane-Matching** algorithm is applied sequentially to each r_{h3} road. The resulting lane offset distances are stored in FD_l and L_f, updating the lane-level and road-level networks accordingly.

Matching offset distances and generating the remaining lanes: The output of the **Lane-Matching** algorithm is the `best-matching-slice` between the predecessor r_1 (with lane set L_{r1}) and the target road r_2 (with lane set L_{r2}). Since the lane orders in L_r and L_f are aligned, the offset distances (FD_l) from the matched lanes can be used to determine the positions of the new lanes in r_2. Additionally, the remaining lanes are generated by extrapolating from the already established FD_l values, based on their indices in L_f.

Table 4.1 below illustrates an example of the lane matching and generation process. It shows the predecessor and target road, their raw indices, turning labels, offset distances, the `best-matching-slice`, and the newly generated offset distances for the target road.

Road	Turning Labels (T_l)	Offset Distances (FD_l, L_f)	Best-Matching-Slice	Slice Indices	Inherited Offset Distances (*slice*)	Generated Offset Distances (*complete*)
Target r_2	left|through|right	N/A	through|right	[1, 2]	0.0|-3.25	**3.25**|0.0|-3.25
Pre_w r_1	through|right	0.0|-3.25	through|right	[0, 1]	0.0|-3.25	-

Table 4.1.: Example of best-matching-slice and generated L_f for third-order roads.

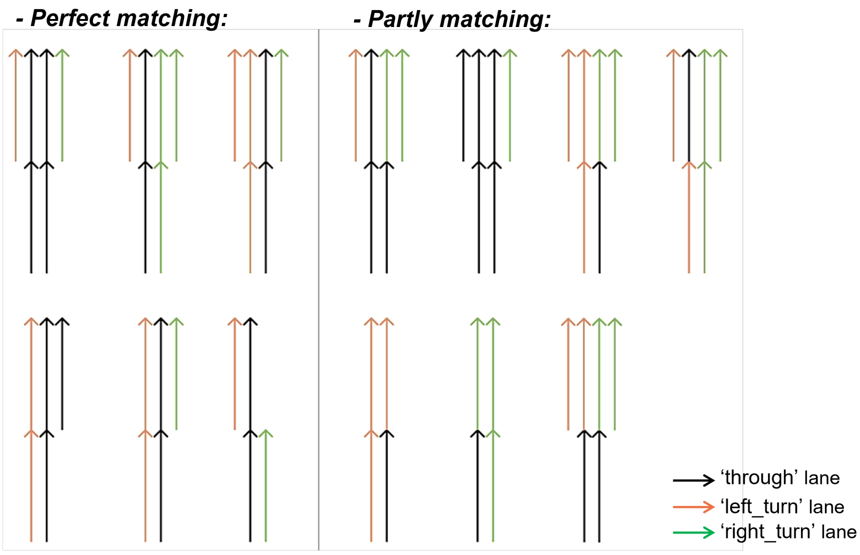


Figure 4.13.: In 3rd order offsetting process, the matching situation could be ideal or not ideal, i.e. adjacent lanes match partly.

4.4.1.4. Fourth-Order Roads Fourth-order roads r_{h4} include all bi-directional roads in the road-level network $\boldsymbol{R}$. As discussed in Section 2.3.2, analyzing raw OSM data reveals potential discrepancies between the vector direction of roads and the actual travel direction. This makes the **lane alignment** goal for bi-directional roads more challenging to achieve. To address this, we modify the concept of the 'road centerline' for two-way roads as the illustration in Figure 4.14. Specifically, in the fourth-order phase, the original road centerline is treated as the dividing line between *forward* and *backward* traffic flows. All *forward* lanes are offset to the right of the centerline, while all *backward* lanes are offset to the left. Additionally, the vertices of the *backward* lanes are reversed to ensure correct directional representation in vector format.

The key factors for generating fourth-order lanes are as follows:

1. **Number of lanes (L_w) for *forward* and *backward* directions:** For each bi-directional road $\boldsymbol{r}$, the total number of lanes is split into *forward* and *backward* lanes. Unlike the previous phases, here the two directional sets of lanes must be treated separately. The lane index within L_w determines its relative offset from the centerline, with larger index values corresponding to a larger offset distance from the centerline geometry.

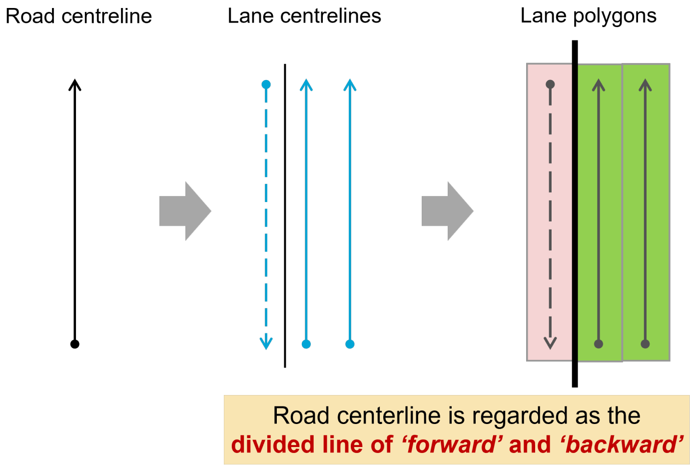


Figure 4.14.: Generation method for 4th-order road, the bi-directional roads.

2. **Bike lanes affiliated with *forward* and *backward* directions:** Although the *'cycleway=*'* tag is not explicitly associated with the *forward* or *backward* direction tags, the position of bike lanes can still be easily determined. A *'right_bike'* lane is placed adjacent to the rightmost *forward* lane, while a *'left_bike'* lane is placed next to the leftmost *backward* lane, with the vector direction reversed to match the correct traffic flow.

3. **Turning labels (T_w) for *forward* and *backward* directions:** Similar to L_w, turning labels for each direction are handled separately. The order of turning labels may need to be adjusted to ensure correct lane assignment. Regardless of the turning label T_l, if a lane is designated as *forward*, it will always be positioned to the right of the centerline; if it is a *backward* lane, it will always be on the left side of the centerline.

By applying these factors, the fourth-order phase ensures proper lane generation for bi-directional roads. Each lane is assigned a specific driving direction and a turning label (T_l) to indicate potential traffic changes. While perfect **lane alignment** between two-way roads may not always be feasible, auxiliary lanes—positioned on either the *forward* or *backward* side—can still be identified and adjusted similarly to how auxiliary lanes are managed on one-way roads. This is made possible by dividing bi-directional roads into two separate centerlines in the traffic-level network, each equipped with its own set of attributes stored within $\boldsymbol{R}_w$. These attributes allow for the creation of independent lane-level adjacencies for the *forward* and *backward* lanes, distinguishing and processing lanes in both directions while maintaining accurate adjacency within the lane sets L_w and L_r.

4.4.1.5. Special Instructions In certain special cases, such as unevenly distributed *forward* and *backward* lanes connected to an intersection, the lane generation process needs to be adjusted to maintain proper connectivity. These cases often occur in two-way roads where the number of lanes differs significantly between directions. When there is a substantial imbalance in the number of lanes between the two directions, connectivity issues at intersections are likely to arise. The primary reason for this problem is that multiple lanes adjacent to an intersection are generated using two different references for the "through" lanes, this discrepancy can lead to an imbalanced lane layout, complicating the **lane connectivity** maintaining.

To address this issue, an alternative method is required. In such cases, the raw road centerline should be reconsidered as the center of the final road polygon, which helps balance the lane generation on both sides. This ensures that the majority of lanes in one direction do not excessively incline to one side. This approach also ensures that **lane connectivity** is achieved in a reasonable manner, whether for single-single road connections or single-dual road connections. Additionally, the geometry of connected one-way slip roads will still accurately depict the curvature of movement in one direction as the general processing assumption (see Figure 4.15); the shape of the lanes can better reflect the physical world, and the integrity of lane-level connectivity is preserved even in complex configurations with uneven lane distribution.

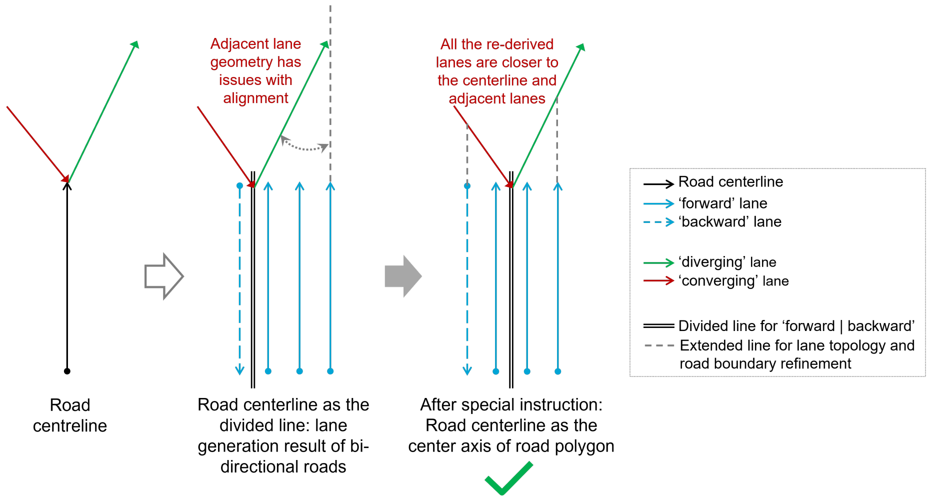


Figure 4.15.: Special instruction schema to resolve the issues caused by uneven lane distribution in some bi-directional roads.

The regenerated factors and approaches are:

1. Find two road segments in the lane-level network rw that share the same road index I_r but have different lane directions (reflect on the different I_w). The sum of the forward lanes Lnw^{for} and the backward lanes L_{nw}^{back} gives the total number of lanes for the road.

2. Road width is calculated from the lane width W_l and the total number of lanes. For each lane, a new offset distance FD_{il}^{b} is calculated to position the lane geometries symmetrically around the road centerline.

3. Let i represent the index of the lane in the road-level lane list L_r (starting from 0), and FD_{il}^{b} the offset for each lane:

$$
\begin{aligned}
&R\_width = W_l \times (L_{nw}^{for} + L_{nw}^{back}) \\
&FD_{il}^{b} = \frac{L_n - i}{2 \times W_l}
\end{aligned}
\tag{4.16}
$$

   A positive FD_{il}^{b} places the lane on the left side of the road, relative to the centerline, and a negative value places it on the right side.

Replace the original F_l with new FD_{il}^{b} and adapt the values in L_f for each lane of the special conditions, the new geometries can be generated following the modification.

4.4.2. Lane-level Network Connectivity Modification

Modification scenarios After completing the hierarchical reconstruction, all lanes are generated and positioned correctly to meet the **lane connectivity** requirements for adjacent roads. This subsection outlines the methods used to modify lane geometries, ensuring proper connections for slip roads, auxiliary lanes, and cycleways. Note that, in this subsection, all the motorized roads we modified are one-way roads at road level.

- Slip roads: A slip road must accurately connect to its predecessor's turning lane (the point where it diverges) and to its exact successor (the point where it converges).
- Separated cycleways: A cycleway may either connect to another separated cycleway or merge with a bike lane. The modification ensures that the 'cycleway-to-bike lane' junction is geometrically accurate and properly aligned.
- Auxiliary lanes: These lanes, typically used for turning maneuvers, contain a *'branch'* section that diverges from or converges to the main traffic flow. This branching occurs at the beginning or end of the lane geometry and is not parallel to the main travel direction. The modification ensures the correct alignment of these transitions.

4.4.2.1. Slip roads modification For each slip road, following the **Adjustment** process outlined in Section 4.3.4, when a *local* grouped stroke is interrupted by an intersection, the slip road typically forms the last segment of that stroke. At the road level, the predecessor of the slip road can always be found within the same stroke, but locating the successor requires crossing into other stroke groups.

1) '*left_fork*' or '*right_fork* slip roads.

Considering the **lane connectivity** at the lane level, it is necessary to use the labelled road fork type F_w, as this indicates the turning direction. For a road labelled '*through_fork*', the lane geometry is already correct. However, for roads labelled as '*left_fork*' or '*right_fork*', the slip road lane initially retains the shape of the raw slip road centerline, meaning the road centerline starts from the end vertex of the predecessor's centerline (VE_r) and ends at the start vertex of the successor's centerline (VS_r). The ideal geometry for these forks is:

- For a '*left_fork*' road, the modified lane geometry should start from the end vertex of the corresponding turning lane (typically an auxiliary lane marked with a *left* value in T_l), and end at the start vertex of the converging lane, generally positioned on the left side of the successor road.
- For a '*right_fork*' road, the same principle applies, but the alignment is based on the right side of the road.

To achieve the desired geometries for slip roads, the following information is required:

1. Local adjacency to identify the predecessor Pre_w within the same grouped stroke and global adjacency to locate the successor Suc_r across stroke groups.
2. The lane positions of both the predecessor Pre_w and successor Suc_r, which can be determined by their relative offset distances, L_f.

3. The lane geometries of the slip road, its predecessor Pre_w, and its successor Suc_r, all generated in Section 4.4.1. The key geometric elements are the end vertex of Pre_w and the start vertex of Suc_r.

For slip roads with multiple lanes, or in cases where auxiliary lanes lack the correct turning label (T_l), the most robust approach is to ensure that the leftmost lane of a *'left_fork'* connects with the leftmost lanes of both its predecessor and successor. The same logic applies to modifying a *'right_fork'*, ensuring proper alignment along the right boundary of the road.

2) Slip roads connect to two-way roads.

To handle slip roads connecting to two-way roads, beyond the *'left_fork'* and *'right_fork'* labels, we introduced the *'from_bi-direction'* and *'to_bi-direction'* types in Section 4.3.4. A *'from_bi-direction'* road diverges from a two-way road, while a *'to_bi-direction'* road converges into a two-way road. The modification method here also focuses on identifying the correct connecting lanes as their predecessor(s) and successor(s).

Given that the generation methods for one-way and two-way roads differ, using relative offset distances L_f to align the lanes between a two-way road and these fork roads is insufficient. Instead, we aim to ensure that the fork roads are aligned with both boundaries of the two-way road. This method guarantees that the lane connectivity between the two-way road and its associated fork roads is geometrically and topologically correct, as depicted in Figure 4.16.

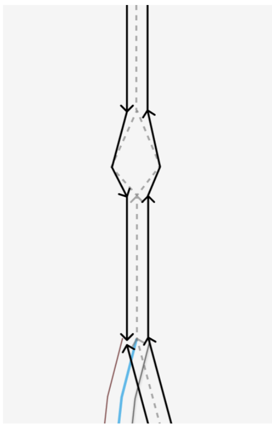


Figure 4.16.: Modify the lanes of fork-labelled roads to ensure the correct topology and geometry relations with their adjacent lanes in bi-directional roads.

By using the geometries of the predecessor's and successor's lanes as reference points, the curvature of the slip road geometry is largely maintained as represented by the raw centerline. Meanwhile, adjustments to the start and end points of the slip road are made to ensure **lane connectivity**, both in terms of topology and geometry. The modification scenarios are shown in the left image of Figure 4.17.

4.4.2.2. Cycleway modification The transitions between "travelling from a bike lane diverging into a separated cycleway" or "from a separated cycleway converging into a bike lane" are conceptually similar to the process described earlier for slip roads. Due to OSM pre-processing, the extracted bike lanes are assigned a different index from the raw road

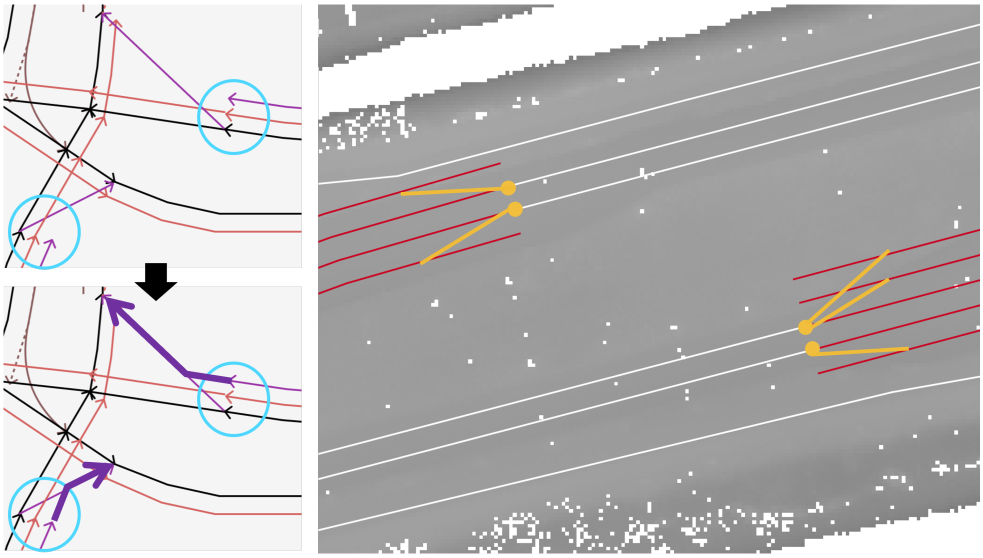


Figure 4.17.: Left: modify slip road lane centerline geometry. Right: modify auxiliary lane centerline geometry.

index, represented as I_r for roads and I_w for bike lanes in the traffic-level network $\boldsymbol{R}_w$. Additionally, strokes are grouped based on their *name* and *highway* types, which generate distinct group values (GS_w) to differentiate between vehicle and cycling traffic modes.

For each cycleway, the unique group value GS_w is used to establish its adjacency relations (Pre_w, Suc_w), and to determine whether the adjacent road is a bike lane by comparing their indices, I_r and I_w. If a "cycleway-bike lane" adjacency is detected, adjustments are necessary for the start or end vertex of the cycleway, as well as the partitioning of the cycleway's geometry.

As explained in Section 4.4.1, bike lanes are generated and placed adjacent to their corresponding vehicle lanes. In this modification step, we avoid altering the geometry of bike lanes, keeping them parallel to the vehicle lanes, and focus solely on modifying the geometry of separated cycleways. The methods, elements, and attributes required for these adjustments are identical to those used in the slip roads modification (see Figure 4.18).

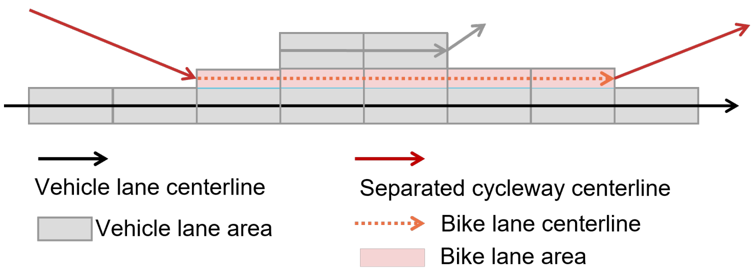


Figure 4.18.: Modify "cycleway-bike lane" centerline geometry

4.4.2.3. Auxiliary lanes modification After the completion of lane generation for each road centerline, the continuous curvature of a grouped stroke is transformed into lanes with distinct, accurate positions. However, adjacent roads may have differing numbers of lanes ($L_n w$) and unaligned lane distribution (L_f), resulting in "dangling" geometries of extra lanes after offsetting. The goal of this step is to resolve the dangling shapes by modifying the transition parts of those auxiliary lanes (modification method as shown in the right image of Figure 4.17).

Similar to previous modifications, the key is identifying the correct adjacent lanes for alignment. Auxiliary lanes diverge from the main route of their predecessor, and converge to the main route of their successor, so identifying the corresponding lanes in adjacent roads is crucial.

To address this, the following approach is used:

1. **Compare lane counts and offsets:** Traverse each 'non-cycling' grouped stroke $\boldsymbol{s}$. For each adjacent road pair, compare the number of lanes ($L_n w$) and offset distances (L_f) between the target road r_t and its predecessor r_{pre} or successor Suc_w. If the vehicle lane counts or L_f values are misaligned, auxiliary lanes require adjustment.
2. **Find the *matching-slice*:** Identify the *matching-slice* for each pair of roads based on their relative lane offset distances (FD_l). Using the *forward* direction as a reference, auxiliary lanes diverge from or converge to the leftmost or rightmost lane of the *matching-slice*.
3. **Modify the diverging and converging parts:** The target auxiliary lane (l_t) connects to the lane (l_c), which has the minimum relative distance to l_t. This minimum relative distance, denoted as $Dist_{min}$, guides the modification:

   $$Dist_{min} = FD_l^t - FD_l^c \tag{4.17}$$

   where FD_l^t is the offset distance of the target auxiliary lane, FD_l^c is the offset distance of the connected auxiliary lane.

