International Journal of Geo-Information

MDPI

Article

Automatic Extraction of Road Interchange Networks from Crowdsourced Trajectory Data: A Forward and Reverse Tracking Approach

Fengwei Jiao, Longgang Xiang * and Yuanyuan Deng

State Key Laboratory of LIESMARS, Wuhan University, 129 Luoyu Road, Wuhan 430079, China; fwjiao@whu.edu.cn (F.J.); deng_yyuan@whu.edu.cn (Y.D.)
* Correspondence: geoxlg@whu.edu.cn

Abstract: The generation of road interchange networks benefits various applications, such as vehicle navigation and intelligent transportation systems. Traditional methods often focus on common road structures but fail to fully utilize long-term trajectory continuity and flow information, leading to fragmented results and misidentification of overlapping roads as intersections. To address these limitations, we propose a forward and reverse tracking method for high-accuracy road interchange network generation. First, raw crowdsourced trajectory data is preprocessed by filtering out non-interchange trajectories and removing abnormal data based on both static and dynamic characteristics of the trajectories. Next, road subgraphs are extracted by identifying potential transition nodes, which are verified using directional and distribution information. Trajectory bifurcation is then performed at these nodes. Finally, a two-stage fusion process combines forward and reverse tracking results to produce a geometrically complete and topologically accurate road interchange network. Experiments using crowdsourced trajectory data from Shenzhen demonstrated highly accurate results, with 95.26% precision in geometric road network alignment and 90.06% accuracy in representing the connectivity of road interchange structures. Compared to existing methods, our approach enhanced accuracy in spatial alignment by 13.3% and improved the correctness of structural connections by 12.1%. The approach demonstrates strong performance across different types of interchanges, including cloverleaf, turbo, and trumpet interchanges.

Keywords: road interchange; forward and reverse tracking; transition node; trajectory continuity; crowdsourced trajectory data

check for updates

Academic Editors: Wolfgang Kainz and Hartwig H. Hochmair

Received: 15 March 2025
Revised: 13 June 2025
Accepted: 15 June 2025
Published: 17 June 2025

Citation: Jiao, F.; Xiang, L.; Deng, Y. Automatic Extraction of Road Interchange Networks from Crowdsourced Trajectory Data: A Forward and Reverse Tracking Approach. *ISPRS Int. J. Geo-Inf.* **2025**, *14*, 234. https://doi.org/10.3390/ijgi14060234

Copyright: © 2025 by the authors. Published by MDPI on behalf of the International Society for Photogrammetry and Remote Sensing. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Road interchanges are critical components of road networks, using grade-separations to align intersections of multiple surface transport routes at different heights (grades), preventing disruptions to traffic flow on other routes. They play a crucial role in optimizing traffic and reducing congestion [1,2] in urban areas. Research shows these areas are prone to accidents, especially in merging section [3]. Accurate road information (geometry and topology structures) is essential for reliable navigation services, enhancing safety and efficiency for drivers [4,5]. Traditional approaches to field mapping are costly due to labor-intensive processes and inefficient real-time updates [6].

Automatic extraction of road networks using multi-source geospatial data, e.g., remote sensing images, or crowdsourced trajectories, has long been a focus. In particular, extracting the geometric and topology structures of road interchanges from remote sensing

images has garner significant scholarly attention. Hu et al. [7] proposed a two-step approach involving road footprint classification for incremental growth and pruning of road trees, achieving 81% extraction accuracy in highway interchange areas. Zhang et al. [8] developed the NodeConnect model, which extracts road nodes and infers connectivity to represent road networks as undirected graphs. However, spatial occlusion between overpasses (upper-level straight lanes) and ramps (lower-level straight lanes) often causes fragmented lower-level road generation or pseudo-intersections at overlapping areas, leading to topological errors [7–9]. Additionally, the geometric complexity of interchanges, coupled with geometric and textural noise in remote sensing imagery (e.g., interference from vegetation, buildings, and vehicles), poses challenges in reconstructing interchanges with both geometric fidelity and directed topological connectivity. However, crowdsourced trajectory data goes beyond capturing road shapes and locations; it also unveils intricate road connections and densely clustered pathways that aerial images might miss. Long trajectory continuity enables the identification of underlying topological connections in multi-level road networks, effectively distinguishing geometric and topological structures of overlapping upper- and lower-level roads. Furthermore, trajectory flow direction aids in discerning divergence and convergence properties at key interchange nodes, facilitating the generation of directed road structures—attributes inherently unattainable from remote sensing images alone.

The use of trajectory data to infer road maps has been a subject of extensive research for many years. Map inference [10–13] entails leveraging trajectory data to identify, reconstruct, and refine road networks over time. However, previous methods often produce maps with low accuracy, especially in areas with complex structures [14–16] such as highway interchanges. In this section, we divide the current methods of map inference based on trajectory data into four categories: clustering-based methods, entry/exit points-based methods, trace-merging methods, and intersection-linking methods. The clustering method employs distance and azimuth information from trajectory data to delineate clusters, thereby facilitating the construction of the internal structure of highway interchanges. Stanojevic et al. [17] grouped successive observations in each trajectory into clusters with each cluster center as a vertex and edges formed between clusters based on their sequence. However, this approach considers only pairs of consecutive observations, neglecting the broader connectivity of the trajectory. As a result, it may lead to erroneous connections, such as fragmented or discontinuous patterns, particularly in areas with dense overpasses and multi-level trajectories. The entry/exit points-based methods detect intersection boundaries, identify interface points between intersection boundaries and connecting roads, reconstruct geometries by analyzing movement patterns between identified entry/exit nodes. Deng et al. [16] calculated similarity using the longest common subsequence to cluster the turning relationships within the interchange. However, the topology connection is not extracted and missed. The trace-merging method iteratively traverses all trajectories, merging trajectory points with near-parallel distributions based on positional and directional information. This process gradually combines nodes and edges to construct a road network graph. By modeling the attractiveness of trajectory points and grouping similar trajectories, Cao et al. [18] constructed a road network through traversal of the trajectory collection. He et al. [19] detected direction changes in the trajectory flow using local peaks, traced new road sections at bifurcations, and merged them based on the future alignment of the flow. Intersection-linking methods divide the interchange into several small intersections. By identifying large-angle turning points, this method segment long trajectories at different elevations. Wang et al. [20] extracted turning rules within intersections and centerlines of external road segments, and forms connections through boundary access points to create a road network. The method hinders the effective use

of trajectory continuity. As a result, it fails to distinguish between adjacent roads and generates false intersections at the interweaving sections of upper and lower-level roads, leading to topological errors. Table 1 systematically compares four trajectory-based road network construction method in terms of their characteristics and performance. In recent years, some researchers have fused vehicle trajectories and remote sensing images to recognize intersections or extract roads. Sun et al. [21] enhances road detection accuracy and generalizability to unmapped areas via specialized 1D filters and augmented training strategies. Qin et al. [22] proposes an incremental road network update method that integrates multi-source geospatial data through Hidden Markov Model-based change detection and deep learning-driven road extraction.

Table 1. Summary of trajectory-based map inference methods.

Method Category	Key Characteristics	Strengths	Weaknesses
Clustering	Groups GPS points into clusters via k-means, then connects cluster centroids	Fast and robust to trajectory data sparsity	Challenging to infer non-planar features; sensitive to the number of initial seeds and their locations
Entry/exit points-based	Detects entry/exit points at intersections, reconstructs roads via movement patterns	Effective for modeling turning relationships and detecting detailed geometry structure	Miss topology connection
Trace-merging	Starts from seed nodes and grows iteratively using trajectory continuity	Handles complex topologies (e.g., multi-level roads)	Spurious branching because of intertwined trajectories
Intersection-linking	Detects intersections first, then links them	Effective for intersection detection (including U-turns) and road connectivity preservation	Hard to distinguish between adjacent lanes