   Combined with $L_n w$, denote the *matching-slice* between r_t and its predecessor r_{pre} as MS, the potential scenarios are as follows (positive $Dist_{min}$ means the left side, negative means the right side):

$$\begin{cases} \text{If } L_{nw}(r_t) > L_{nw}(r_{pre}): \begin{cases} \text{If } Dist_{min} > 0, & l_t \text{ of } r_t \text{ diverges from leftmost lane } l_c \text{ of } MS \text{ in } r_{pre}, \\ \text{If } Dist_{min} < 0, & l_t \text{ of } r_t \text{ diverges from rightmost lane } l_c \text{ of } MS \text{ in } r_{pre}. \end{cases} \\ \text{If } L_{nw}(r_t) < L_{nw}(r_{pre}): \begin{cases} \text{If } Dist_{min} > 0, & l_t \text{ of } r_{pre} \text{ converges to leftmost lane } l_c \text{ of } MS \text{ in } r_t, \\ \text{If } Dist_{min} < 0, & l_t \text{ of } r_{pre} \text{ converges to rightmost lane } l_c \text{ of } MS \text{ in } r_t. \end{cases} \end{cases}$$

4.4.3. Lane-level Network Connectivity Correction

In this final stage of lane-level network generation, both topological and geometrical corrections are applied to ensure the accuracy and connectivity of the lanes. The topology correction focuses on validating and fixing the lane connections across intersections and between grouped strokes; following this, geometry correction addresses errors in the lane geometries, ensuring that adjacent lanes have smooth and accurate geometrical correspondence. By applying two corrections, the lane geometries are adjusted to reflect the true structure of the lane-level network.

4.4.3.1. Lane Topology Correction From the raw OSM data, the model at road level has a graph-like structure with correct topology. During the generation process, **grouped strokes** and **lane connectivity** are used as constraints to maintain lane-level topology within each grouped stroke. However, cross-group topology (i.e., the lanes adjacent to another group of lanes) is not validated.

Since grouped strokes are commonly interrupted by intersections, roads potentially facing topological errors are slip roads or those representing 'virtual lanes' at intersections. As discussed in Section 4.4.2, labelled slip roads have already been modified using their cross-group successors. Thus, the primary focus for validation is the 'virtual lanes'. A typical characteristic of these roads is their intricate adjacency, having multiple predecessors and successors at all levels (A_c and AL_c as complete adjacency matrices representing the complexity). The accurate topology structure generation for virtual lanes is beyond the scope of this study. Therefore, to maintain **lane connectivity** as self-defined (mentioned in 4.2.1), we need to identify the 'main route' among the connected roads and validate their connectivity to achieve this goal.

Following the approach used in third-order generation for **Lane-Matching**, when prioritizing the main route for connectivity, matching relative offset distances (F_l) in adjacent roads becomes crucial. The validation method can be reduced to identifying mismatching issues of L_f between the target road and the referenced road (i.e., its predecessor or successor as the main route). The steps for detecting and correcting errors are as follows:

1. Mark the 'main route' connected roads for each road $\boldsymbol{r}$ (at the road level). The criterion is the interior angle, where the minimum angle indicates the 'through' driving direction and defines the 'main route'.

2. Compare the sets of relative offset distances L_f for adjacent road pairs. If there is no exact matching value between the two sets, it indicates that there is no **lane matching**, and the topology validation fails.

3. Specify the target road in the unconnected adjacent road pair (r_{tar}, r_{ref}):

   - If the predecessor(s) in the global adjacency of the validated road is a one-way road, and they are correctly connected, then this road is considered the referenced road r_{ref} and does not need regeneration.

   - Otherwise, roads with multiple predecessors in the global adjacency are likely incorrectly generated due to intricate traffic conditions. These are marked as r_{tar}.

4. Regenerate r_{tar} by modifying the endpoint of the road segment (start or end vertex). The slight correction of lane geometry adjusts the shared vertex to address the topological and geometric discrepancies while preserving curvature and overall topology. See the illustration in Figure 4.19.

4.4.3.2. Lane Geometry Correction After the topology correction, the final step of lane-level network generation, represented as linear features, is to fix the geometry errors between adjacent lanes. These errors are caused by the general **offset** geometric data processing method, which returns a (Multi)LineString at a specific distance from the object on either its right or left side. The validation method we used ensures consistent distances between adjacent lanes but does not address small errors, such as intersecting or minor gaps at the endpoints, which have not been resolved in previous steps.

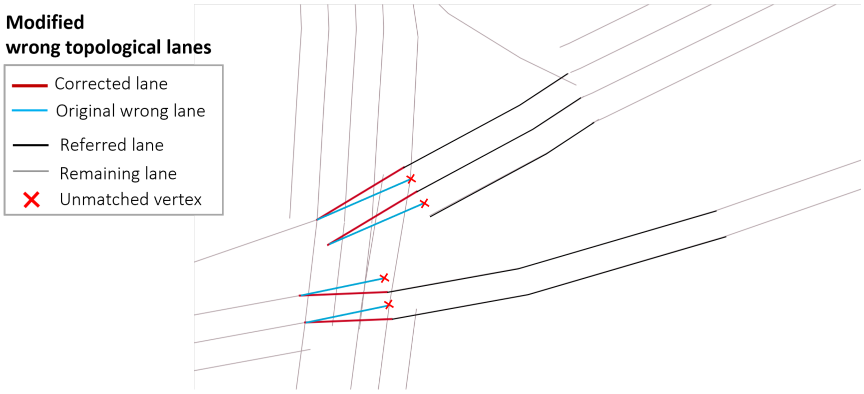


Figure 4.19.: Lane topology correction

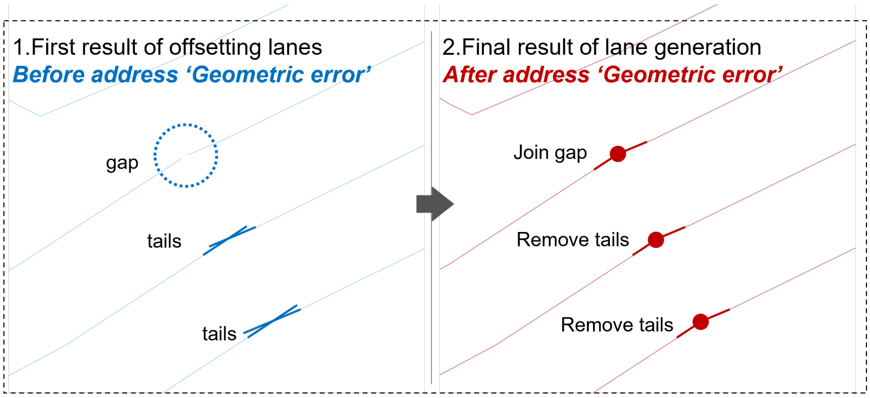


Figure 4.20.: Lane geometry correction

By applying the **Snap** and **Remove tails** operations (Figure 4.20), we further refine the generated lane geometries. The corrected geometry can then be regarded as a true representation of the lane-level network.

4.5. 2D Road Polygon Reconstruction

Introduction The 2D Road Polygon Reconstruction process transforms the linear representation of lane centerlines into a detailed areal representation, capturing the full extent of road surfaces and boundaries. This stage builds upon the lane-level network to produce road polygons that represent the road's spatial characteristics, including lanes, road widths, and intersections. The transition from lane centerlines to a complete road polygon involves a series of steps, including lane polygon generation, dissolution into road polygons, and post-processing for geometric accuracy.

At the 2D areal representation stage, operations are performed at the road level again to ensure the proper spatial and topological relationships between road polygons. The final areal representation ensures that overlapping sections are removed, distinct polygons are created for different areas, and the semantic information related to road attributes is preserved. The resulting polygons resemble "*ribbons*" that accurately represent the spatial arrangement of roads and intersections, allowing for a precise understanding of traffic space.

- **General Method for Polygon Generation** (4.5.1): This step creates road polygons by buffering lane centerlines to form lane polygons, which are then merged into cohesive road polygons for each segment.
- **Post-processing for Geometry Errors Detection and Modification** (4.5.2): This phase corrects geometry errors, such as overlaps and misaligned polygons, ensuring a compact and consistent representation. Node degree plays a key role in identifying true intersections, guiding the removal of overlaps and correcting polygon boundaries. Post-processing method addresses overlapping areas, connectivity issues, and the definition of intersection polygons. It fine-tunes the connected road ribbon boundaries, ensuring continuity and consistency across the road network. This step ensures that road polygons are free of overlaps and remaining errors removed as much as possible, which is vital for a clean and usable final representation.
- **Intersection Geometry Generation** (4.5.3): To accurately define intersection areas, capturing their complex spatial characteristics and ensuring they integrate smoothly with the rest of the road network. A "one-size-fits-all" approach generates intersection polygons for different configurations, ensuring smooth boundaries and proper alignment with road segments to ensure that intersections are properly modelled and represented.

4.5.1. General Method for Polygon Generation

The process of generating lane and road polygons from linear geometry involves the following steps:

First, buffer the linear geometry. Start by applying a buffer to each lane's centerline to create lane polygons. The buffer size is determined by the lane width, ensuring that the lane polygons accurately represent the physical space occupied by each lane.

Second, dissolve lane polygons. Once the individual lane polygons are generated, they are dissolved based on the same road index (I_r) to create a single, unified polygon representing the entire road. This process ensures that all lane polygons (L_P) belonging to the same road are grouped together into one road polygon (P_r).

Third, inherit attributes for the areal representation of roads (r_A). During the polygon generation, several key attributes are transferred from the linear road network to the areal representation, including L_n (number of lanes), D_r (3D relationships), and also derived the road width (W_r) and road polygon types (HA_r).

4.5.2. Post-processing: Geometry Errors Detection and Modification

4.5.2.1. Purposes of Post-processing Post-processing is essential for addressing geometric errors and ensuring the generation of compact and complete road polygons, as well as distinct intersection polygons. This phase corrects three primary issues:

1. **Tackle the overlapping and redundant areas:** While earlier steps corrected some lane-level errors, the use of a generic **buffer** approach during polygon generation often results in overlapping or redundant areas, even when the linear geometry is accurate.

2. **Solve the remaining lane connectivity errors:** Most two-way roads (excluding special cases discussed in Section 4.4.1) and virtual lanes at intersections, have inherited connectivity issues that were not fully addressed at the lane level. These errors propagate and become more pronounced during the polygon generation phase.

3. **Intersection Polygon Generation:** Intersections are complex areas formed by the joining of multiple lanes, including entering and exiting segments, and virtual lanes within intersections. These parts represent the spatial areas where vehicles slow down, stop, or change direction. However, in the linear road model, intersections are only identified by nodes and their degree. As a result, a specific method is required to distinguish intersection polygons from regular road polygons, ensuring that the spatial representation of intersections accurately reflects their functions in the road network.

4.5.2.2. Node Degree-based Road Polygon Correction Method At the road level, **node degree** is crucial in determining whether an area is a standard road connection or an intersection that requires special treatment. The '**true node degree**' of an intersection, denoted as N_t, must match the number of joint roads (J_r) at that intersection node $\boldsymbol{c}$. The polygons and boundaries of the connected roads are formed by all lanes converging at the node. Figure 4.21 interprets the different node degrees and their road polygons.

Based on the node degree, we propose a correction method for generating both road and intersection polygons, structured around the true node degree N_t:

1. **Node degree N_t is 2:**
   - When the node degree is 2, the roads at this node form a straightforward connection between two road segments.
   - The method we designed ensures that the two road polygons are directly connected without gaps or overlaps. These distinct road polygons have unique road indices, which can be linked back to the raw road index from OSM.
   - The polygons encompass all necessary lane polygons, accurately representing the transportation space as it exists in the physical world.

2. **Node degree N_t larger than 2:**
   - When the node degree is greater than 2, this indicates an intersection where multiple roads converge.
   - In such cases, the intersection polygon needs additional elements to depict transition areas, which connect to the entering and exiting lanes. These parts represent the spatial areas where vehicles slow down, stop, or change direction. The boundary lines separating these sections, often resembling stop lines in traffic design.

Note that, the definition of '**true node degree**' is based on how intersections are identified in the improved road network. Since the network has been refined to a lane-level and distinguishes between motorized traffic and cycling traffic, intersections must also consider these traffic modes. An important aspect of this concept is that an intersection is only identified when more than two intersection legs of the same traffic type (either motorized

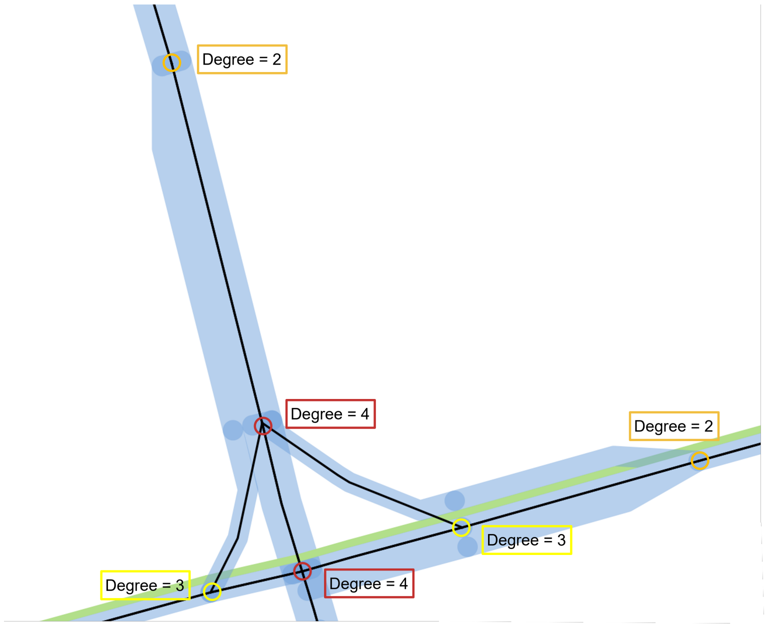


Figure 4.21.: Different Node degree and the corresponding polygons.

or cycling) converge. If the total number of joint roads at a node exceeds two but fewer than three roads of the same type meet, these roads are considered to be "*passing-through*" the area without a turning function. As a result, the true node degree is determined by filtering out these "*passing-through*" roads, leaving only the relevant lanes contributing to the intersection.

4.5.2.3. Node Degree and Road Polygon Correction The node-based method is essential for resolving polygon topologies, particularly in how road polygons connect at intersections and ensuring geometric consistency throughout the road network. As discussed in Section 2.3.2, OSM data often presents issues with road centerlines, where intersections are not accurately represented. For instance, in a "T" junction, the expected node degree (N_k) is 3, but if one road segment continues through the intersection without breaking, the derived node degree might only be 1, leading to the incorrect polygon topology and a failure to identify the true intersection.

To address geometric errors by revealing their relationship with node degrees, the following steps are proposed:

1. **Recalculate the node degree:** When a road centerline is not properly split at an intersection (e.g., when a road intersects in the middle of another segment), split the road centerline into two parts. This ensures proper node representation and updates the node degree by counting the correct number of road segments converging at the intersection.
2. **Remove overlapping parts for road ribbons:** Focus on nodes with a true node degree $N_t = 2$, which typically represent roads that should directly connect without gaps or overlaps. Most nodes with $N_t = 2$ are within grouped strokes. After recalculating the node degree, we can further identify road pairs with the same true node degree by

filtering out false intersections based on road types (i.e., highway type H_r). Apply a general method to remove overlapping areas and create distinct road ribbons for the connected road pairs, ensuring each has the correct attributes (see left image in Figure 4.22).

3. **Clip protruding areas to address errors:** When the node degree is incorrect due to issues at the road level, errors such as protruding polygons in intersections, like protruding area in a "T" junction, can occur (see right image in Figure 4.22). These errors can be detected and resolved using node-degree-based methods, the applied scenarios are:

   - Identify roads of different types, which lead to differing road polygon widths and potential protrusions.

   - Compare the recalculated node degree with the original, and trim the endpoints of road polygons that improperly extend into uninterrupted road centerlines.

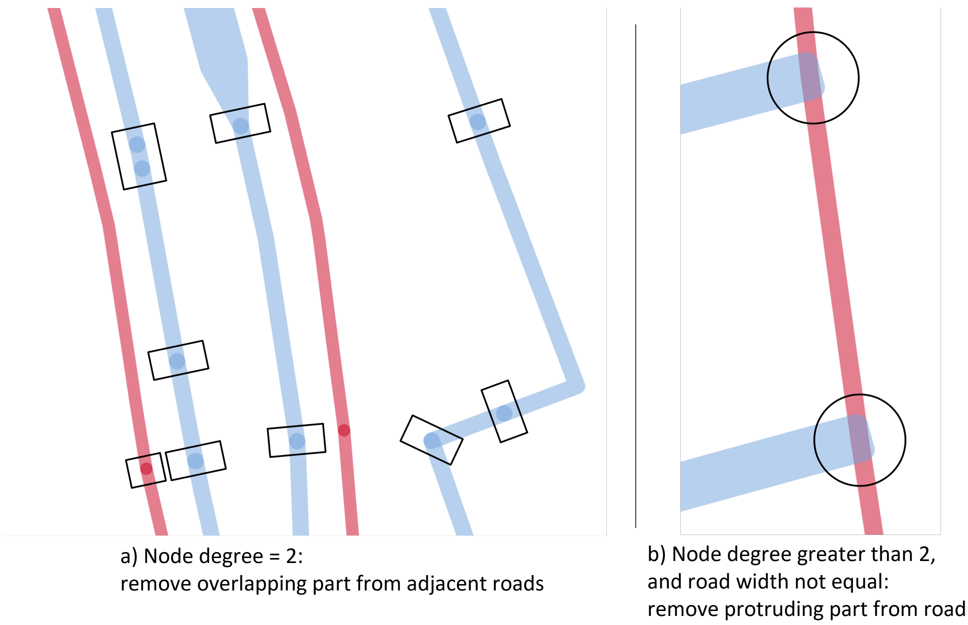


Figure 4.22.: Post-processing methods driven by the corrected true node degree.

After applying the previous modifications and corrections to node degree, true node degree, and road polygons, the set of intersections $\boldsymbol{C}$ at the road level must be updated. Following Eq. (4.3), each true intersection $\boldsymbol{c}$ is revised by replacing the original node degree N_k with the recalculated true node degree N_t. Additionally, roads associated with minority traffic types and "passing through" roads are removed from the set of joint roads J_r. Consequently, the entering roads JS_r and exiting roads JE_r are adjusted to reflect the corrected intersection structure. Each updated road polygon $\boldsymbol{r}_A$ and intersection in linear model $\boldsymbol{c}$ serve as the basis for subsequent intersection generation.

4.5.3. Intersection Geometry Generation

Real-world intersections are often complex due to the variety of *intersection legs* and spatial arrangements. In our lane-level digital model, the full adjacency matrix for all involved lanes (including entering, exiting, and turning lanes) has not been fully recovered. Instead of analyzing and classifying intersections individually, we adopt a **'one-size-fits-all'** solution for generating all intersection types. The primary objective is to generate road-level intersection polygons that offer a compact, accurate boundary for each intersection, using the entire set of road polygons.

The complexity of intersection types makes this approach necessary:

1. **Varied intersection configurations:** Intersections can range from simple to highly intricate, involving multiple roads, lane types, and traffic modes. A uniform solution simplifies the generation process.
2. **Missing virtual lanes and turning corners:** In transportation design, intersections often include entering lanes, exiting lanes, virtual lanes and corner geometries to accommodate turning movements and other kinematic rules. These are absent in the lane polygons, in such cases, the intersection polygon needs additional elements to depict transition areas, so the intersection geometry generation must capture the full transportation area and resolve the chaotic node structures at intersections.
3. **Intersection polygon goals:** The final intersection polygon must represent a distinct area, including smooth turning corners, while ensuring it fits tightly with the connected road polygons. The boundary lines separating road polygons and intersection polygons, often resemble the stop lines concept in traffic design Zhang et al. [2016]. There should be no gaps or overlaps between the intersection and the modified road polygons.

By leveraging semantic information, we can categorize intersections based on the traffic modes they support:

- Motorized road intersections: Designed for vehicle traffic. If bike lanes are affiliated with vehicle lanes, they are included in the motorized road polygon for intersection generation.
- Cycleway intersections: Comprised solely of separated cycleways, these intersections serve bicycle traffic.
- Cycleway-bike lane intersections: A special type designed for transitions between separated cycleways and affiliated bike lanes, ensuring smooth connectivity for cyclists travelling between the two.

Steps for generating an intersection polygon To generate an intersection polygon from the linear model $\boldsymbol{c}$ and road polygons $\boldsymbol{RA}$ (as outlined in the areal model in Eq. (4.9)), follow these steps:

1. **Retrieve Node Degree and Road Polygons:** At the intersection node $\boldsymbol{c}$, retrieve the joint road indices J_r with the generated road polygons in $\boldsymbol{RA}$. Use JE_r (entering roads) and JS_r (exiting roads) to identify the merging positions on the corresponding road polygons based on the road-level graph topology. Then, retrieve the associated road polygons for further processing.

2. **Trim Joint Road Polygons:** Set a trimming length to split each joint road polygon into two parts: the trimmed part converges toward the intersection node, representing the portion of the road that merges into the intersection; the remaining part still reflects the normal road space (see Figure 4.23). The trimmed parts are stored in R_P of the intersection polygon $\boldsymbol{c}_A$, while the remaining portions are updated in the road polygon $\boldsymbol{r}_A$ as the new r_P.

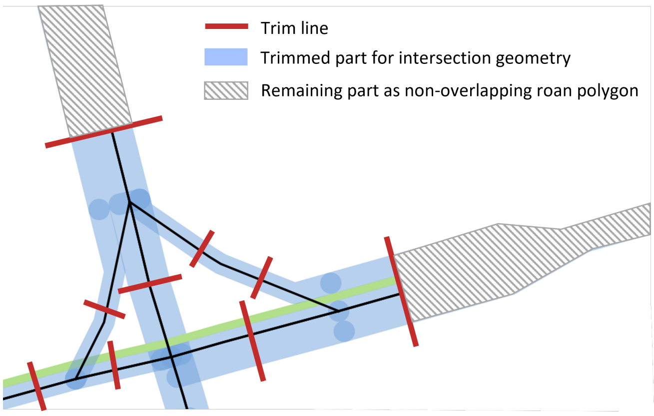


Figure 4.23.: Schema of dividing intersection geometry from road polygons

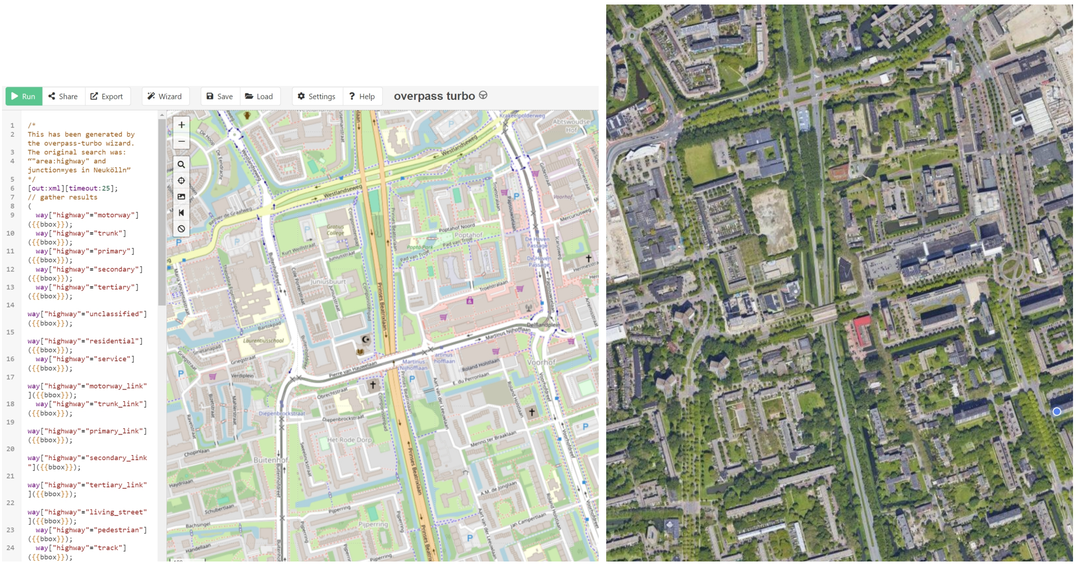
3. **Traverse and Store Trimmed Parts:** Traverse all joint roads, extract the trimmed parts, and store them in the set of merged polygons R_P. The number of trimming actions and elements stored in R_P should match the node degree N_k.

4. **Merge Trimmed Parts to Form Intersection Polygon:** Finally, merge all the trimmed polygons in R_P to create the intersection polygon. Store this distinct polygon as P_c, finalizing the intersection polygon generation.

5. Implementation

5.1. Datasets, software and libraries

Datasets To develop and test the proposed OSM-driven road network reconstruction method, OSM road datasets have been collected. The datasets used in this research are derived from OSM using the Overpass API. Data was retrieved in both XML and GeoJSON formats to capture detailed information on road geometries, attributes, and metadata for a defined region (Overpass API for data collection as shown in Figure 5.1a). This dataset includes road types (e.g., motorways, cycleways, primary roads and residential streets), centerline geometries, and key attributes such as lane counts, highway types, turn directions, and bearing angles. The test region is Delft, the Netherlands, and covers an urban built environment with approximately 400 road segments, and also encompasses a variety of intersections and mixed multimodal transportation scenarios (Figure. 5.1b), providing a comprehensive and detailed base for lane network generation and high LoDs model reconstruction. The use of GeoJSON format allows for efficient spatial analysis, attribute extraction and visualization, while XML provides hierarchical, detailed metadata on the road network structure.



(a) Overpass turbo (b) Satellite image of research region

Figure 5.1.: Dataset and region.

Software and Libraries The implementation of this research involved a range of software tools and libraries, chosen for their capabilities in geospatial data processing, visualization, and road network analysis.

The primary programming environment was Python, with development carried out using PyCharm and JupyterBook within Visual Studio Code. These environments facilitated the integration of various geospatial and computational libraries.

The core software and libraries supporting the methodology included:

Geospatial Visualization and 3D Tools

- **QGIS**: This open-source GIS software was used for visual inspection and verification of mid-product outputs. Specifically, it was applied to visualize the generated road network segments, check connectivity, and assess geometric accuracy during the intermediate stages of processing.
- **MeshLab**: Primarily used for 3D visualization, MeshLab enabled the examination of road meshes and 3D geometric structures during the advanced stages of the project, where 3D mesh generation and representation of road networks became crucial.

Road Network Analysis

- **OSMnx**: This library was fundamental for downloading and analyzing OSM road network data. It allowed easy retrieval of road network graphs and provided functionalities to analyze network attributes such as mode degree (the number of connections a node has) and neighbour relationships. OSMnx was particularly useful for creating the base road network graph used throughout the project.
- **NetworkX**: In conjunction with OSMnx, NetworkX was utilized for the topological analysis of the road networks. It handled the underlying graph structure, allowing for the calculation of shortest paths, connectivity analysis, and topology validation at various stages of the lane network reconstruction.

Geometric and Geospatial Data Processing

- **Shapely**: Shapely was critical for geometric data manipulation. It provided tools to perform operations such as buffering, intersection, and union on geometric objects, which were essential for processing road centerlines and generating lane geometries from them. It also helped maintain the spatial relationships between various road segments.
- **Geopandas** and **pandas**: Geopandas extended the capabilities of Pandas to handle geospatial data. It was used to read, write, and process GeoJSON files and other spatial data formats. The library was instrumental in managing geospatial attributes such as road geometry and tags (e.g., highway types) that were necessary for the classification and analysis of road segments. Pandas, a powerful data manipulation library, was used for handling the non-geometric attributes of the OSM data. It was employed to manage large tabular datasets, filter and classify road segments based on their attributes, and perform data aggregation.

Computational and Mathematical Tools

- **Numpy**: Numpy was utilized for numerical operations throughout the project. It was used to perform geometric calculations, such as angle computations between road segments, and for efficiently handling array-based data structures that represented the road network's geometric properties.

- **Matplotlib**: This library was employed for data visualization, allowing the creation of custom plots and figures. It was particularly useful for visualizing road and lane networks, displaying geometric relationships between road segments, and generating plots that illustrate the results of the lane reconstruction process.

Together, these tools formed a robust computational framework for the generation, analysis, and visualization of lane-level road networks, supporting the step-by-step methodology outlined in this research.

5.2. OSM Road Network Initialization

5.2.1. OSM data pre-processing

Steps & Notice: To tackle the various arrangements of extracted roads, backward bike lanes, etc, we need to follow some steps in the implementation process.

1. **Extract various arrangements of bike lane:** need to notice the 'cycleway'='lane' means 2 sides have lanes; the vector direction need to reverse; and vector direction also needs to adjust if the bike lane is 'cycle:left', it means on the 'backward' side;

2. **Order of *turn* value:** Fix the 'turn' tags: each lane has its 'turn' label, and always be encoded as a list, 0 index of the list is the leftmost lane in the driving direction of the carriageway that lane belongs to;

3. **Number of lanes:** If the *lanes=** and *turn:lanes=** have no value, and *oneway=yes* as a one-way street, then set the number of lanes as 1; if reference tags are empty and value of *oneway=** is not 'yes' (commonly found in 'living_street', 'residential' and 'unclassified' roads, set the number of lanes as 2.

4. **Align the attributes:** All the motorized roads $\boldsymbol{r_w}$ in traffic-level $\boldsymbol{R_w}$ represent single directional roads, number of lanes L_{nw} equals the amount of values contains in turning attribute T_w. In a nutshell, every lane has its turning label.

5.3. Grouped Strokes Generation

Specifics in first phase: Initial grouping

1. Handle condition 1: Address the cases that roads have no '*name*' values. In the data we use for this study, a few of the roads (less than 5%) have no values of '*name*' key. To avoid errors, assign two initial unique group index values for all *non-name* motorized roads and cycleways.

2. Handle condition 2: Group the specific types with their non-link counterparts. As mentioned in Section 4.3.4, the '*_link*' suffix indicates a ramp road, thus the road with the suffix is one of the embranchments of its main route; and so on.

3. Handle condition 3: Group *'highway=residential'* and *'highway=living_street'* roads which have the same road *'name'*, since the level of these two road types in the urban road hierarchical structure is low, and the definitions of them are ambiguous. Therefore, grouping two types together when they share the same road *'name'* is beneficial to obtain more continuous grouped strokes rather than many partitions that are unable to reflect the driving routes.

Figure 5.2 are the initial grouping results for dividing strokes.

Implementation of Second Phase: Adjustment This phase addresses the misclassification of ramp roads through the following steps:

1. **Identification of Predecessors and Successors:** Establish two types of adjacency connections within the traffic-level graph, denoted as G_w.
   a) **Global Adjacency:** Create adjacency connections for all roads in the network, identifying the global set of predecessors and successors, denoted as Pre_r^g and Suc_r^g.
   b) **Local Adjacency:** Establish adjacency connections within each grouped stroke, identifying the local set of predecessors and successors, denoted as Pre_r^l and Suc_r^l.
2. **Traversal and Comparison:** Traverse all motorized roads in G_w, comparing Pre_r^g and Pre_r^l for each road r_w. Based on the discrepancies and predefined detection criteria (illustrated in Figure **??**), adjust the group index for diverging roads. Specifically:
   - A misclassified road will lack a predecessor in the local adjacency Pre_r^l, but will have a single predecessor in the global adjacency Pre_r^g, which corresponds to the main route from which it diverged.
   - The identified predecessor in Pre_r^g must have multiple successors in Suc_r^g, one of which is the targeted slip road to be corrected.
3. **Adjustment of Group Index:** Reassign the misclassified slip road to the group of its correct predecessor Pre_r, removing it from its original grouped stroke. The GS_w value of the adjusted road is updated to match the GS_w value of Pre_r.
4. **Recreate local adjacency:** Use the new group strokes to recreate the adjacency for the grouped roads. Update the Pre_w and Suc_w for each road $\boldsymbol{r}_w$.

Figure 5.3 are the grouping results of adjustment for correcting the slip roads of strokes.

Fork labelling This section describes the detailed implementation for labelling forks, with a special focus on one-way roads.

To identify and label diverging one-way roads, we use the method outlined in the previous **Adjustment** phase and focus on roads of the *'_link'* highway type. The key criterion for identifying these roads is as follows: If a road has multiple successors in its Suc_w set, and the direction of this road aligns with its successors, then all these successors are considered as *forks*. These branches are the primary targets for labelling in this step:

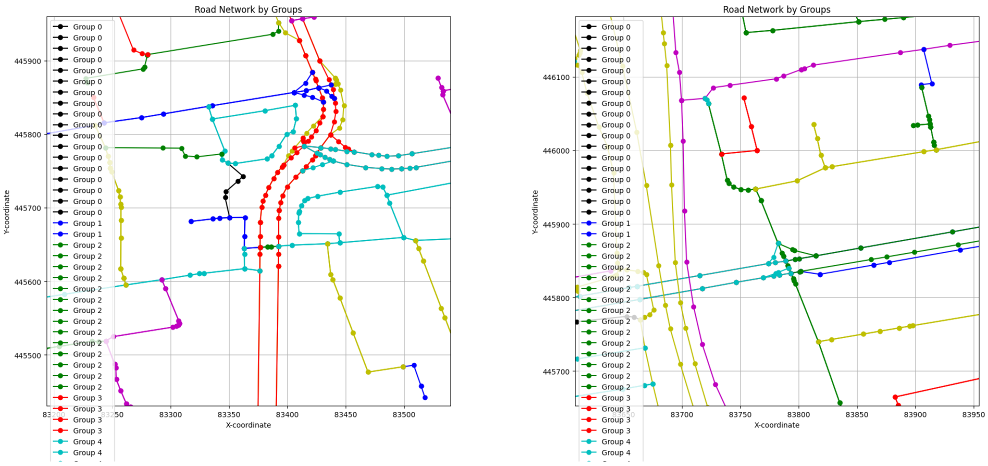


Figure 5.2.: 1st grouping result

- **Detailed Angle Calculation:** The exact fork label for each *branch* is determined by calculating the interior angle θ_t between the branch and its predecessor. The method for calculating this angle is similar to the approach described in Equation (4.13) in Section 4.3.3.

- **Determining Diverging Orientation:** To classify the orientation of each diverging branch as left, through, or right, we denote θ_p^e as the end bearing angle of the predecessor road, and θ_r^s as the start bearing angle of the target road (the branch). The orientation of the fork is then determined based on these angles:

  - *Left Fork*: Occurs when the angle difference indicates a counterclockwise rotation.

  - *Through Fork*: Identified when the angle difference is minimal, suggesting alignment with the predecessor.

  - *Right Fork*: Occurs when the angle difference indicates a clockwise rotation.

$$\text{Fork Type} = \begin{cases} \text{right}, & \text{if } 0^\circ < (\theta_r^s - \theta_p^e) < 90^\circ \text{ or } 0^\circ < (360^\circ - \theta_p^e + \theta_r^s) < 90^\circ \\ \text{left}, & \text{if } 0^\circ < (\theta_p^e - \theta_r^s) < 90^\circ \text{ or } 0^\circ < (360^\circ - \theta_r^s + \theta_p^e) < 90^\circ \\ \text{through}, & \text{if } |\theta_r^s - \theta_p^e| < 2^\circ \text{ or } (360^\circ - \theta_p^e + \theta_r^s) < 2^\circ \text{ or } (360^\circ - \theta_r^s + \theta_p^e) < 2^\circ \\ \text{None}, & \text{otherwise.} \end{cases} \tag{5.1}$$

The implementation results for all the labelling fork roads are shown in Figure 5.4.

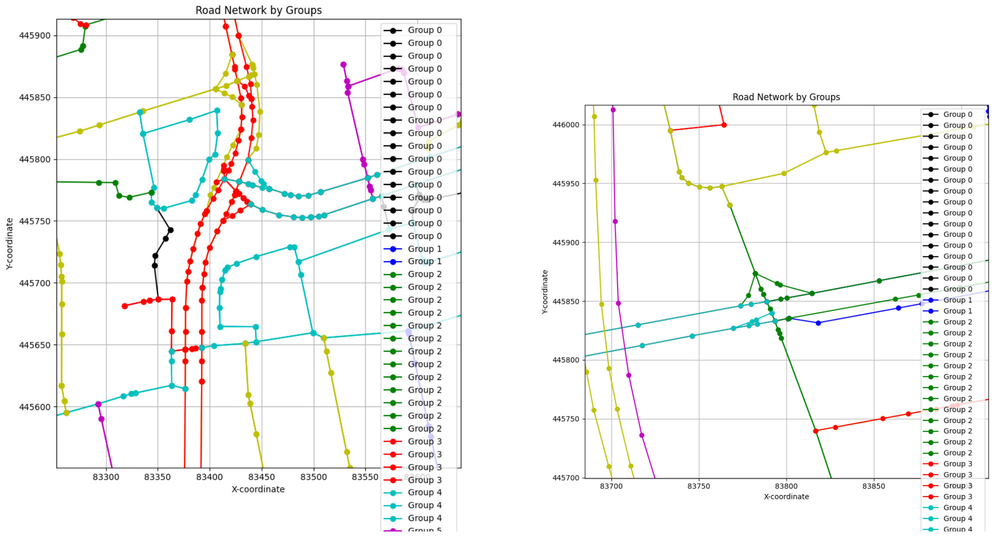


Figure 5.3.: 2nd grouping result

5.4. Hierarchical Reconstruction

5.4.1. Specifics of first-order lane generation

The process of generating first-order lanes is outlined below. The detailed method for offsetting first-order roads is provided in Algorithm 5.1:

1. Extract the number of vehicle lanes from L_w. The centerline geometry of r_{h1} is positioned at the center of all vehicle lanes. The offset method varies depending on whether the number of lanes is odd or even. The relative offset distances are stored in F_l, one of attributes of lane $\boldsymbol{r}_l$, representing the set of each lane's relative offset distance from the road centerline.

2. Identify the affiliated bike lane(s) and calculate their relative distance to the centerline geometry of r_{h1} based on the T_w attribute and the neighboring vehicle lane(s)' offset distances. These relative offset distances are also stored in F_l.

3. For all cycleways, the lane width is treated as the road width, and therefore, no offset implementation is required in the linear representation.

4. Combine the vehicle lanes and affiliated bike lanes to form the complete set of lane offset distances in the road-level network $\boldsymbol{R}$ as L_f, where these lanes share the same raw road index I_r.

5.5. Lane-level Connectivity Modification and Correction

Specifics of geometric implementation the correction of these scenarios is identified and designed to solve the topology issues first to achieve the **lane connectivity** goal. In "Method-

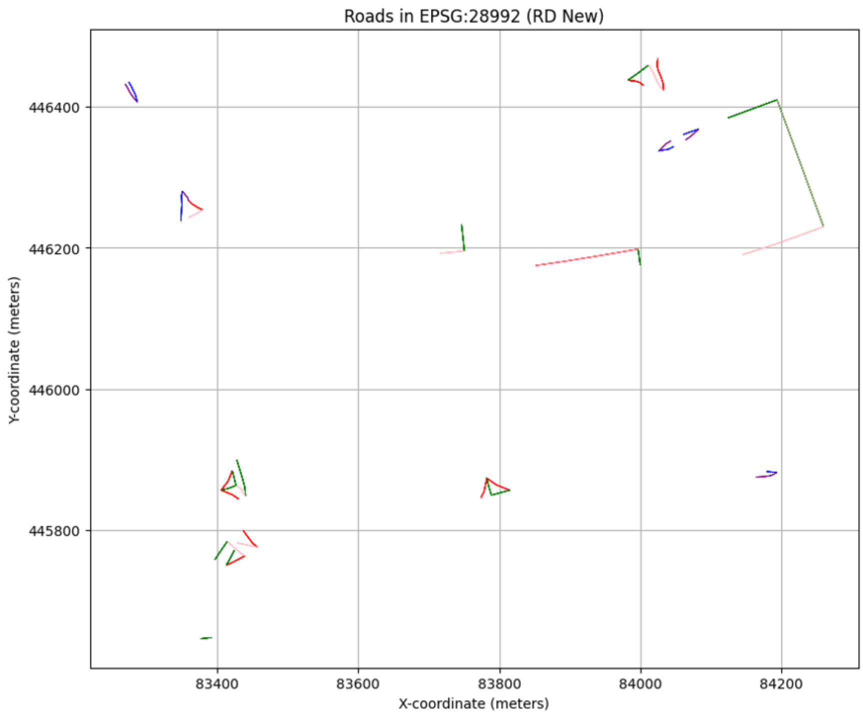


Figure 5.4.: Fork results

ology" Section 4.4.3 we focus on detecting the target lanes and their correct reference lanes. While the final outcomes are the lanes with correct topology and geometry relationships, thus we need to delve into the details of geometric processing as the attached explanation.

1) slip roads and cycleways

For all the scenarios below we aim to fix, we conducted a similar geometric processing method since whether the targets and references are, the goal is the same: to use a new vertex as the new endpoint of the lane segment, meanwhile keeping the original curvature as much as possible.

- slip roads;
- cycleways;

To depict the traffic flow movements from the curvature of line geometry, we choose to draw the extended line (or reversed extended) of the referred lane and find the intersecting point as the point where two adjacent lane centerlines meet. The extra extended segment is collinear with the referred lane geometry, which is regarded as the true traffic flow; thus, using the extended segment to replace the original segment(s) as the joint section, the new geometry is generated and satisfies the topology and geometry requirements spontaneously.

2) auxiliary lanes

As we know, the correction for auxiliary lanes aims to develop the *wedge-shape* for the start or end partition of the auxiliary; for the rest partitions of lane, still remain parallel to the lanes of main route. The main focus of this geometric processing method is to locate the point on the original lane centerline as the endpoint of the *wedge-shape*; then draw a simple line to the referred lane. One of the options that we used for implementation is to set a distance from the referred node, then use the perpendicular line PL_t of traffic flow to get a new perpendicular line PL_n that intersects with the target lane centerline, the only one intersecting point is the endpoint we need for the newly generated *wedge-shape* line.

Double implementation of one lane

For some lanes, the geometry needs to be adjusted twice, for instance, the start vertex and segment of a slip road are processed according to the predecessor lane, and the successor lane also shifts to a relative distance from the raw centerline. Encountering this condition, the corrected lane geometry should be stored in the data model as V_l, thus, the sequential implementation can be conducted based on the previous one, to avoid the errors and duplicated procedures.