Additionally, some researchers have explored methods for generating the three-dimensional (3D) structure of highway interchanges using LiDAR data or crowdsourced trajectories with elevation information. For instance, Cheng et al. [23] introduced the concept of "structure unit" to decompose a complex multilayer interchange into simple units. Zhou et al. [24] designed the multi-constraint energy model to enhance curve reconstruction performance in different highway scenes. LiDAR data captures the fine details of highway interchanges, which is crucial for accurate modeling and reconstruction. However, Airborne LiDAR systems [25] can be costly to deploy and maintain, which may restrict their use in certain projects. The overall expenses, including the specialized equipment for data collection and the labor required for both acquisition and processing, make LiDAR a more expensive option compared to crowdsourced trajectory data. Consequently, there has been increasing interest in using crowdsourced trajectory data, which offers a more cost-effective alternative for large-scale analysis of highway interchanges and roads. Several studies have aimed to extract 3D information about road networks from crowdsourced trajectories that include elevation records. For example, Ren et al. [26] utilized semantic segmentation of elevation data to identify slopes and level sections in roads and ramps. Yang et al. [27] further improved the accuracy of segmenting ramps and main roads, as well as detecting the vertical order of road segments. A major limitation of this method is that most trajectory datasets are typically collected without reliable elevation data, as many GPS-based systems do not consistently include elevation information. Even when elevation data is available, its quality and reliability are often questioned [28]. Inaccuracies in elevation measurements can result from errors in GPS signals, inconsistencies in the device's altitude readings, or the limitations of consumer-grade sensors [29,30]. As a result, the extraction of precise 3D

road geometry from such data can be challenging, and the accuracy of the derived models may be compromised.

Constructing a high-accuracy road graph of highway interchanges using crowdsourced trajectory data encounters two main challenges. Firstly, the intricate geometry and topology of interchanges, which include crossing stacked roads at different heights, dense ramps, and indistinguishable adjacent roads, present significant obstacles. Secondly, the reliability of crowdsourced trajectory data is compromised by biases, inaccuracies, and inconsistencies arising from GPS errors, diverse driver behaviors and varying traffic conditions [29]. The presence of non-vehicle trajectories, such as pedestrian and cyclist paths, further complicates the extraction of interchange road structures. Additionally, the spatial distribution of trajectories exhibits heterogeneity [31] influenced by factors such as road class and regional disparity.

Although considerable research have been dedicated to automating the generation and real-time updating of road networks, recent studies highlight that existing algorithms tend to work well for common road structures [10–13] but fail to generate high-accuracy maps in complex interchanges [14–16]. Most of the existing map inference methods are devoted to generating road interchange map without fully utilizing trajectory continuity and flow information. They often make wrong connections between adjacent roads [17,19] and misidentify intersections where different road levels cross but do not meet at the same grade [20]. This paper presents a novel approach for generating road interchange map based on crowdsourced trajectories. This method is distinct to other approaches for the following reasons: Our approach combines intersection linking and trace-merging. We have found that by using the continuity of trajectories to track the splitting and merging of flow paths, and by incorporating direction and distribution information of forward movement, we can effectively identify intersections within road interchanges. This approach allows us to accurately pinpoint the critical nodes of road interchanges. In summary, we make the following contributions:

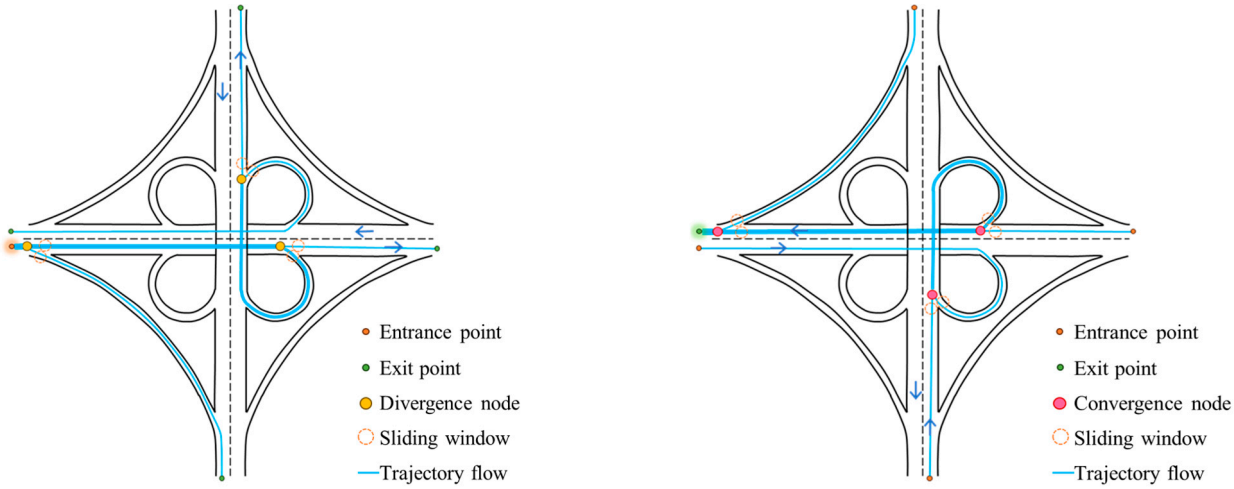
- We develop a novel road network generation method particularly for road interchanges. This method adopts a divide-and-conquer strategy by decomposing road interchange networks into subnetworks for various types of road interchanges and addresses the challenge posed by the multi-level complex structure and crowdsourced data quality issues.
- A forward and reverse tracking mechanism that leverages long-term trajectory continuity is proposed to accurately identify transition nodes at road interchanges. This method avoids generating false intersections in areas where upper and lower-level roads interweave. Transition nodes are categorized into divergence nodes (via forward tracking) and convergence nodes (via reverse tracking).
- Extensive experiments were conducted using real-world trajectory data. The proposed method achieves an average improvement of 13.3% in GEO $-$ F_1 score and 12.1% in TOPO $-$ F_1 score, outperforming two baseline methods. The method demonstrates robust performance across various types of road interchanges, including cloverleaf, turbo, and trumpet interchanges.

The remainder of this paper is organized as follows. Section 2 reviews the related work. Section 3 presents the proposed method for generating road interchange map. The experimental analyses are detailed in Section 4. Conclusions and future works are discussed at the end of the paper.

2. Methodology

Traces represent sequences of discrete points that depict a vehicle's driving path. Typically, vehicles enter the highway via an on-ramp and exit via an off-ramp. As illustrated in

Figure 1a, trajectory flows entering the same entrance continuously diverge at the intersection and exit the road-interchange region through different exits. Conversely, trajectory flows from different entrances continuously merge at the intersection and exit the region via the same exit (Figure 1b). Divergence nodes are defined where trace bundles following the road network fork at intersections (diverge or off-ramp) while the convergence nodes are defined where trace bundles match with each other (merge or on-ramp).



(**a**) (**b**)

Figure 1. Trajectory flow divergence and convergence. (**a**) Trajectory flow divergence; (**b**) Trajectory flow convergence.

By leveraging the continuity of trajectories, we introduce a method that starts at the entrance/exit points to detect divergence nodes through forward tracking (i.e., following the trajectory sequence), thus creating a subgraph of the highway interchange. Conversely, convergence nodes are identified through reverse tracking (i.e., opposite to the trajectory sequence), and the road subgraphs from multiple directions are then integrated to construct the road interchange network.