5.6. 2D Road Polygon Generation

Intersection Generation The intersection generation process (illustrated as Figure 5.5 shown) involves handling various types of intersections through tailored solutions to ensure accurate areal representation and seamless connectivity between road polygons. Here are the main solutions used for different cases:

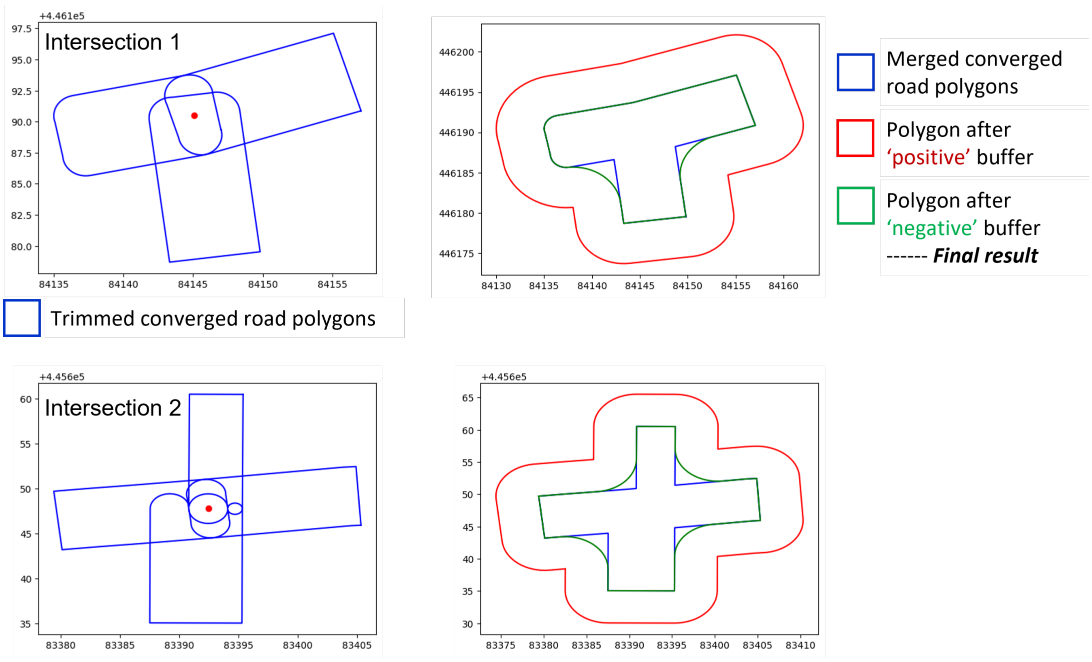


Figure 5.5.: Intersection buffer method for geometry results.

- **Custom Trim Lengths:** For different intersection types, the trimming length is adjusted based on factors like the width of joint roads. This flexibility ensures that the turning areas are well-defined and suitable for the intersection geometry.

- **Handling Short Roads:** If a road segment is shorter than the specified trimming length, it is not trimmed further. Instead, the entire road polygon is used to form the intersection polygon, maintaining continuity in the areal representation.
- **Continuous Road Polygons:** When an intersection node O_c is located along the centerline of a continuous road, the road polygon is trimmed twice to create two intersection *legs*. This process preserves the structure of the intersection and ensures smooth connectivity between segments.
- **Buffer Method for Turning Corners:** Using the Shapely library's `buffer` function, turning corners are generated through a two-step buffering process:
    1. **Dilation:** The merged road polygon is expanded using a positive buffer value.
    2. **Retraction:** The expanded polygon is then contracted with a negative buffer value, with the radius of the turning point defined in this step. This process creates smooth curves that represent the extra area required for vehicle turning movements.
- **Roundabouts as Circular Intersections:** Roundabouts are treated as a combination of smaller intersections and internal short road polygons. Since roundabouts are typically represented as closed polylines in OSM data, if a converging road in the set J_r is labelled as a roundabout in attribute D_r, all associated roads and their trimmed polygons are combined. This forms a single intersection polygon that covers the entire roundabout area, including the connecting protruding parts.
- **Merging Overlapping Intersection Polygons:** The initial method treats any node with $N_k > 2$ as an intersection. However, real-world complex intersections often consist of multiple smaller intersections, allowing diverse traffic flows. After generating individual intersection polygons, an additional step identifies and merges overlapping intersection polygons. If two polygons P_c overlap, they are combined into a single polygon that covers the entire transition area. This iterative process ensures that all intersections are represented as independent, non-overlapping areas, accurately reflecting the complex geometry of real-world intersections.

Algorithm 5.1: FIRST-ORDER LANE CENTERLINES GENERATION

Input: A first-order road centerline geometry r_{h1}, the number of its vehicle lanes *n_lanes*, vehicle lane width *v_width*, bike lane width *b_width*, the position side of bike lane T_{h1}, *highway* type H_{h1}, raw road index I^r_{h1}, and new road index I^w_{h1}.

Output: *lanes_dict*: the dictionary of lane offset distance relative to r_{h1}, equivalent to L_f of $\boldsymbol{r}$ in $\boldsymbol{R}$.

Initialize $lane\_dict \leftarrow \{\text{'lane'}:[\,], \text{'cycle'}:[\,]\}$;
Initialize $flag \leftarrow$ `True`;
$road\_o \leftarrow \frac{v\_width}{2}$;
if $H_{h1} \neq$ *'cycleway'* ***and*** $I^r_{h1} = I^w_{h1}$ **then**
  Initialize $lane\_lst \leftarrow [\,]$;
  for $i \leftarrow n\_lanes$ ***to*** 1 ***step*** -1 **do**
    if *flag* **then**
      if $i = 1$ **then**
        $lane\_lst$.`append`(0);
      else
        $lane\_lst$.`append`$(-road\_o \times (i-1))$;
        $flag \leftarrow$ `False`;
    else
      $lane\_lst$.`append`$(road\_o \times i)$;
      $flag \leftarrow$ `True`;
  $lanes\_dict[I_r][\text{'lane'}] \leftarrow$ `sorted`$(lane\_lst,$ `reverse=True`$)$;
if $H_{h1} =$ *'cycleway'* ***and*** $I^r_{h1} \neq I^w_{h1}$ ***and*** $I_r \in lanes\_dict.$`keys()` **then**
  $bike\_o \leftarrow \frac{b\_width}{2}$;
  if *'right'* $\in I^w_{h1}$ **then**
    $right\_d \leftarrow (lanes\_dict[I_r][\text{'lane'}][0] - road\_o - bike\_o)$;
    $lanes\_dict[I_r][\text{'cycle'}]$.`append`$(right\_d)$;
  else if *'left'* $\in I^w_{h1}$ **then**
    $left\_d \leftarrow (lanes\_dict[I_r][\text{'lane'}][1] + road\_o + bike\_o)$;
    $lanes\_dict[I_r][\text{'cycle'}]$.`append`$(left\_d)$;
  $lanes\_dict[I_r][\text{'cycle'}] \leftarrow$ `sorted`$(lanes\_dict[I_r][\text{'cycle'}],$ `reverse=True`$)$;
return *lanes_dict*

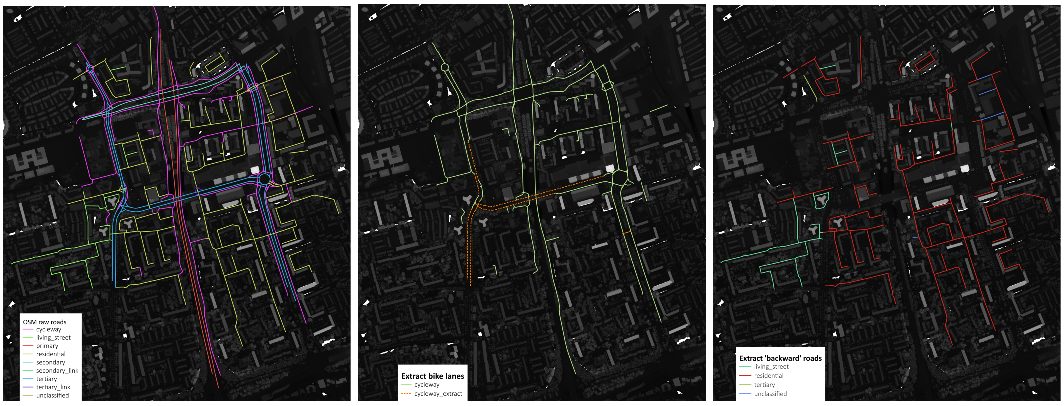
6. Results and analysis

The development of a comprehensive digital road model is composed of multiple intermediary stages, each contributing to the final result. This section outlines the key sub-results, progressing from raw OSM road-level networks to lane-level networks, the areal representation of the road network.

6.1. Results of 2D linear representation

6.1.1. Results of pre-processing and grouped stroke generation

The goal of OSM data preparation and pre-processing is to extract additional roads and lanes, complete the road-level network $\boldsymbol{R}$ to facilitate the development of the traffic-level road network $\boldsymbol{R}_w$.



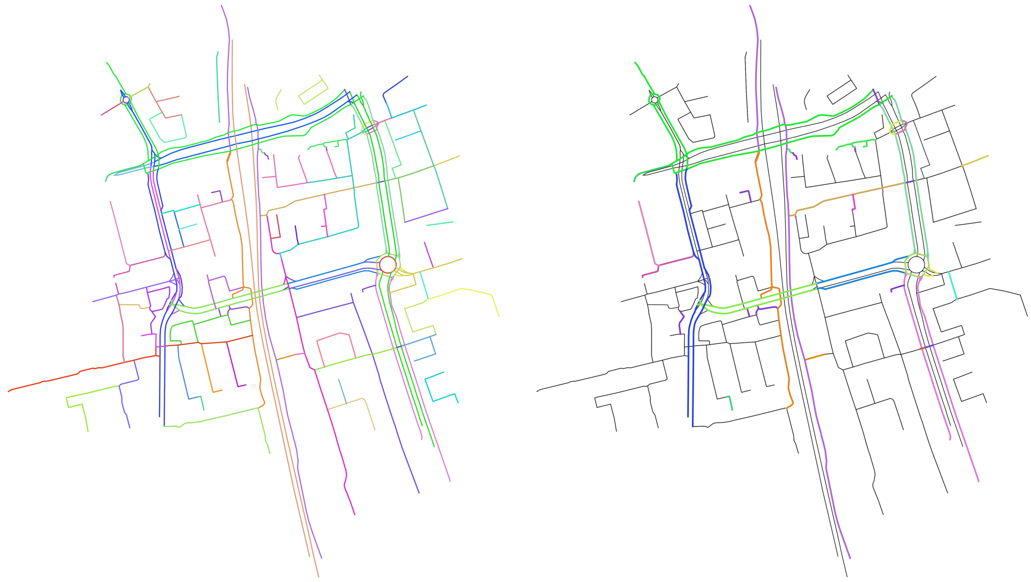
(a) OSM raw roads (b) Extracted bike lanes (c) Extracted 'backward' roads

Figure 6.1.: OSM road data pre-processing results.

1) OSM data pre-processing:

1. Raw data has 393 elements as the road centerlines (see Figure. 6.1a);
2. Extracted bike lanes from raw data, 41 bike lanes, including 3 *left_bike* lanes, and 38 *right_bike* lanes (see Figure. 6.1b);
3. Identified 120 two-way roads, so added 120 *backward* road centerlines with the reversed vector direction (see Figure. 6.1c).

4. Total number of elements after data pre-processing is 554, we use 554 road centerline as the traffic-level road network for the following lane generation.



(a) First result of grouped strokes, distinguished by random colors

(b) Grouped strokes of cycling roads are visualized with colors, motorized roads are black

Figure 6.2.: First result of grouped stroke for all roads after pre-processing.

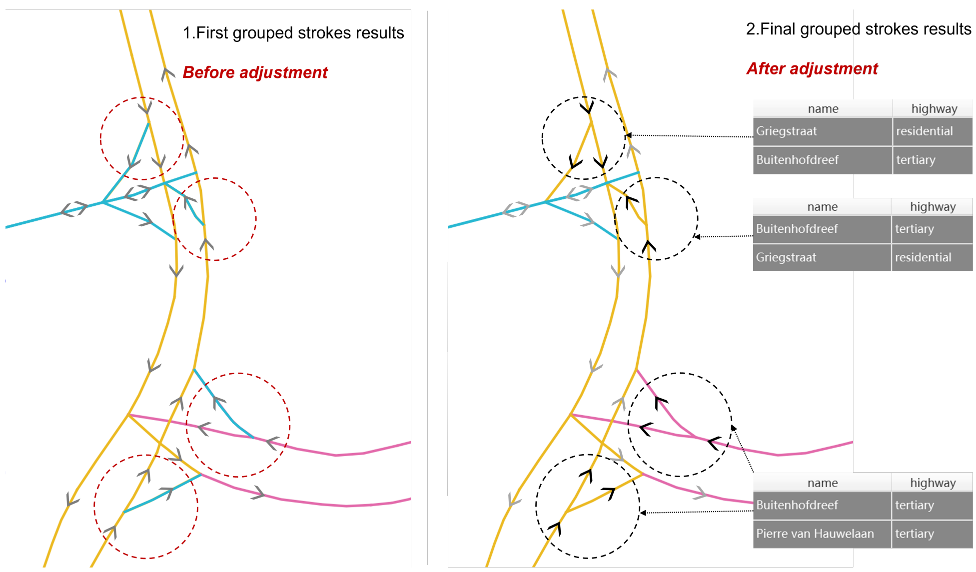


Figure 6.3.: Comparison of the adjustment results of wrongly labelled slip roads.

2) Grouped strokes generation: Used 554 elements to generate grouped strokes S. As we stated in Section 4.3.4, the generation includes two steps, to validate that the steps are valid and essential, here we show the first group generation result and the adjusted result:

- First grouped result: There are 104 groups, of which 29 groups are cycleway strokes, as shown in Figure 6.2a, 6.2b, respectively.
- Second grouped result: 104 groups, and fixed 8 slip roads to the corrected groups, according to the predecessors and traffic flows. In Figure 6.3, the segments with the same color labeling belong to the same group. After the adjustment implementation, the incorrectly placed slip roads have been adjusted to match the group stroke of their predecessors.

Figure 6.4.: Results of all the labelled slip roads.

2) Slip roads labelling: Further labelled all the slip roads to the types we stated in Section 4.3.4. 42 roads are labelled from 554 roads in total, including the extracted *backward* roads (see Figure 6.4 and 6.5):

- 9 ***left_fork*** roads, of which 5 roads are the ramps from one-way roads, and the other 4 roads belong to bi-directional roads (i.e., marked as *'forward'* or *'backward'* roads;
- 10 ***right_fork*** roads, of which 9 roads are the ramps from one-way roads, and 1 road belongs to bi-directional roads;
- 13 ***through*** roads, of which 6 roads are the ramps from one-way roads, and 7 roads belong to bi-directional roads;
- 5 pairs as the slip roads diverge and converge to one two-way road, so 5 ***from_bi-directional*** and 5 ***to_bi-directional*** roads.

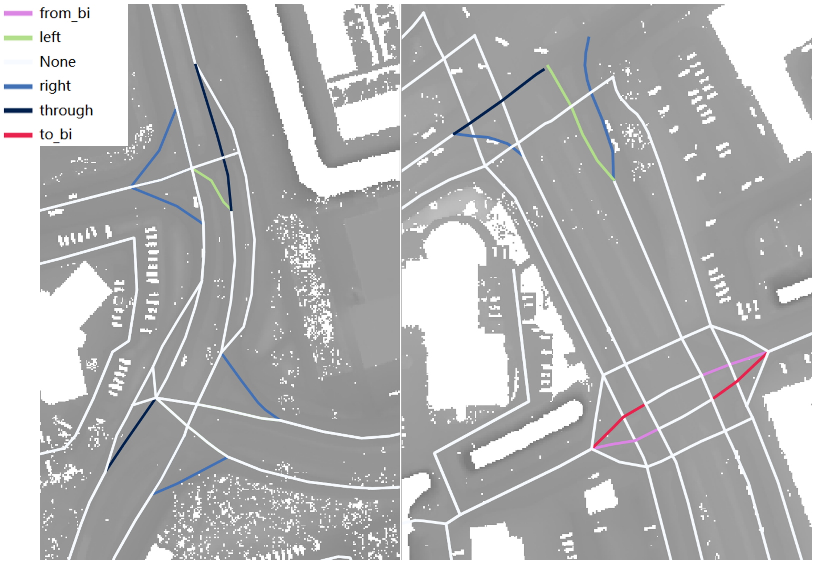


Figure 6.5.: Representative slip roads labels.

6.1.2. Results of hierarchical reconstruction

Lane-level network generation is a procedural process, from the first-order road selection, to the primary lane-level results, to the correction version of the final lane-level network $\boldsymbol{R}_l$, we show the mid-products and the sub-results as the demonstration of the methodology and implementation.

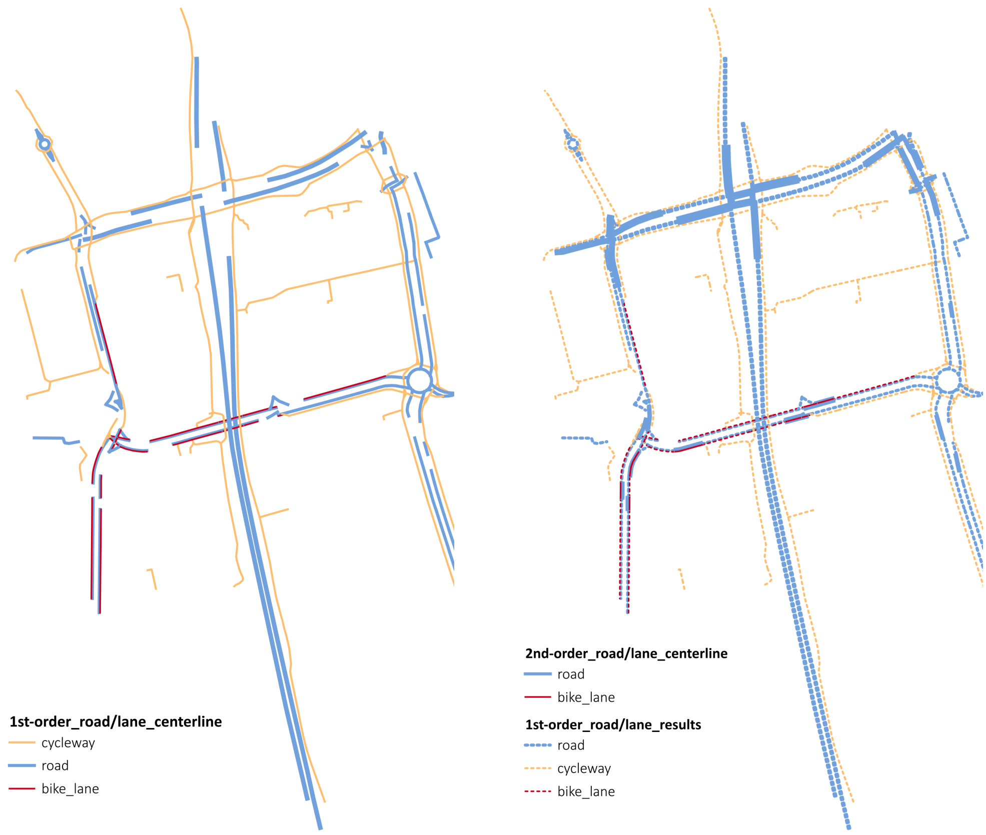
Before choosing the roads for the first-order roads, the minimum number of vehicle lanes for each stroke s_{min} is a vital attribute as a reference for all the road-level one-way roads. Thus, the number of lanes L_n and L_{nw} in $\boldsymbol{R}$ and $\boldsymbol{R}_w$ are the results that we obtain in the first phase, OSM data pre-processing, but also one of the prerequisites of this hierarchical reconstruction. In Figure 6.6 we can see the difference of L_n for each road, and combined with the result of grouped strokes, the minimum number of vehicle lanes for each stroke s_{min} is confirmed for the further processing.

Figure 6.6.: Number of lanes in each road after data pre-processing.

Results	**First-order (r_{h1})**	**Second-order (r_{h2})**	**Third-order (r_{h3})**	**Fourth-order (r_{h4})**
Number of roads r	237	42	7	107
Number of lanes l	258	127	20	252
Number of vehicle lanes	111	119	15	246
Number of cycleways	126	0	0	0
Number of bike lanes	21	8	5	6
Number of slip roads	21	4	0	17
Number of auxiliary lanes	0	54	3	32
Number of forward lanes	-	-	-	121
Number of backward lanes	-	-	-	125
Road direction M_r	Oneway	Oneway	Oneway	Two-ways

Table 6.1.: Results of all the order roads and generated lanes.

1) first-order roads r_{h1} reconstruction According to the criteria we designed in Section 4.4.1, we chose 237 roads as the first-order roads; and after the lane generation, we obtained 258 lanes in total as the result of this step. The results include 111 vehicle lanes, 126 cycleways and 21 bike lanes (see Figure 6.7a).



(a) First-order roads and generated lanes (b) Second-order roads and generated lanes

Figure 6.7.: First-order and Second-order roads and generated lanes with accurate location.

2) second-order roads r_{h2} reconstruction The second-order road selection is based on the number of lanes L_n and s_{min} of the stroke. The difference between the first-order and second-order results is the generation of auxiliary lanes. Besides the 'through' lanes and the potential extracted bike lanes, the auxiliary lanes to turn are derived and located accurately. We selected 42 roads in total as the result of this step, generated 127 lanes. The results include 119 vehicle lanes (65 'through' lanes, and 54 auxiliary lanes with specific turning labels), and 8 bike lanes (see Figure 6.7b).

3) third-order roads r_{h3} reconstruction After the selection of first and second order, the rest of the one-way roads are the third-order roads. Using the 'Lane-Matching' method we can generate the lanes, including 'through' lane, auxiliary lanes and bike lanes. We selected 7 roads in total as the result of this step, generated 20 lanes. The results include 15 vehicle lanes (12 'through' lanes, and 3 auxiliary lanes with specific turning labels), and 5 bike lanes. Figure 6.8 shows the result of first-order, second-order and third-order lanes, further, in Figure 6.9 we can observe the location of lanes to prove the results satisfy the **lane connectivity** goals.

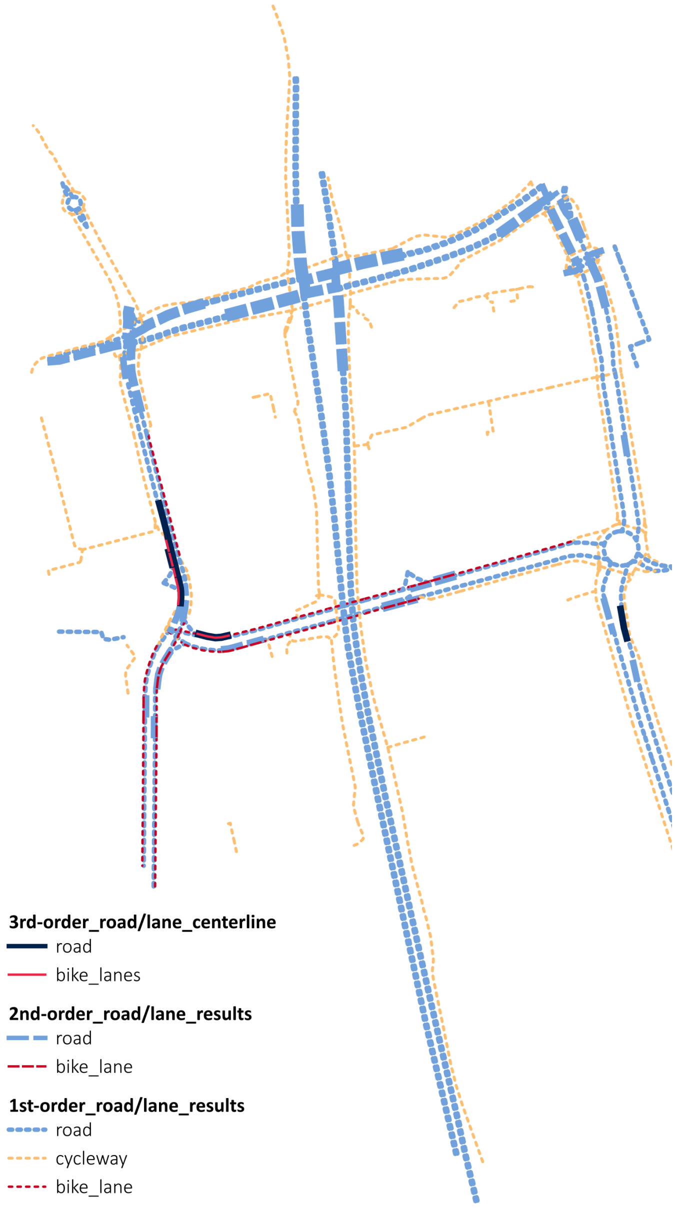


Figure 6.8.: Third-order roads and generated lanes with accurate location.

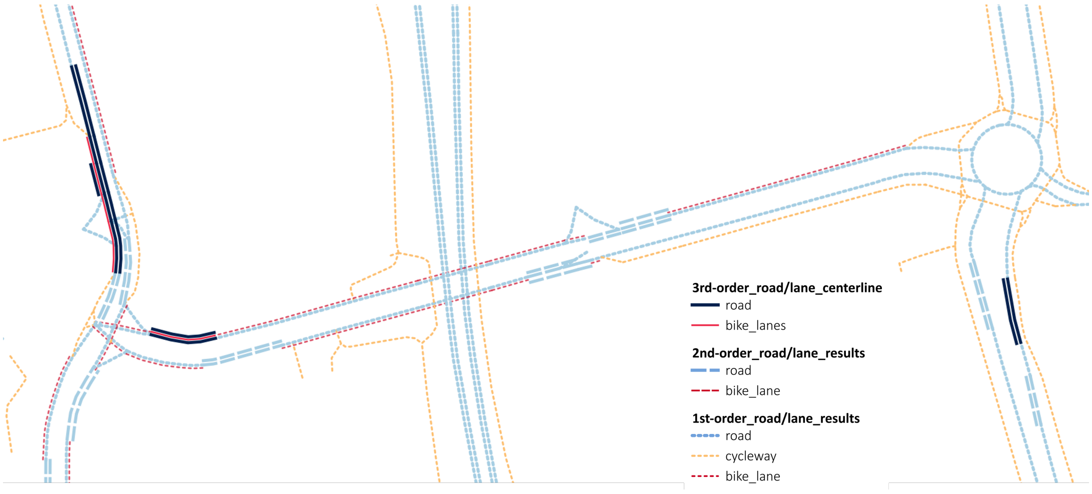


Figure 6.9.: Representative results of 3rd-order lanes generation, connecting with other lanes and converging to intersetions.

Figure 6.10.: Fourth-order and other order roads and generated lanes with accurate location.

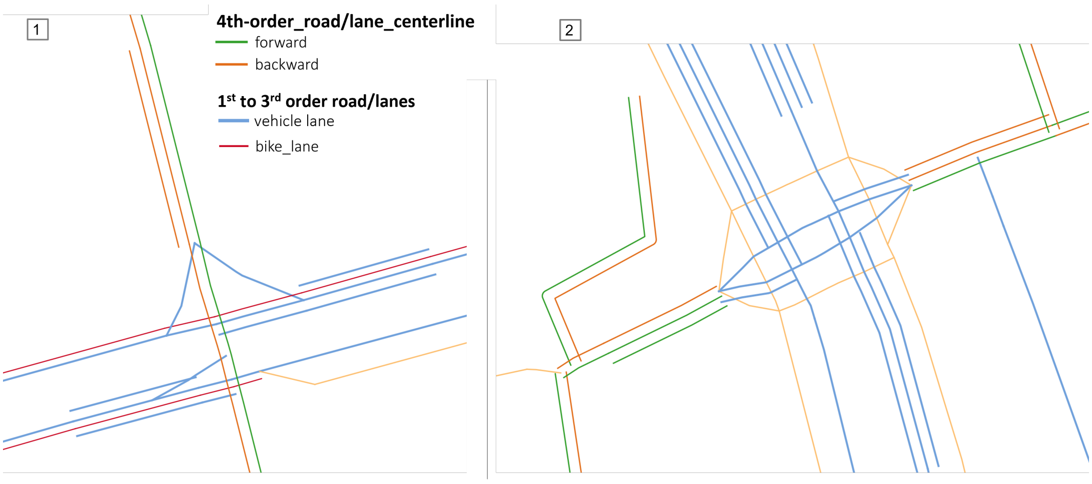


Figure 6.11.: Representative results of 4th-order lanes generation, connecting with other lanes and converging to intersetions.

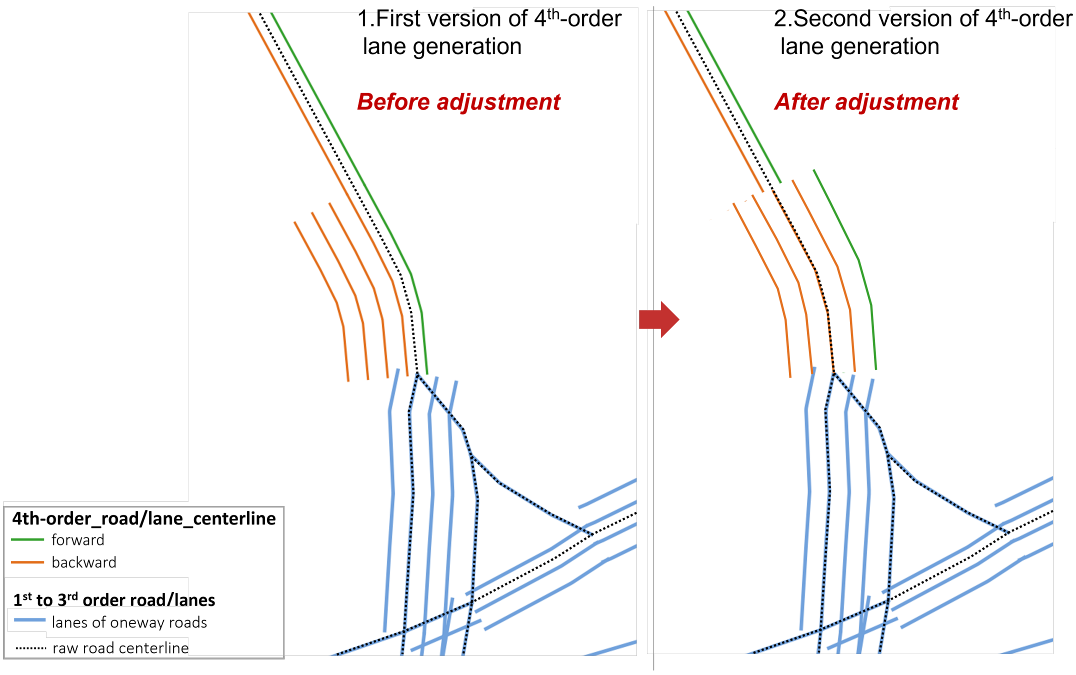


Figure 6.12.: Representative results of special cases regeneration, comparison of first generation result and the refined one.

4) fourth-order roads r_{h4} reconstruction The final order is for all the bidirectional roads and their lane generation. There are 107 two-way roads in this step in total, in which exist 4 roads with the uneven distribution of *'forward'* and *'backward'*, each of them connects to an intersection, the unevenly distributed lanes are the auxiliary lanes for turning. Finally, we generated 121 *'forward'* vehicle lanes and 125 *'backward'* lanes, also 6 bike lanes located on double sides of 3 two-way roads. Figure 6.10 shows the final result of all the lanes after the fourth-order lane generation complete. Figure 6.11 highlights the intricate scenarios as one-way roads connect to two-way roads and form the intersections.

5) special cases regeneration This step is for the regeneration of special cases, such as unevenly distributed bidirectional roads. As we stated in Section 4.4.1 as a general method, after the detection and lane amounts comparison (the L_{nw} of *forward* road r_w^{for} and *forward* road r_w^{back}), only one road need to be regenerated, the final results and the comparison as shown in Figure 6.12.

6.1.3. Results of lane network modification and correction

All the lanes generated from the previous phase need to be checked and modified to achieve the topological and geometric requirements.

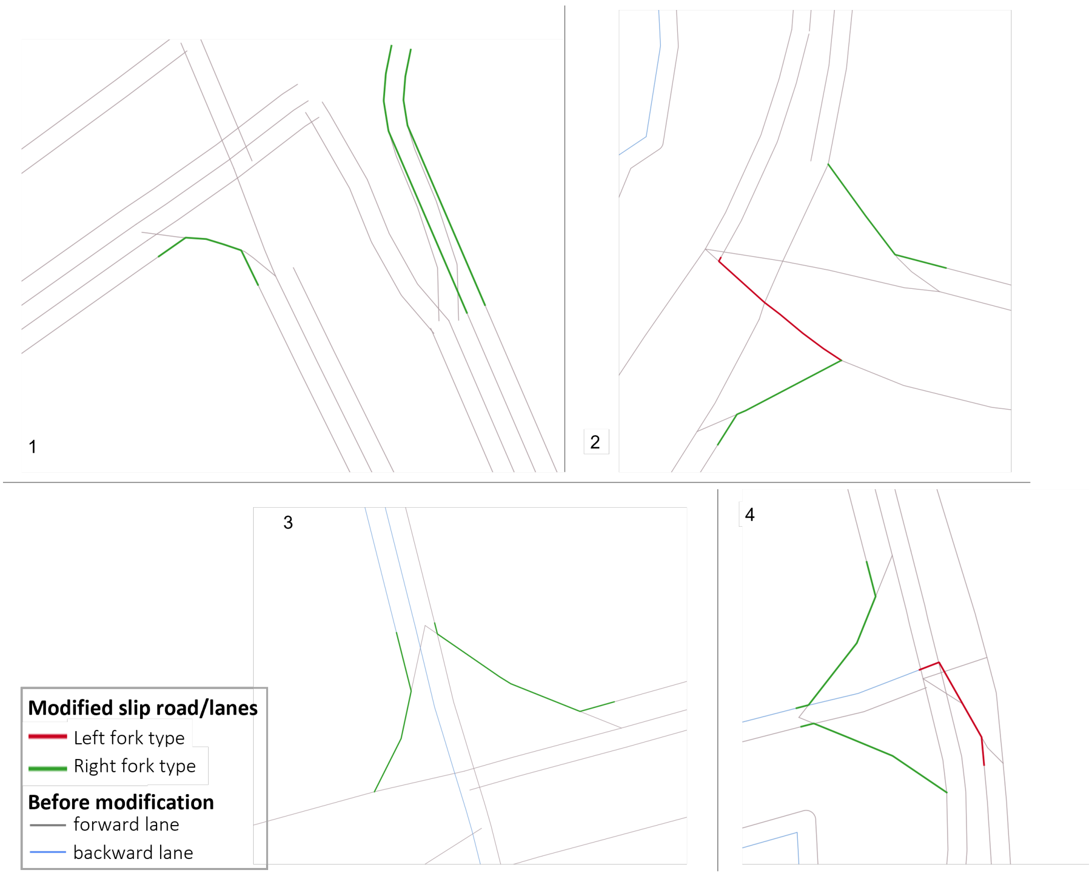


Figure 6.13.: Results of lane modification for the 'one-way' slip roads.

1) Lane modification Most of the labelled slip roads need to be modified, to ensure that they connect to their correct adjacent lanes. In total:

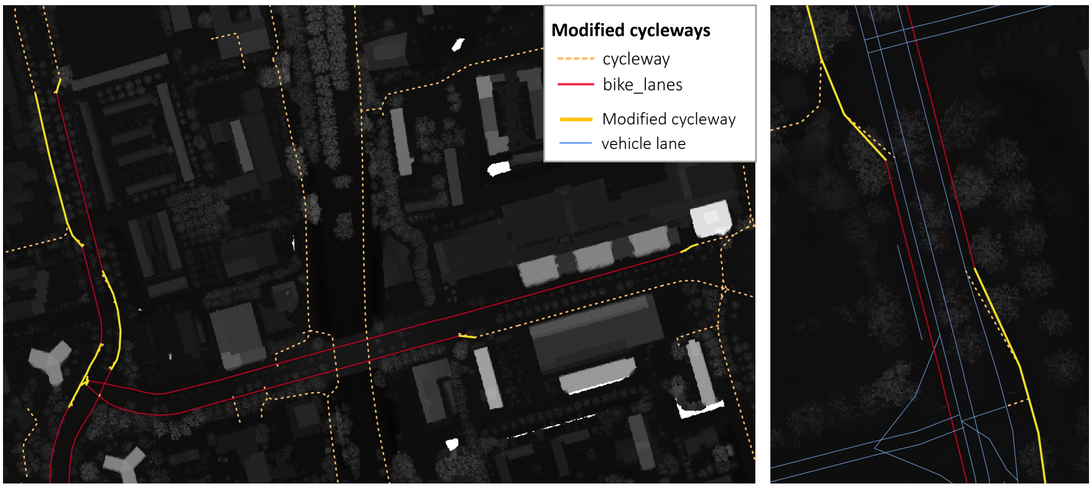


Figure 6.14.: Results of lane modification for the cycleways to align with bike lanes.

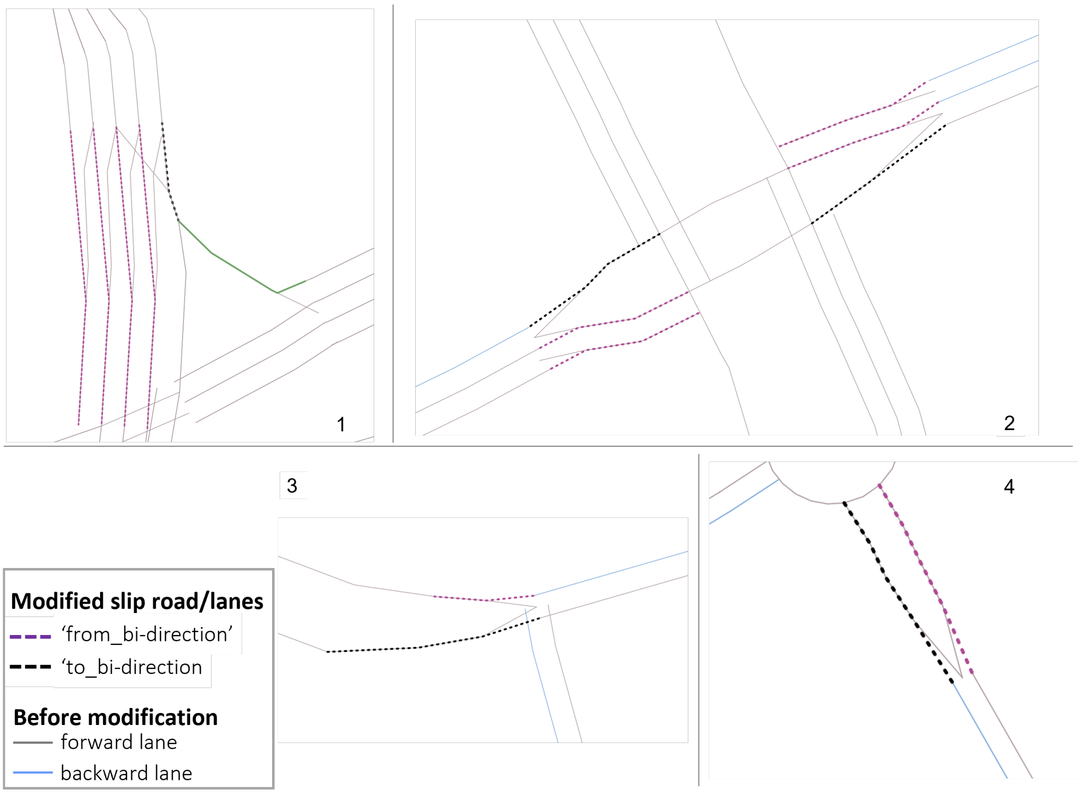


Figure 6.15.: Results of lane modification for the slip roads connect to 'two-way' roads.

Figure 6.16.: Results of lane modification for the auxiliary lanes to align with 'through' lanes.

- 26 lanes as the slip road are corrected. There are 12 lanes that generated from 'one-way' roads in raw data (see Figure 6.13), of which 2 lanes are '*left_fork*' type, 10 lanes are '*right_fork*' type, 4 lanes are modified twice (i.e., both of start and end vertice are modified to align with the correct predecessor and successor). And 9 lanes are '*from_bidirection*' type, 5 lanes are '*to_bidirection*' type, and they belong to 5 labelled roads (see Figure 6.15).

- 8 lanes as the separated cycleway are corrected, 2 lanes among them are modified twice (see Figure 6.14).
- 50 lanes as the auxiliary lane are corrected, of which 5 lanes among them are modified twice, and 7 lanes are extracted from two-way roads in raw data (see Figure 6.16).

2) Lane correction: Topology aspect To validate and correct the topology integrity, we found 2 roads that have topology errors, which means none of the relative offset distances of the road r_{tar} is aligned with the adjacent lanes (predecessor or successor). The first-generation lanes and the corrected lanes are shown in Figure 6.17, we corrected 4 lanes in total.

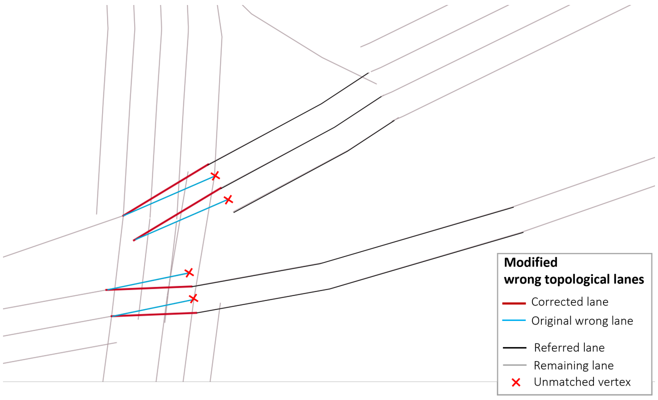


Figure 6.17.: Results of lane correction: Topology aspect.