As illustrated in Figure 2, the geometric and topological structures of a highway interchange are represented in a road network model. This model is a directed graph composed of nodes and edges. The nodes represent divergence and convergence nodes, which are the intersections within the interchange, while the edges correspond to the centerlines of road segments, indicating the direction and topological connections of the roads. The geometric structure of the interchange is concerned with the shape and distribution of the roadways. It focuses on the centerline ensemble of each road segment, with the exit and entrance points serving as the endpoints. The topological structure, on the other hand, deals with the connectivity and direction of the network. This is represented by nodes and directed paths that form connections between them. The generation of the topological structure is based on the extraction of the geometric structure. Essentially, the geometric layout of the roads provides the foundation for determining how they are connected in terms of directionality and intersections, which ultimately defines the interchange's overall topological framework.

In this paper, we propose a three-step method that converts sequences of tracked points into a routable road-level map for highway interchanges, as illustrated in Figure 3.

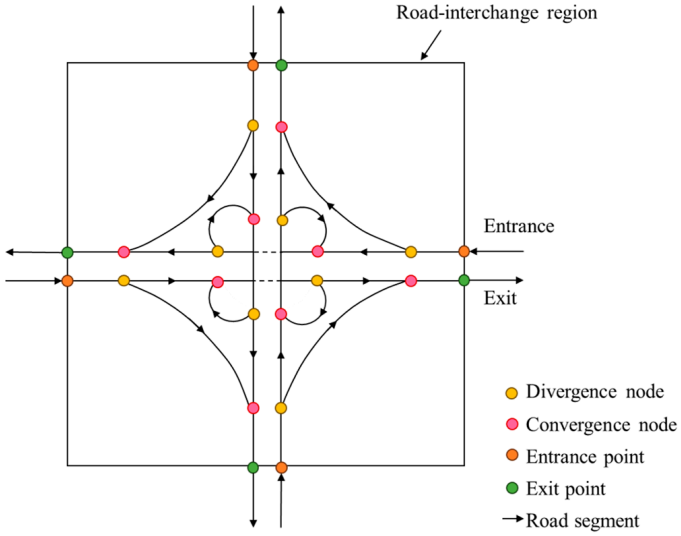


Figure 2. Road interchange network model.