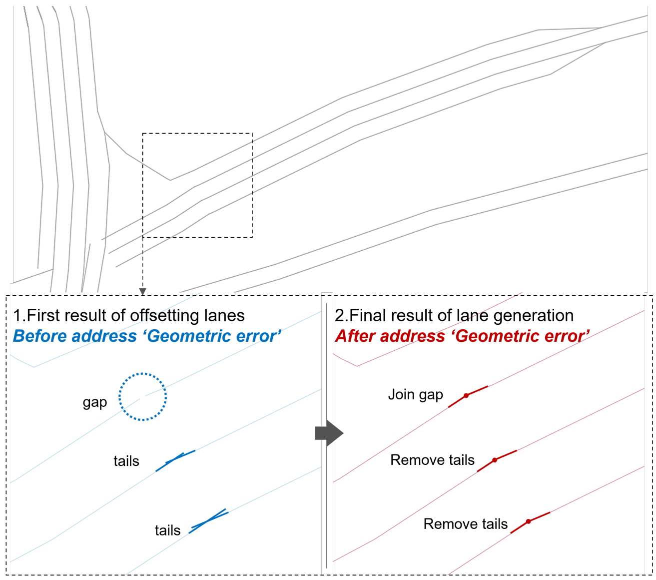


Figure 6.18.: Representative results of lane correction: Geometry aspect, join gaps and remove tails.

2) Lane correction: Geometry aspect All the lanes that have a **non-zero** relative offset distance FD_l encounter geometric errors since the general '**offset**' method we employed in the hierarchical generation phase. Figure 6.18 shows the typical example for comparison and the impact of this correction step.

Figure 6.19.: Final result of lane generation.

6.1.4. Results of final lane-level network

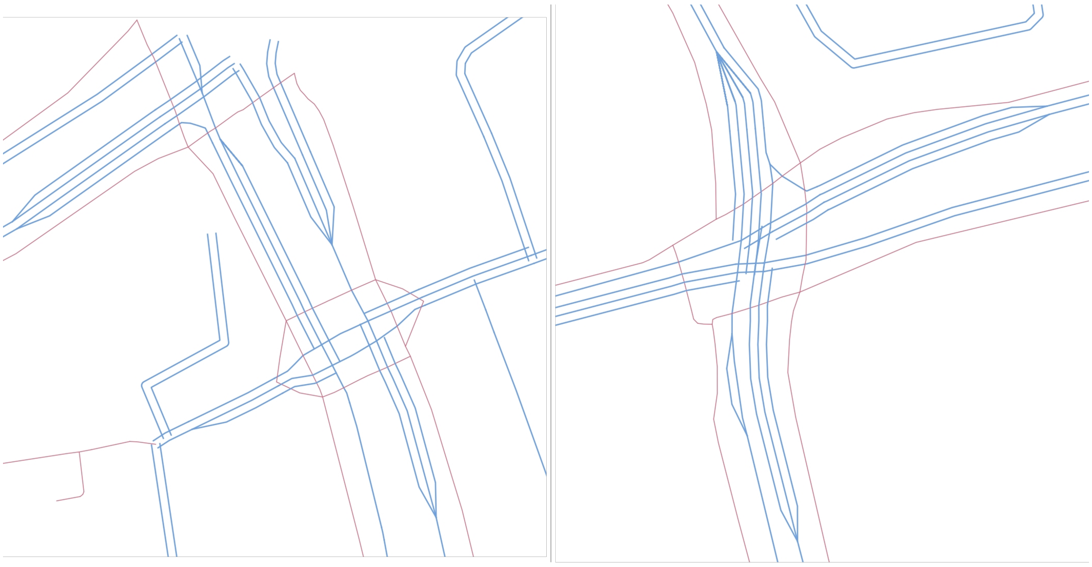


Figure 6.20.: Multiplex scenario 1 of final result of lane generation.

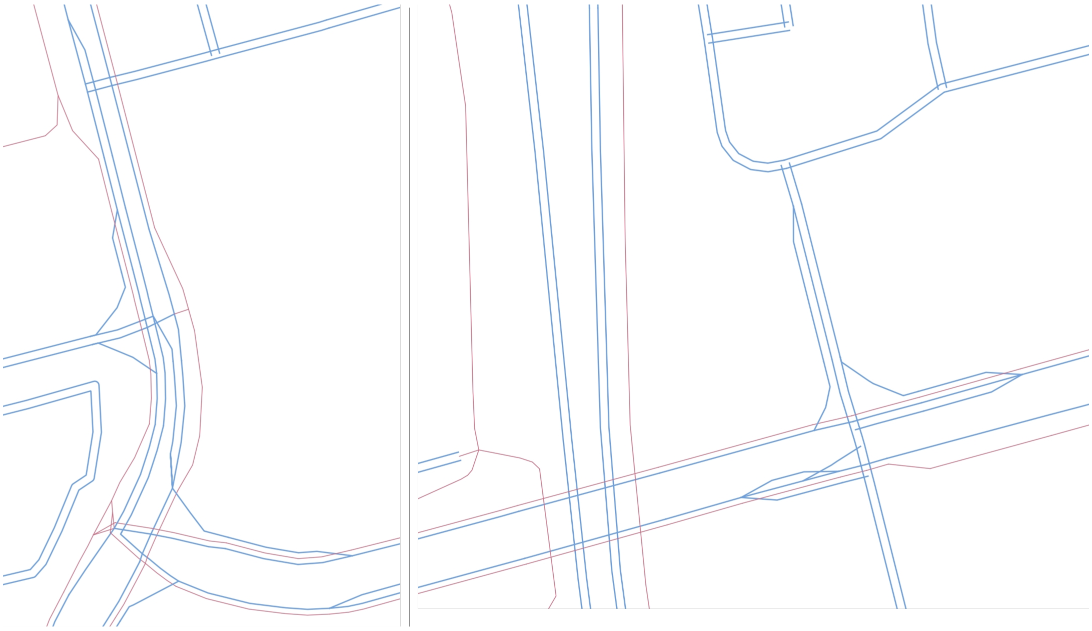


Figure 6.21.: Multiplex scenario 2 of final result of lane generation.

The final result of the "**Lane Generation**" phase is shown in Figure 6.19. It contains:

- Normal roads/lanes and transition parts of roads, normal lanes and auxiliary lanes.

- Intersections:
  - Single directional and type intersection;
  - Mixed type intersection: vehicle and cycleway;
  - Mixed direction : from_bi, to_bi;
  - Mixed direction and type intersection;
  - Cycleway-bike lane condition;
  - Unconnected bidirectional roads;
  - Single and bidirectional roads and not ensure the topology connection.

6.2. Results of 2D areal representation

Using the lane polygons, we can get two dissolved results, see Figure 6.22:

- Dissolved by road indices;
- Dissolved by road traffic modes (semantic information).

Figure 6.23 is the example for illustrating the typical result of general road polygon generation based on the multiple times of lane centerline modification. To achieve the ideal lane geometry for areal representation we conducted implementation for special cases (as introduced in Section 4.4.1) and auxiliary lanes.

6.2.1. Road polygons of general method

As we introduced in Section 4.5.2, the key point of this post-processing phase is the **node degree-based** method. Thus, node degree N_k correction is one of the important results of this phase.

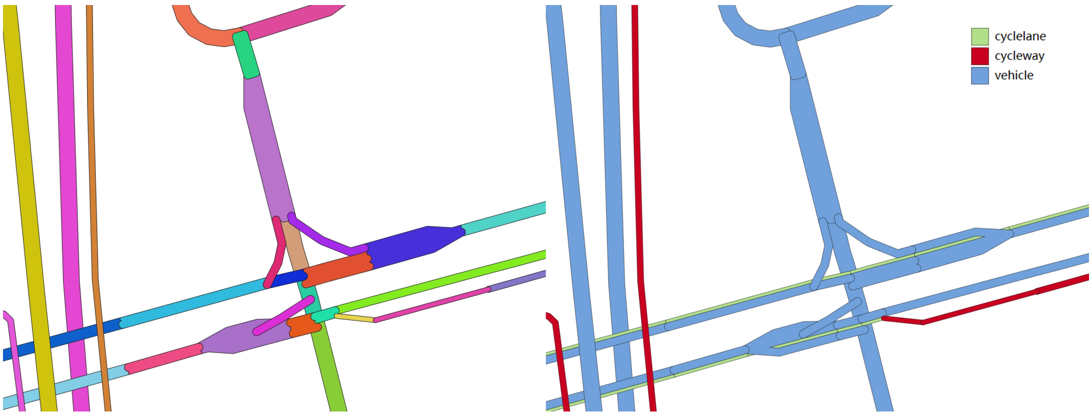


Figure 6.22.: Two dissolved results: left is merged by road indices, right is merged by the lane types.

1) Result of node degree correction

The main goal of the correct "*true node degree*" is to identify the road centerlines which is not interrupted in the intersection nodes. Note that, the "*true node degree*" here is the total number of joint segments, including all types of roads, thus, the division of different types, and the results of distinguishing the vehicle roads and cycleways are not shown here.

2) Result of road polygon post-processing

- Non-overlapping roads: we trimmed 227 road polygons to remove the overlapping parts, based on the joint node degree of same traffic type is 2; of which 148 motorized roads, 59 cycleways, see Figure 6.25.
- Polygon errors fixed: 84 road polygons are fixed or modified since the error detection; of which 7 nodes of roads are found and road polygons are trimmed (Figure 6.26) to avoid errors because of the different road widths, caused by the different road types in the intersection, or the roads with different numbers of lanes converged, representative cases as shown in Figure 6.27.

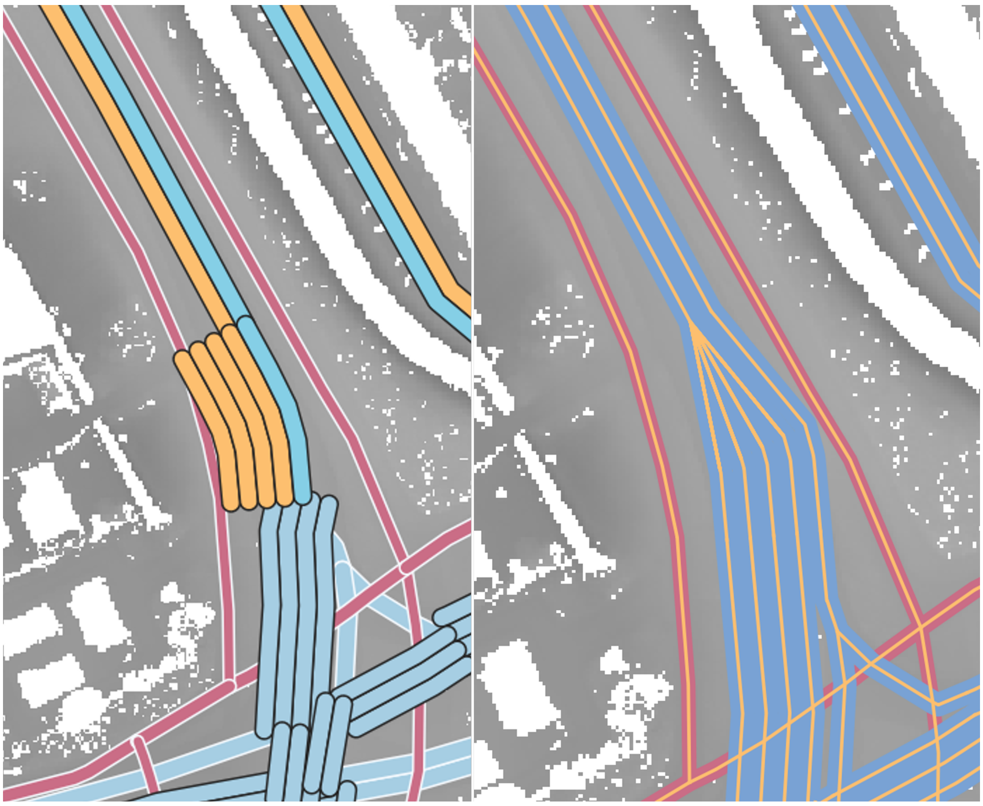


Figure 6.23.: Road polygon generation for special case: dual-carriageways with slip roads

6.2.2. Results of post-processing

Figure 6.24.: The corrected nodes and their true node degree as N_k for each intersection.

Figure 6.25.: Results of all trimmed polygons to remove the overlapping parts as the adjacent pairs.

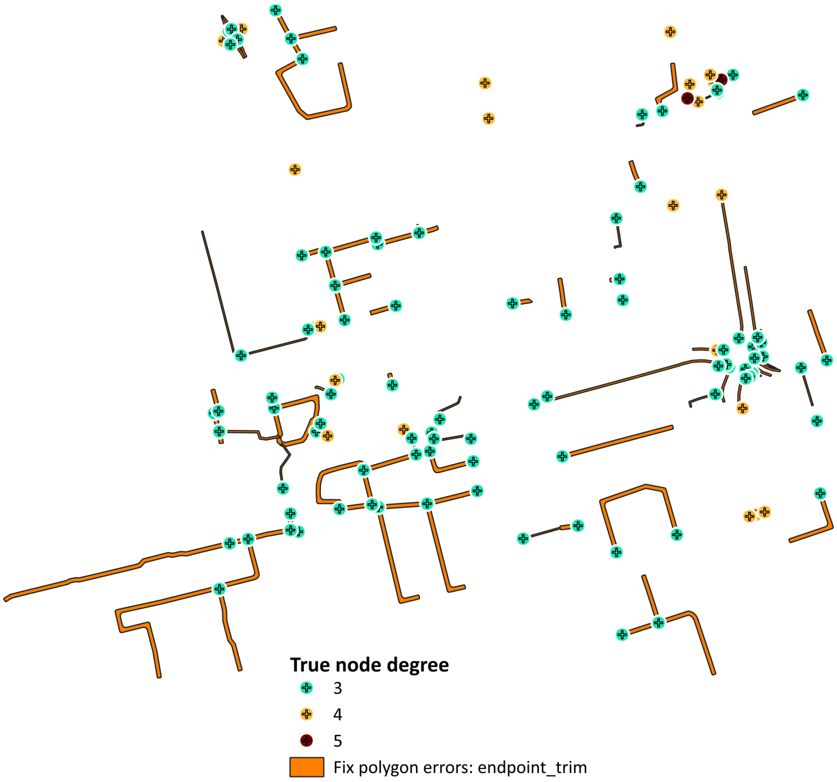


Figure 6.26.: Results of all trimmed polygons to avoid the errors at endpoints.

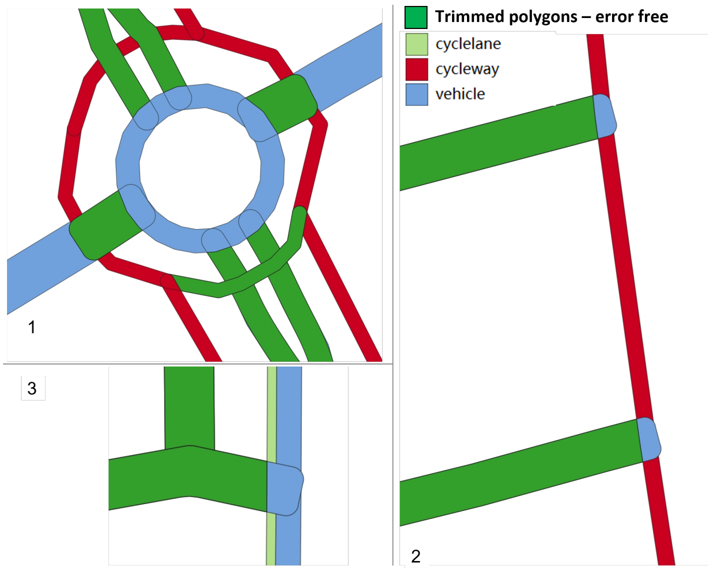


Figure 6.27.: Examples of how to trim endpoint of the polygon, the dark green polygons are the correct results.

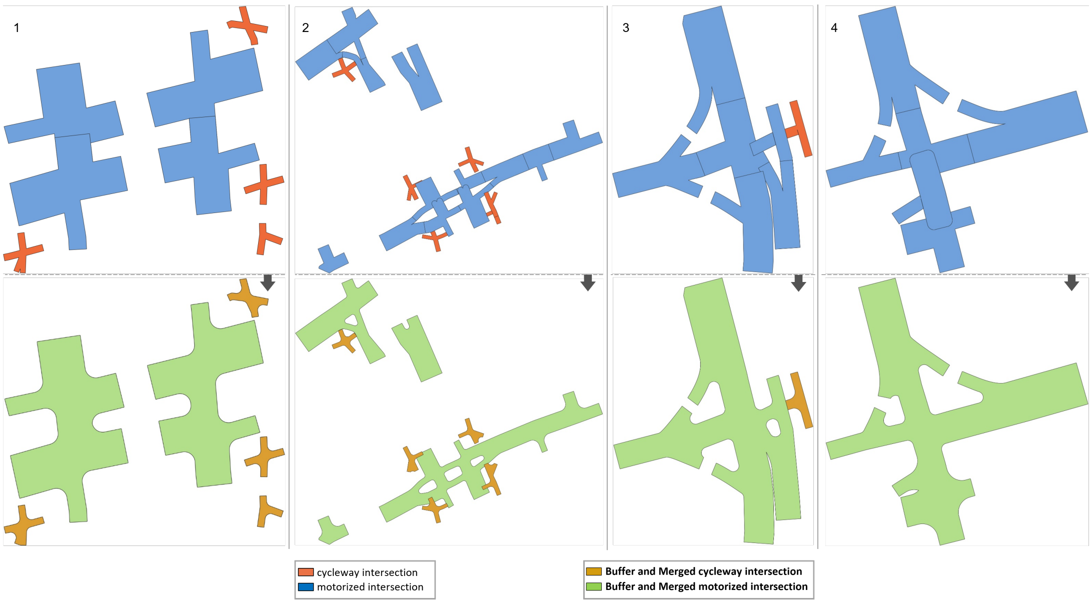


Figure 6.28.: Results of typical intersection geometry generation.

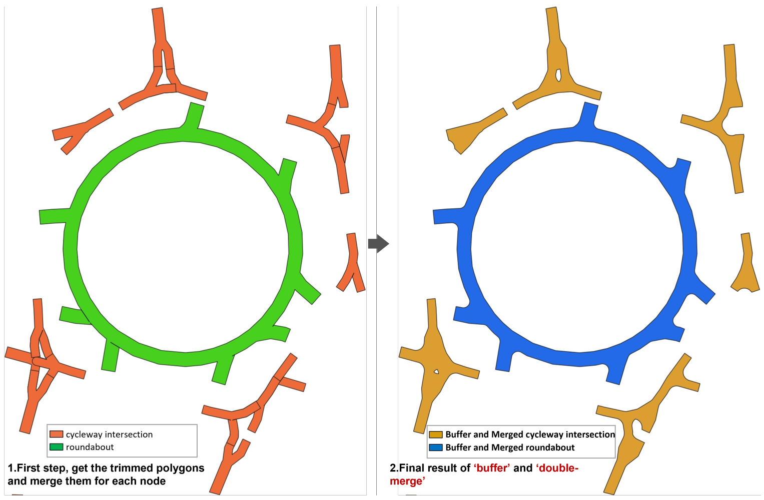


Figure 6.29.: Results of roundabout geometry generation.

3) Result of intersection polygon generation

After applying three steps: trim converged polygons, merge, and buffer, we generated the final intersections, 162 in total. Since during the process, we considered the semantics, the traffic types of converged roads in one intersection, we can further divide intersections as 109 'motorized intersection', 2 roundabouts, and 49 'cycleway intersection'. According to their shapes, the common types of intersections are:

- T/Y intersection: motorized and cycleway intersection;

- cross intersection: motorized and cycleway intersection;
- three ramps intersection;
- asterisk-shape intersection.

Figure 6.30.: Results of all intersection polygons without overlapping.

As some intersections are close to others, and they form a complex intersection area in the urban traffic environment, thus, traversing all the intersections, we detected the overlapping intersections as the ’multiplex intersection’ and merged them again to create a complete polygon to represent the area, see Figure 6.28 and 6.29. Finally, after ’double-merge’ implementation, the number of intersections is 108, of which 66 motorized intersections, 2 roundabouts, and 40 cycleway intersections (Figure 6.30).

3) Result of intersection polygon generation

After applying three steps: trim converged polygons, merge, and buffer, we generated the final intersections, 162 in total. Since during the process, we considered the semantics, the traffic types of converged roads in one intersection, we can further divide intersections as 109 ’motorized intersection’, 2 roundabouts, and 49 ’cycleway intersection’. According to their shapes, the common types of intersections are:

- T/Y intersection: motorized and cycleway intersection;
- cross intersection: motorized and cycleway intersection;
- three ramps intersection;
- asterisk-shape intersection.