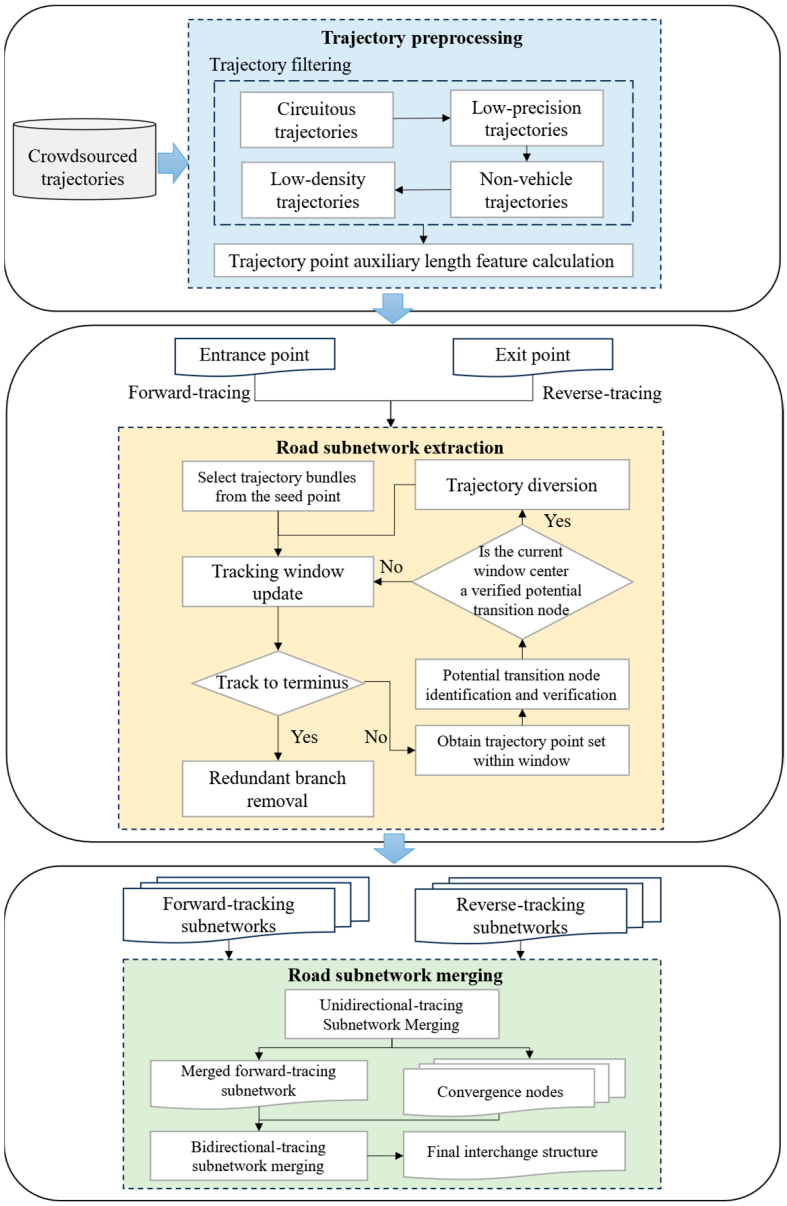


Figure 3. Workflow of the proposed method.

First, the crowdsourced trajectory data undergoes preprocessing to filter out noisy or irrelevant trajectories. Constraint rules are applied to exclude circuitous, low-precision, non-vehicle, and low-density trajectories based on factors such as morphological features, heading angle, speed, and other relevant features.

Next, the road subgraphs corresponding to multiple entrance/exit points are extracted. A single entrance/exit is selected as a seed point, and potential transition nodes, where trajectory flows either diverge or converge, are identified after each update of the tracking window. These potential transition nodes are further verified by combining the direction and distribution information of forward movement. Trajectory bifurcation is performed at the transition nodes, and tracking continues until the end of the trajectory bundles. After redundant branches are removed, the road interchange subgraph is extracted.

Finally, a two-stage fusion of the road interchange subgraphs is performed. First, the subgraphs from forward and reverse tracking are fused separately, considering heading, trajectory bundle information, and spatial relationships. Then, the forward fused structure is combined with the convergence nodes from reverse tracking to extract a geometrically complete and topologically accurate road interchange network.

2.1. Trajectory Preprocessing

Crowdsourced trajectory data comes from various sources and typically includes trajectories from categories such as automobiles, bicycles, and pedestrians. Upon observation, there may be a cluster of buildings near the road interchange, with footpaths, urban greenways, and other features underneath. In addition, due to GPS errors, signal blocking by tall buildings, and uncertainties in the behavior of moving objects, the original data often has quality issues such as positioning drift, messy patterns, and redundancy. If not addressed, these problems can negatively impact the extraction of road interchange networks.

After analysis, two types of trajectories need to be filtered: (1) trajectories from non-road-interchange regions, which include circuitous trajectories, non-vehicle trajectories, and low-density trajectories; and (2) abnormally deviating trajectories, which refer to low-precision trajectories that deviate locally or integrally from the road surface. High-confidence trajectories distributed on road interchanges typically exhibit the following characteristics: (1) narrow geometric shape with a large spatial span; (2) smooth trajectory with fewer drift points; (3) high average and maximum velocity; (4) high density and homogeneity. Based on these features, this paper proposes a trajectory filtering strategy consists of the following steps:

1. Circuitous trajectory filtering: Circuitous trajectories, characterized by frequent directional changes within localized spatial ranges, form compact morphologies distinguishable via geometric metrics, as illustrated in Figure 4a. The minimum enclosing rectangle (MER) of such trajectories provides two key indicators: the aspect ratio, which identifies narrow or clustered patterns (low values), and the area, reflecting their limited spatial extent. These traits correlate with a low morphology index (SI). Trajectories with SI below a predefined a threshold value are classified as circuitous trajectories and eliminated during preprocessing. The threshold value is defined as the mth-percentile of all trajectory morphology indices. SI is calculated as described below:

$$SI = w \cdot R_m + (1 - w) \cdot S_m \tag{1}$$

where R_m and S_m represent the values normalized to the aspect ratio and area of the minimum enclosing rectangle for each trajectory, respectively, and w is the influence weight. Since the aspect ratio and area of the minimum enclosing rectangle are equally important in determining whether the trajectory is distributed on the overpass, we set the value of w to 0.5.

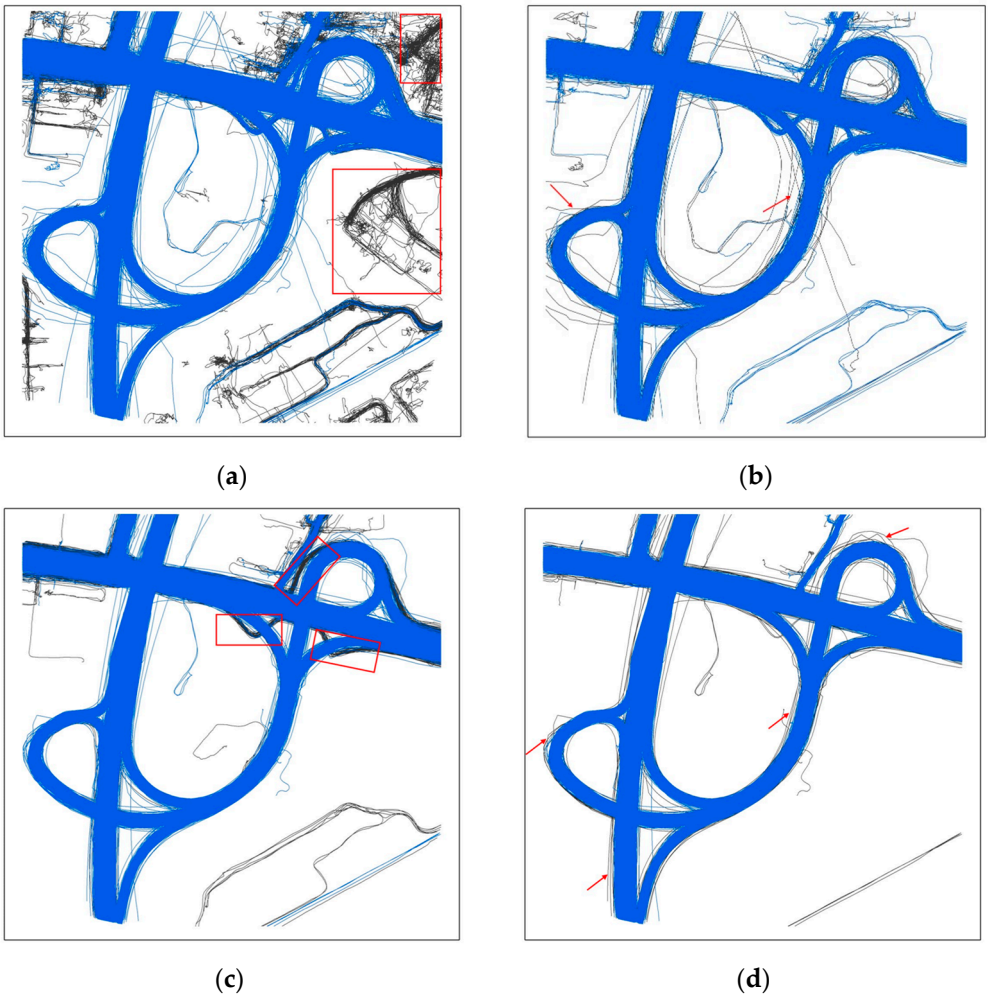


Figure 4. Data preprocessing pipeline for crowdsourced trajectory filtering: (**a**) Dafapu Overpass trajectories within a rectangular region: trajectories retained after Step 1 circuitous trajectory filtering (blue), original trajectories (black), and annotated circuitous examples (red boxes); (**b**) Trajectories retained after Step 2 low-precision trajectory filtering (blue), trajectories retained after Step 1 (black), and annotated low-precision examples (red arrows); (**c**) Trajectories retained after Step 3 non-vehicle-trajectory filtering (blue), trajectories retained after Step 2 (black), and annotated non-vehicle examples (red boxes); (**d**) Trajectories retained after Step 4 low-density trajectory filtering (blue), trajectories retained after Step 3 (black), and annotated low-density examples (red arrows).

2. Low-precision trajectory filtering: Low-precision trajectories are defined by a high proportion of points deviating significantly from the road surface, as shown in Figure 4b. To quantify this deviation, we compare two orientation metrics: (1) the original heading angle (θ_1) of individual trajectory points, reflecting instantaneous vehicle direction, and (2) the direction angle (θ_2) derived from adjacent points, representing the local road segment's morphology. The angular distance dis_θ between θ_1 and θ_2 serves as a deviation indicator—values exceeding a threshold δ_θ suggest abnormal lateral drift. Since angular differences in raw angle space do not linearly correlate with geometric distance, we project angles onto a unit circle to compute dis_θ accurately, as illustrated in Figure 5. The dis_θ is calculated as follows:

$$dis_\theta = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} x_i = \cos\theta_i\,, \quad y_i = \sin\theta_i \quad (i = 1, 2) \tag{2}$$

where θ_1 represents the original heading angle of trajectory points, θ_2 represents the direction angle calculated from adjacent trajectory points, (x_i, y_i) are the coordinates of