As some intersections are close to others, and they form a complex intersection area in the urban traffic environment, thus, traversing all the intersections, we detected the overlapping intersections as the ’multiplex intersection’ and merged them again to create a complete polygon to represent the area, see Figure 6.28 and 6.29. Finally, after ’double-merge’ implementation, the number of intersections is 108, of which 66 motorized intersections, 2 roundabouts, and 40 cycleway intersections (Figure 6.30).

7. Conclusion and Future Work

In this final chapter, the research questions of this graduation project we defined in Section 1.3 are reviewed in order to assess the degree in which they have been finally addressed. Based on this, our contributions to the current road network modelling and reconstruction are presented, along with the limitations of our proposed methodology. With respect to these limitations, some future work is also recommended.

7.1. Research Overview

To answer our main research question: "***"How can we achieve a comprehensive areal representation of road network model with topological and geometric correctness, and enrich its semantic information by only using the open geospatial datasets?"***", it is necessary to address each subquestion in detail. The conclusions drawn from answering these subquestions will collectively form the complete solution to the overarching research problem. Each subquestion explores a specific aspect of the methodology, contributing to the overall goal of generating a precise and semantically rich road network model.

Questions:

- **Subquestion 1:** ***How can the information and attributes contained in OSM road data be utilized to generate and optimize 2D linear and areal representations of road network models?***

Answer:

To answer the first subquestion, we developed a comprehensive methodology for OSM data pre-processing, as outlined in Section 4.3.2. This methodology builds upon the background knowledge of OSM road data structure and tag interpretation provided in Section 2.3, particularly in Section 2.3.2, which summarizes the strengths, limitations, and uncertainties of OSM road data representations.

Our approach includes a filtering process to extract essential and valid information from a limited number of OSM tags, which was designed to avoid the challenges of dealing with numerous tags and to ensure the methodology can be applied across various conditions. The decision on which tags to use was based on a thorough understanding of the background concepts discussed in Section 2, including road components, road types within traffic flows, and the importance of analyzing the "good continuity" of a road network. These factors, combined with key geometric design principles that impact lane arrangement, turning movements, and road shape variations, guided our selection.

We identified nine critical OSM tags for the model: *'highway=*'*, *'cycleway=*'*, *'name=*'*, *'oneway=*'*, *'lanes=*'*, *'turn=*'*, *'junction=*'*, *'bridge=*'*, *'tunnel=*'*. Additionally, suffixes such as *':forward'*, *':backward'*, *':left'* and *':right'* were carefully considered to address varying traffic conditions.

Phases/Tags	*highway*	*cycleway*	*name*	*oneway*	*lanes*	*turn*	*junction*	*bridge*	*tunnel*
Road-level (R)	x	x	-	x	x	x	x	x	x
Traffic-level (R_w)	x	x	x	x	x	-	-	-	-
Lane-level (R_l)	x	x	-	x	x	x	-	-	-
Grouped strokes (S)	x	x	x	x	-	-	-	-	-
Road polygon (RA)	x	x	-	x	x	-	-	-	-
Intersection polygon (C, CA)	x	x	-	-	x	-	x	x	x

Table 7.1.: Uasges of the tags and corresponding values.

After extracting these valid tags and their values, we initiated the construction of a road-level network, denoted as $\boldsymbol{W}$, which consists of road centerlines $\boldsymbol{R}$ and intersection nodes $\boldsymbol{C}$, to store both the geometry and attributes of the roads.

The information extracted through this process is frequently utilized as key inputs in subsequent steps to achieve the final goals of the research. Table 7.1 provides a detailed overview of how these tags and attributes are applied in each major stage of the methodology.

• **Subquestion 2:** ***Is it possible to enhance the resolution and LoDs of a road network model from the original road-level to a high-detailed lane-level network using only OSM data?***

Answer: **Yes.**

To enhance the resolution and LoDs of road network models, as outlined in Section 2.2.1, several key factors are critical for this improvement: the number of lanes, carriageways, and the traffic types of roads and their lanes. Following the extraction of valid attributes as described in the first subquestion, the essential data required for generating a lane-level network is now available. This provides a solid foundation for enhancing the network resolution.

To bridge the gap between the road-level and lane-level networks, we designed an intermediate stage known as the **traffic-level** network ($\boldsymbol{R}_w$). This network facilitates the transformation from a basic road-level structure to a more detailed lane-level configuration by incorporating traffic flow data and ensuring that each lane is accurately represented. Thus, the methodology achieves a higher detailed representation of the road network, fully leveraging OSM data.

• **Subquestion 3:** ***Can the topology relationships, lane-to-lane adjacency for representing traffic flow movements, be preserved through OSM data processing and detailed generation methods?***

Answer: **Yes.**

In Section 4.2.1, we introduced three levels of road network representation, focusing on resolving common issues related to topological errors. A key concept, **lane connectivity**, was introduced, emphasizing the need to maintain consistency between the overall road topology and the detailed lane topology. This principle guided our approach to lane-level network generation.

To preserve lane-to-lane adjacency and ensure topological correctness, the methodology was divided into several phases, starting with hierarchical generation and followed by lane connectivity corrections. Utilizing the *'highway'* attributes (H_r, H_w, and H_l), we distinguish between different traffic modes and directions of traffic flow, which are determined during OSM pre-processing and the generation of the traffic-level network.

The first phase of lane generation identifies reliable lanes that serve as references for building the topology structure. By establishing both global and local adjacency (grouped strokes), the method accurately maintains lane-to-lane relationships. Several strategies were developed to ensure correct lane adjacency:

1. Separate generation and modification of vehicle lanes and cycling lanes to prioritize H_l consistency and achieve proper lane matching.

2. Slip road labels (F_r) were used to identify turning lanes, ensuring topological correctness through proper connections.

3. Turn attributes (T_r and T_l) helped to determine the target lanes for slip road modifications.

4. Finally, comparing T_l values and relative offset distances (FD_l) allowed auxiliary lanes to find their corresponding 'through' lanes, completing the wedge-shaped modification of lane geometry.

These steps collectively ensure that the lane-to-lane adjacency is preserved throughout the network, providing a reliable and accurate representation of traffic flow within the model.

• **Subquestion 4:** ***How can we generate 2D road polygons as errorless geometries to accurately capture road shape details, including changes in road width and intersection variations?***

Answer:

The process of generating 2D road polygons involves several key steps aimed at eliminating errors while accurately capturing road shape details. First, the modifications made during the lane-level network generation help prevent errors from the linear representation from propagating to the areal representation. By addressing these issues early, the model retains its accuracy as it transitions to areal geometry.

For the remaining errors inherent in the raw OSM road centerlines, we developed a **node degree-based** method to ensure correct road connections. The true node degree plays a critical role in identifying intersections. We also utilized the *'highway'* (H_r) and *'lanes'* (L_n) attributes to detect potential errors at intersections, such as protruding or overlapping parts that could distort the road geometry.

Once errors related to intersections were identified, we applied geometric processing techniques to resolve issues like overlapping, protruding, and inconsistent road divisions. This method systematically addressed each error, ensuring that normal road ribbons were corrected before proceeding further.

For intersections, we developed a robust method that works for all types of intersection shapes, producing errorless intersection polygons. The key strengths of this approach include:

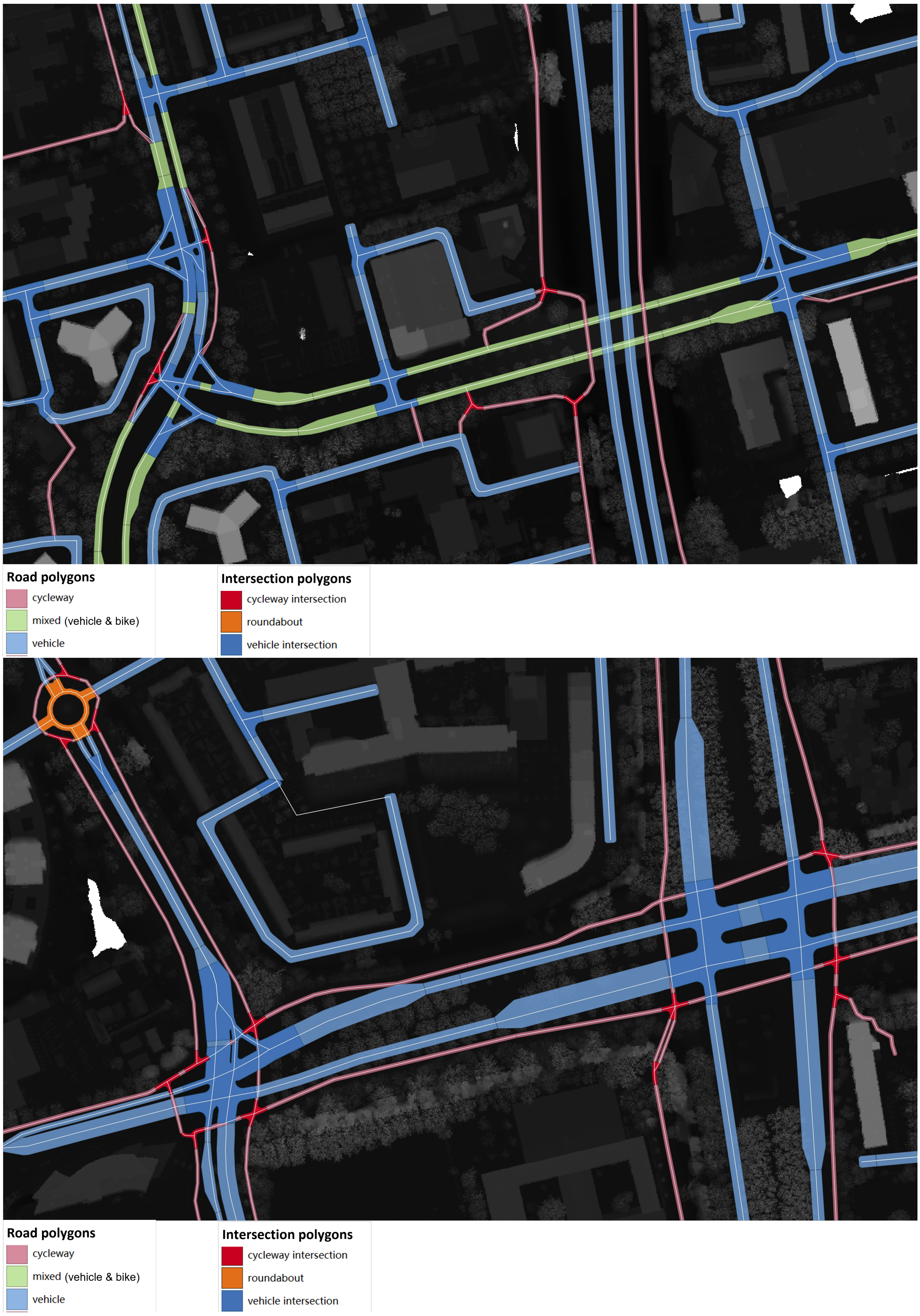


Figure 7.1.: From OSM road centerlines to semantically enriched road polygons as areal representation.

1. Resolving complex overlapping issues by merging trimmed polygons of converging roads, ensuring a smooth, cohesive shape.
2. Incorporating turning corner curvatures into intersection polygons, which optimizes the areal representation and follows transportation geometric design principles.
3. Addressing lane connectivity errors—especially those caused by missing road information—by generating compact, reliable intersection polygons that reflect real-world road areas (see Figure 7.2).

This comprehensive approach ensures that the final road polygons are error-free and accurately represent road width variations and intersection complexity (Figure 7.1 shows the reconstruction results, the white polylines as the input: OSM road centerlines, and the polygons of road ribbons and intersections are the areal representation as outcomes).

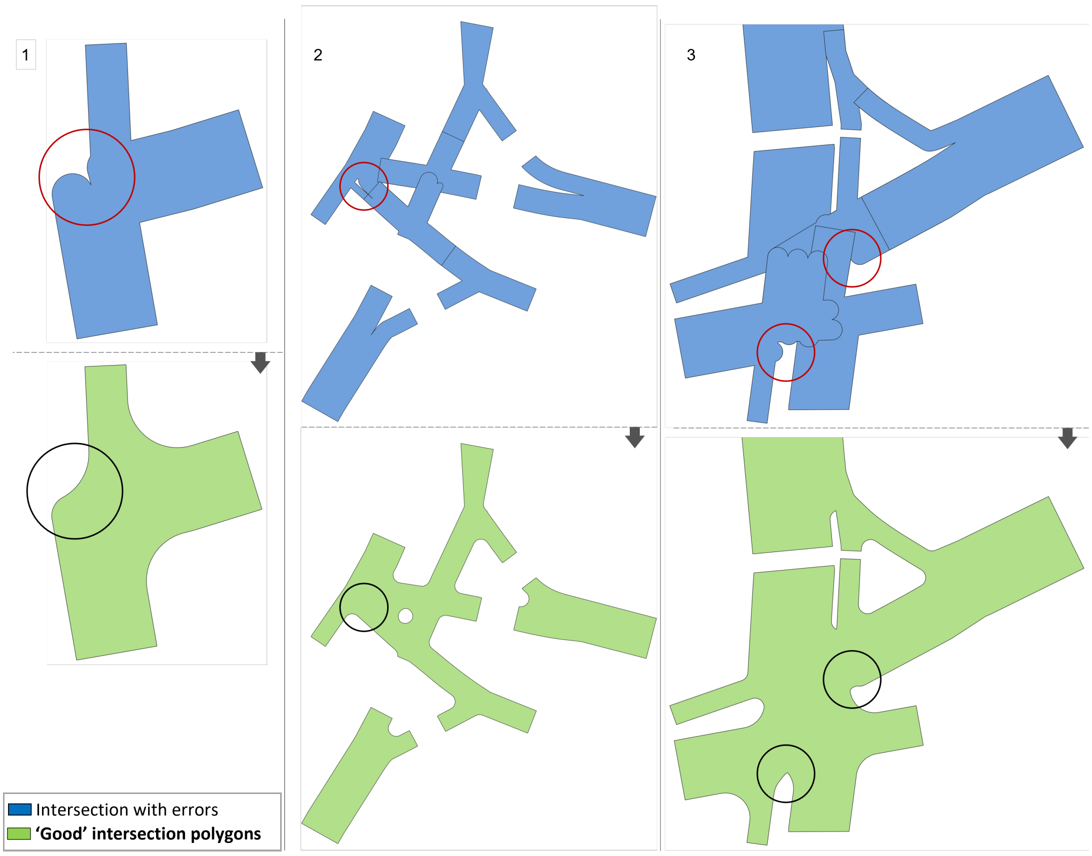


Figure 7.2.: Intersection generation method helps to fix the errors of lane geometry.

7.2. Impacts and Contributions

Throughout this project and thesis, several concepts have been introduced as part of our methodology. While these ideas are not entirely novel and can be traced back to related work, we believe our approach offers valuable contributions to the current state of the art. The most significant contributions are outlined below:

- **Utilization of OSM Road Data:** Unlike previous studies, as discussed in Chapter 3, which often rely on data fusion from multiple datasets, our methodology utilizes only OSM data to generate high-detailed lane-level linear and areal representations of road networks. Other studies have demonstrated that road network generation and model reconstruction are challenging tasks that typically require supportive datasets. By relying solely on OSM data, we explore its full potential, expanding the possible use cases for similar tasks. This comprehensive examination of encoded OSM road data is a fundamental contribution that opens up new possibilities for the application of open-source geospatial data.
- **Redefining the Concept of the Road Centerline and road *center*:** A close analysis of OSM road modelling reveals uncertainties in the relationship between the road centerline, road polygon, and road width. This insight prompted us to challenge the traditional understanding of the "road centerline." Instead, we focus on using the curvature of the road centerline to represent traffic flow, reflecting the movement of all affiliated lanes. While this curvature provides reliable data, our methodology emphasizes generating a more accurate lane layout based on a comprehensive view of the road. This redefinition of the OSM road centerline serves as the cornerstone of our methodology. The validity of this new approach is proven through our results, with room for further experimentation and comparison to traditional methods.
- **Integration of Road Network Reconstruction and Related Concepts:** Our methodology integrates three key areas of study: 1) Graph theory for modelling the topology of road networks at different levels, using concepts such as node degree, digraph, multidigraph, predecessors, and successors in our process. 2) Strokes theory for analyzing the continuity of roads. We extend this from the road level to the lane level, which is critical for lane topology. 3) Geometric design principles from transportation engineering, particularly for the high-detailed representation of lane geometry, such as auxiliary lanes. These interdisciplinary integrations significantly enhance the method design and implementation, providing both supplementary conditions and constraints that anchor key points in our process. This approach not only strengthens our methodology but also offers a foundation for future work that draws from diverse fields to solve complex problems.
- **Lane Connectivity with Semantic Enrichment:** One of our key contributions is maintaining the consistency between road-level, lane-level, and areal-level topologies. We treat the raw road-level topology as the "ground truth," ensuring that lane connectivity, enriched with semantic data, aligns with real-world conditions. Many studies on lane-level digital maps focus solely on vehicle navigation systems. However, our approach incorporates semantic information for different traffic modes, ensuring that not only vehicle lanes but also cycling lanes and other modes of transport are accurately represented. This mixed-use approach allows for detailed scenarios, such as the priority given to cycling lanes situated between vehicle lanes. In regions like the Netherlands, our method accurately reflects the road space distribution, supporting advanced studies like bicycle traffic safety analysis and contributing to detailed 3D city models.
- **Detailed Areal Processing and Representation:** Traditional methods for road boundary extraction typically focus on external boundaries. Our areal representation, how-

ever, captures detailed characteristics, including lane polygons, road polygons, and intersections. For example, the representation of continuous traffic lanes or cycling routes can be used for urban management, transportation modelling, and visualization. This level of detail offers higher granularity in applications, enabling more precise analysis and decision-making in urban planning and transportation management.

- **Integration of Topology, Geometry, and Semantics:** Our methodology seamlessly combines topology, geometry, and semantics from OSM data to optimize the final result. For instance, the labelling of slip roads integrates both geometric and semantic elements, which further supports lane-level topology correction. Similarly, our approach to generating intersections begins with addressing road-level topology, detecting geometric errors, and ultimately ensuring that topological correctness leads to accurate, detailed geometry. This integrative approach highlights the importance of fully utilizing available data to create solutions, providing a foundation for future research in this field.

7.3. Limitations

While our methodology contributes to the current state of the art, there are certain limitations that we would like to address. These limitations become apparent when our lane-level network generation algorithms depend on accurate road centerline data or encounter issues that prevent the generation of a correct model. Recognizing these limitations allows us to recommend future improvements to enhance the viability of our framework.

7.3.1. Dependency on OSM Data Quality

Since our methodology relies on several OSM tags and attributes for processing, it is highly dependent on the quality and completeness of the OSM road data. Missing or incorrect attributes can lead to propagation errors that affect the final output, as these issues remain undetected in the absence of additional data sources. Because we use only OSM as the input, without data fusion from other sources, errors inherent in the OSM data cannot be corrected or verified. In regions with well-mapped road networks, the primary limitation lies in the accuracy and quality of the data. However, in areas with incomplete road data (e.g., missing tags), our methodology may struggle to reconstruct missing lanes and then cannot achieve the goal of a high-detailed model; or process incomplete semantic attributes (e.g., the absence of *'name=*'*), this can result in inaccuracies in stroke grouping and subsequent processing steps, particularly when critical lane information is unavailable.

7.3.2. No optimization for the raw road centerlines

In the real world, road geometries typically exhibit smooth curves and meet the condition of *C1 continuity*, meaning the curves are continuous in both position and tangent direction when two roads join. This ensures a smooth, non-abrupt transition at intersections and along the road. However, due to data simplification in the OSM data, curves are often represented as a series of straight segments, which breaks the C1 continuity. These segmented representations cause the road and lane geometries to be composed of jagged lines rather than smooth curves.

In our methodology, we directly utilize the raw centerline geometry without applying any optimization to restore *C1 continuity*. This decision leads to geometries that do not meet the smoothness conditions of real-world roads, and thus, the lane-level networks and the road polygons as areal representation inherit this lack of continuity.

7.3.3. Efficiency Constraints

The aim of our study was not to develop a fully automated road network reconstruction approach. Due to the number of manual adjustments required for optimization and correction, our methodology follows a procedural workflow that must be applied in sequence. The more detailed the reconstruction and the more errors to be addressed, the more steps are required, which may affect the overall processing efficiency. While automation of the process could enhance efficiency, it falls outside the scope of our current work.

7.4. Future work

Based on the limitations presented in Section 7.3, we would like to recommend some ideas for future work. These recommendations aim to further refine our proposed methodology and expand its applicability to a broader range of contexts. The following suggestions highlight key areas for enhancement:

7.4.1. Test more datasets for more regions

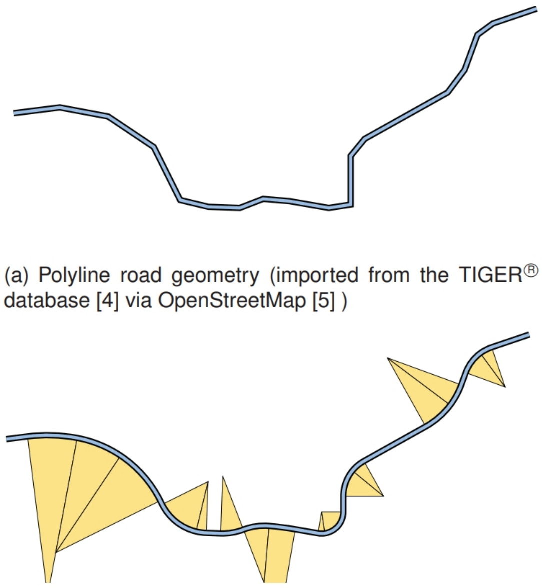
One of the primary limitations of our methodology is its reliance on the quality and completeness of OSM data, which can vary greatly between regions. Future work should test the methodology across more diverse geographic areas, including regions with different levels of OSM data completeness. This would help identify region-specific challenges and determine how robust the current framework is when applied to areas with sparse or incomplete data. Additionally, testing in regions where multiple road types and traffic conditions are prevalent could offer insights into improving the adaptability of the methodology to different road networks and mapping conditions.

7.4.2. C1 continuous curve optimization

As discussed in the limitations, the road and lane geometries generated by our method do not currently satisfy *C1 continuity*, leading to jagged or segmented curves rather than smooth transitions between roads. Future work could focus on incorporating techniques to optimize the road centerlines to restore *C1 continuity* before lane generation (Fig. 7.3 Wilkie et al. [2012]). One potential approach could involve applying curve-fitting algorithms to approximate the original, smooth road geometry from the segmented data. This would ensure more realistic and smooth road representations, particularly important for intersections and curved road segments, where abrupt changes can misrepresent actual traffic flows. Such an enhancement would contribute to more accurate and visually consistent lane-level models.

7.4.3. Automatic reconstruction

While our methodology is procedural, requiring manual adjustments to optimize and correct results, an important area for future research is automating the reconstruction process. Developing an automated workflow for generating lane-level networks and correcting topological or geometric errors could significantly improve the efficiency of the methodology. This might involve machine learning techniques for pattern recognition in road networks or advanced algorithms that can automatically detect and resolve common errors, such as missing lanes or incorrect connections. Automation would make the method more scalable and easier to apply to large datasets or regions with varying complexities, making it more suitable for applications such as city-wide digital twins or automated mapping systems for autonomous driving technologies.



(a) Polyline road geometry (imported from the TIGER® database [4] via OpenStreetMap [5])

(b) An *arc road* derived from the above polyline. The orange arcs show the center and radius of each arc used to give the road its smooth appearance.

Figure 7.3.: Smooth the C0 road polyline to C1 continuous curve. Wilkie et al. [2012]

A. Reproducibility self-assessment

A.1. Marks, and Self-reflection

Input Data: Score: [1]

The research data used in this study is exclusively obtained from the openly accessible and well-documented OpenStreetMap, which does not require a license, thereby greatly enhancing the reproducibility of the findings. However, as the data itself is not provided directly with the thesis, users must independently access OSM data (for example, Overpass API) using the documented procedures. This approach ensures compliance with OSM licensing but may require additional time for replication.

Preprocessing Score: [1]

The preprocessing steps are thoroughly documented, including details on workflows, software versions, and dependencies.

Methods, analysis, processing Score: [1]

Methods are described with sufficient detail, and this thesis provides pseudo-code and equations used for processing (covers most of the important algorithms used in the methodology). At the moment of finishing the thesis, the complete pipeline is not wrapped up and uploaded to a public and open repository, thus the availability scores are low.

Computational Environment Score: [3]

The computational environment (Python, required libraries, specific dependencies) is specified, and the thesis provides instructions for setting up a similar environment to facilitate reproducibility, all the required tools and libraries are documented in Chapter 5.

Results Score: [2]

The results include documented data outputs, tables, and visualizations necessary to understand and verify the findings. Additional statistical summaries and output data examples are available to illustrate the results. However, due to the time limitation, a series of output data (lane-level network model results as linear representation, road-level model as areal representation) are not available in a public repository yet.

Due to the time limitation and the complexity of the whole processing pipeline, making all the code, algorithms, and output datasets public requires a huge amount of effort. Thus, reorganize and publicize the methods and data will be a part of future work.

Bibliography

3D City model of Zurich. 3D Stadtmodell Zürich. URL https://www.stadt-zuerich.ch/content/ted/de/index/geoz/plan-und-datenbezug/3d-stadtmodell.html. Accessed 16 Oct. 2024.

AASHTO. *American Association of State Highway and Transportation Officials: A Policy on Geometric Design of Highways and Streets*. American Association of State Highway and Transportation Officials, Washington, D.C., 2011.

A/B Street. A/B Street GitHub Repository. https://github.com/a-b-street/abstreet.

A/B Street-Geometry. Map Geometry. URL https://a-b-street.github.io/docs/tech/map/geometry/index.html. Accessed: 2024-09-17.

A/B Street/ osm2streets. osm2streets GitHub Repository. https://github.com/a-b-street/osm2streets?tab=readme-ov-file.

At-grade. At-grade intersection. URL https://omni.wikiwand.com/en/articles/At-grade_intersection. Accessed: 2024-09-17.

C. Barrington-Leigh and A. Millard-Ball. The world's user-generated road map is more than 80% complete. *PloS One*, 12(8):e0180698, 2017. doi: 10.1371/journal.pone.0180698.

C. Beil and T. H. Kolbe. CityGML and the Streets of New York - A Proposal for Detailed Street Space Modelling. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-4/W5:9–16, 2017. doi: 10.5194/isprs-annals-iv-4-w5-9-2017.

C. Beil et al. Detailed Streetspace Modelling for Multiple Applications: Discussions on the Proposed CityGML 3.0 Transportation Model. *ISPRS International Journal of Geo-Information*, 9(10):603, 2020. doi: 10.3390/ijgi9100603.

S. A. Benz and R. Weibel. Road Network Selection for Medium Scales Using an Extended Stroke-Mesh Combination Algorithm. *Cartography and Geographic Information Science*, 41 (4):323–339, 2014. doi: 10.1080/15230406.2014.928482.

Bike lane. Wikiwand-bike lane. URL https://omni.wikiwand.com/en/articles/Bike_lane. Accessed: 2024-09-17.

F. Biljecki, J. Stoter, H. Ledoux, S. Zlatanova, and A. Cöltekin. Applications of 3D City Models: State of the Art Review. *ISPRS International Journal of Geo-Information*, 4(4):2842–2889, 2015.

F. Boersma. Modelling different levels of detail of roads and intersections in 3D city models. Master's thesis, Delft University of Technology, 2019.

bridge. Openstreetmap key:bridge. URL https://wiki.openstreetmap.org/wiki/Key:bridge. Accessed: 2024-09-17.

Bridge. Wikiwand-bridge. URL https://www.wikiwand.com/en/articles/Bridge. Accessed: 2024-09-17.

Carriageway. Wikiwand-carriageway. URL https://omni.wikiwand.com/en/articles/Carriageway. Accessed: 2024-09-17.

C. Chatzidiakos. Safety-Driven Road Width Estimations from Vector Data. Master's thesis, Delft University of Technology, 2021.

R. Cura, J. Perret, and N. Paparoditis. STREETGEN: In-base procedural-based road generation. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3/W5:409–416, 2015. doi: 10.5194/isprsannals-ii-3-w5-409-2015.

R. Cura, J. Perret, and N. Paparoditis. StreetGen: In-base city scale procedural generation of streets: road network, road surface and street objects. *arXiv preprint*, 2018. doi: 10.5194/isprsannals-ii-3-w5-409-2015.

cycleway. Openstreetmap: Bicycle. URL https://wiki.openstreetmap.org/wiki/Bicycle. Accessed: 2024-09-17.

Cycleway. Openstreetmap key:cycleway. URL https://wiki.openstreetmap.org/wiki/Key:cycleway. Accessed: 2024-09-17.

J. Dai, C. Li, Y. Zuo, and H. Ai. An OSM Data-Driven Method for Road-Positive Sample Creation. *Remote Sensing*, 12(21):3612, 2020. doi: 10.3390/rs12213612. URL https://doi.org/10.3390/rs12213612.

Degree distribution. Wikiwand-degree distribution. URL https://www.wikiwand.com/en/articles/Degree_distribution. Accessed: 2024-09-17.

T. Dey, J. Wang, and Y. Wang. Road Network Reconstruction from Satellite Images with Machine Learning Supported by Topological Methods. In *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 520–523, 2019.

Dual-carriageways. Wikiwand-dual carriageway. URL https://omni.wikiwand.com/en/articles/Dual_carriageway. Accessed: 2024-09-17.

L. Figueiredo and L. Amorim. Continuity lines: aggregating axial lines to predict vehicular movement patterns. In *Proceedings of 3rd Great Asian Streets Symposium*, Singapore, 2004.

L. Gao et al. Novel Framework for 3D Road Extraction Based on Airborne Lidar and High-Resolution Remote Sensing Imagery. *Remote Sensing*, 13(23):4766, 2021. doi: 10.3390/rs13234766.

A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012. doi: 10.1109/CVPR.2012.6248074.

G. Gröger, T. H. Kolbe, C. Nagel, and K.-H. Häfele. Open Geospatial Consortium (OGC) City Geography Markup Language (CityGML) Encoding Standard. Technical Report OGC 12-019, Open Geospatial Consortium (OGC), 2012. URL https://www.ogc.org/standards/citygml. Accessed: 2024-09-17.

X. Gu, M. Zhang, J. Lyu, and Q. Ge. Generating Urban Road Networks with Conditional Diffusion Models. *ISPRS International Journal of Geo-Information*, 13(6):203, 2024. doi: 10.3390/ijgi13060203. URL https://doi.org/10.3390/ijgi13060203.

Harlingen Car Crash Attorney. Turning properly at Texas intersections. URL https://harlingencarcrashattorney.com/blog/turning-properly-at-texas-intersections. Accessed: 2024-09-17.

HERE Technologies -Lane Model. Introduction to Mapping Concepts: Model Lane. URL https://www.here.com/docs/bundle/introduction-to-mapping-concepts-user-guide/page/topics/model-lane.html. Accessed: 2024-09-17.

highway. Openstreetmap key:highway. URL https://wiki.openstreetmap.org/wiki/Key:highway. Accessed: 2024-09-17.

N. Homayounfar et al. Dagmapper: Learning to Map by Discovering Lane Topology. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. doi: 10.1109/iccv.2019.00300.

Intersection. Wikiwand-intersection (road). URL https://www.wikiwand.com/en/articles/Intersection_(road). Accessed: 2024-09-17.

B. Jiang and C. Claramunt. Topological Analysis of Urban Street Networks. *Environment and Planning B*, 31:151–162, 2004.

K. Jiang et al. A Flexible Multi-Layer Map Model Designed for Lane-level Route Planning in Autonomous Vehicles. *Engineering*, 5(2):305–318, 2019. doi: 10.1016/j.eng.2018.11.032.

junction. Openstreetmap key:junction. URL https://wiki.openstreetmap.org/wiki/Key:junction. Accessed: 2024-09-17.

A. Kasmi et al. Map Matching and Lanes Number Estimation with OpenStreetMap. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018. doi: 10.1109/itsc.2018.8569840.

A. Keler, S. Amini, J. Lindner, and K. Bogenberger. Introducing Data-Format-Dependent Road Network Conversion Techniques – Lessons Learned from the Digital Twin Munich. In *Proceedings of the 31st Geographical Information Science Research UK (GISRUK) Conference*, 2023. doi: 10.5281/zenodo.7823414. URL https://doi.org/10.5281/zenodo.7823414.

K. Kenesei. Constructing a digital 3D road network for The Netherlands. Master's thesis, Delft University of Technology, 2021.

A. Labetski. A Framework for Application-Specific Generalisation of Buildings and Roads in 3D City Models. September 11 2017. URL https://3d.bk.tudelft.nl/alabetski/Publications/LabetskiPhDProposal.pdf.

Labetski, A. and van Gerwen, S. and Tamminga, G. and Ledoux, H. and Stoter, J. A Proposal for an Improved Transportation Model in CityGML. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-4/W10:89–96, 2018. doi: 10.5194/isprs-archives-xlii-4-w10-89-2018.

lanes. Openstreetmap key:lanes. URL https://wiki.openstreetmap.org/wiki/Key:lanes. Accessed: 2024-09-17.

M. Li et al. Construction of High-Precision DEMs for Urban Plots. *Annals of GIS*, 29(2): 193–203, 2023. doi: 10.1080/19475683.2023.2182360.

Z. Li and Q. Zhou. Integration of Linear and Areal Hierarchies for Continuous Multi-Scale Representation of Road Networks. *International Journal of Geographical Information Science*, 26(5):855–880, 2012. doi: 10.1080/13658816.2011.616861.

Off-ramp. Off-ramp cambridge dictionary. URL https://dictionary.cambridge.org/dictionary/english/off-ramp?topic=motorways. Accessed: 2024-09-17.

On-ramp. On-ramp cambridge dictionary. URL https://dictionary.cambridge.org/dictionary/english/on-ramp?topic=motorways. Accessed: 2024-09-17.

One-way street. Wiktionary one-way street. URL https://en.wiktionary.org/wiki/one-way_street. Accessed: 2024-09-17.

OpenStreetMap Berlin. Strassenraumkarte - About. https://strassenraumkarte.osm-berlin.org/about.

OSM-elements. Openstreetmap: Elements. URL https://wiki.openstreetmap.org/wiki/Elements. Accessed: 2024-09-17.

OSM-way. Openstreetmap: Way. URL https://wiki.openstreetmap.org/wiki/Way. Accessed: 2024-09-17.

Overpass. Wikiwand-overpass. URL https://omni.wikiwand.com/en/articles/Overpass. Accessed: 2024-09-17.

O. U. Press. *The New Shorter Oxford English Dictionary, Volume 2*. Clarendon Press, Oxford, 1993. Page 2632.

J. Pung, R. D'Souza, D. Ghosal, and M. Zhang. A Road Network Simplification Algorithm That Preserves Topological Properties. *arXiv*, 3(1):1–12, 2022. doi: 10.21203/rs.3.rs-2003220/v1. URL https://doi.org/10.21203/rs.3.rs-2003220/v1.

Ramp. Ramp-cambridge dictionary. URL https://dictionary.cambridge.org/dictionary/english/ramp?topic=motorways. Accessed: 2024-09-17.

F. Schilling and T. Kutzner. *Digital Road Networks: From Data to Representation*. Springer, 2020.

Swisstopo. SwissTLM3D - Swiss Topography Landscape Model. URL https://www.swisstopo.admin.ch/en/landscape-model-swisstlm3d. Accessed 16 Oct. 2024.

G. Tamminga and S. Hoogendoorn. *A Novel Design of the Transport Infrastructure for Traffic Simulation Models*. PhD thesis, Technische Universiteit Delft, 2019. Proefschrift ter verkrijging van de graad van doctor in het jaar.

G. J. Taylor. Roadway Geometric Design I: Functions, Controls and Alignments. URL https://www.cedengineering.com/userfiles/C05-023%20-%20Roadway%20Geometric%20Design%20I%20-%20Functions,%20Controls%20and%20Alignments%20-%20US.pdf.

S. Thom. A Strategy for Collapsing OS Integrated Transport Network(tm) dual carriageways. In *Proceedings of the 8th ICA Workshop on Generalisation and Multiple Representation*, A Coruña, 2005.

R. C. Thomson. The 'Stroke' Concept in Geographic Network Generalization and Analysis. In *Proceeding of 12th International Symposium on Spatial Data Handling*, Vienna, Austria, 2006.

R. C. Thomson and D. E. Richardson. The 'Good Continuation' Principle of Perceptual Organization applied to the Generalization of Road Networks. In *Proceedings of the ICA 19th International Cartographic Conference*, pages 1215–1223, Ottawa, Canada, 1999. ICA/ACI.

Thomson, R. and Richardson, D. A Graph Theory Approach to Road Network Generalisation. In *Proceedings of the International Cartographic Conference*, pages 1871–1880, 1995.

M. Tomko and S. Winter. Experiential hierarchies of streets. *Computers, Environment and Urban Systems*, 32(1):41–52, 2008.

P. Tripathy, P. Rao, K. Balakrishnan, and T. Malladi. An open-source tool to extract natural continuity and hierarchy of Urban Street Networks. *Environment and Planning B: Urban Analytics and City Science*, 48(8):2188–2205, 2020. doi: 10.1177/2399808320967680. URL https://doi.org/10.1177/2399808320967680.

R. J. Trudeau. *Introduction to Graph Theory*. Dover Publications, 1st edition, 1993. ISBN 9780486487328.

tunnel. Openstreetmap key:tunnel. URL https://wiki.openstreetmap.org/wiki/Key:tunnel. Accessed: 2024-09-17.

turn. Openstreetmap key:turn. URL https://wiki.openstreetmap.org/wiki/Key:turn. Accessed: 2024-09-17.

Turning lane. sabre-roads turning lane. URL https://www.sabre-roads.org.uk/wiki/index.php?title=Turning_Lane. Accessed: 2024-09-17.

Underpass. Wikiwand-underpass. URL https://www.wikiwand.com/en/articles/Underpass. Accessed: 2024-09-17.

S. Vitalis, K. Arroyo Ohori, and J. Stoter. Incorporating Topological Representation in 3D City Models. *ISPRS International Journal of Geo-Information*, 8(8):347, 2019. doi: 10.3390/ijgi8080347. URL https://doi.org/10.3390/ijgi8080347.

S. Vitalis, A. Labetski, H. Ledoux, and J. Stoter. From road centrelines to carriageways—A reconstruction algorithm. *PLOS ONE*, 17(2):e0262801, 2022. doi: 10.1371/journal.pone.0262801. URL https://doi.org/10.1371/journal.pone.0262801.

J. Wang, G. Lawson, and Y. Shen. Automatic High-Fidelity 3D Road Network Modeling Based on 2D GIS Data. *Advances in Engineering Software*, 76:86–98, 2014. doi: 10.1016/j.advengsoft.2014.06.005. URL https://doi.org/10.1016/j.advengsoft.2014.06.005.

D. Wilkie, J. Sewall, and M. C. Lin. Transforming GIS Data into Functional Road Models for Large-Scale Traffic Simulation. *IEEE Transactions on Visualization and Computer Graphics*, 18(6):890–901, 2012. doi: 10.1109/tvcg.2011.116.

L. Xu, D. Mouzakidis, P. N. Riyadi, G. K. Nestoras, N. Liu, and S. Köbben. Integrate and Store 3D Roads and Terrain for Navigation Purposes. In *TUD Geomatics Synthesis Project GEO1101*, 2023a.

Z. Xu, Y. Liu, Y. Sun, M. Liu, and L. Wang. Centerlinedet: Centerline Graph Detection for Road Lanes with Vehicle-Mounted Sensors by Transformer for HD Map Generation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023b. doi: 10.1109/icra48891.2023.10161508.

Z. Xu et al. Topo-boundary: A Benchmark Dataset on Topological Road-Boundary Detection Using Aerial Images for Autonomous Driving. *IEEE Robotics and Automation Letters*, 6(4): 7248–7255, 2021. doi: 10.1109/LRA.2021.3097512.

B. Yang, X. Luan, and Q. Li. Generating hierarchical strokes from urban street networks based on Spatial Pattern Recognition. *International Journal of Geographical Information Science*, 25(12):2025–2050, 2011. doi: 10.1080/13658816.2011.570270.

L. Yang, Z. Zuo, R. Wang, Y. Ye, and M. Hu. Matching Road Network Combining Hierarchical Strokes and Probabilistic Relaxation Method. *The Open Automation and Control Systems Journal*, 6(1):268–276, 2014. doi: 10.2174/1874444301406010268.

X. Yu. OSM-Based Automatic Road Network Geometry Generation in Unity. Master's thesis, KTH Royal Institute of Technology, 2019.

Y. Yue, A. G.-O. Yeh, and Q. Li. Road Network Model for Vehicle Navigation Using Traffic Direction Approach. In *Lecture Notes in Geoinformation and Cartography*, pages 613–629. Springer, 2007. doi: 10.1007/978-3-540-68566-1_35.

T. Zhang, S. Arrigoni, M. Garozzo, D. Yang, and F. Cheli. A Lane-level Road Network Model with Global Continuity. *Transportation Research Part C: Emerging Technologies*, 71: 32–50, 2016. doi: 10.1016/j.trc.2016.07.003.

Z. Zhang, Q. Liu, and Y. Wang. Road Extraction by Deep Residual U-Net. *IEEE Geoscience and Remote Sensing Letters*, 15(5):749–753, 2018. doi: 10.1109/LGRS.2018.2802944.

Q. Zhu and Y. Li. Hierarchical lane-oriented 3D road-network model. *International Journal of Geographical Information Science*, 22(5):479–505, 2008. doi: 10.1080/13658810701492324.

Colophon

This document was typeset using LaTeX, using the KOMA-Script class `scrbook`. The main font is Palatino.

TU Delft