θ_i on the unit circle. Trajectories with a high proportion of trajectory points exhibiting low horizontal positioning accuracy, exceeding a specified ratio, are filtered out.

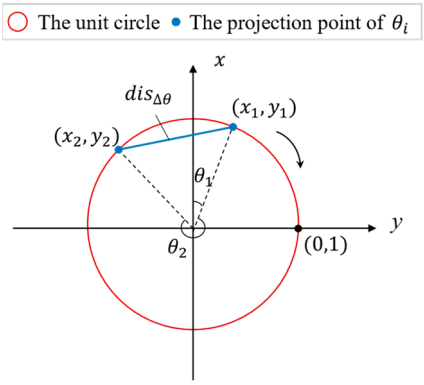


Figure 5. Schematic illustration of angular difference distance.

3. Non-vehicle trajectory filtering: Non-vehicle trajectories (e.g., walking, cycling) are spatially confined to non-motorized lanes and exhibit distinct speed profiles compared to vehicular movement, as illustrated in Figure 4c. Multi-level interchanges prioritize vehicular flow through separated ramps and straight lanes, enabling higher sustained speeds. We leverage this design feature to differentiate vehicle trajectories from non-vehicle ones: trajectories with an overall average speed $V_k < \delta_k$ and maximum speed $V_h < \delta_h$ are classified as non-vehicle trajectories. These trajectories are filtered during preprocessing to retain only vehicular movement patterns.
4. Low-density trajectory filtering: Low-density trajectories are defined by a sparse distribution of points deviating from dense vehicular trace bundles, as illustrated in Figure 4d. Road interchanges, designed for high traffic capacity, exhibit concentrated vehicle trajectory clusters due to sustained traffic flow. We quantify trajectory sparsity using neighboring flux F_t, which measures local trajectory density over extended segments. Trajectories with F_t persistently below an empirically defined threshold τ are identified as low-density outliers and filtered out during preprocessing. This ensures retention of statistically representative vehicular movement patterns while removing anomalous deviations.

Auxiliary length features are computed for each trajectory point, specifically the cumulative length from each point to the endpoints of its corresponding trajectory. This feature indirectly characterizes the connectivity of the trajectories. The data preprocessing results in a high-confidence trajectory set T, located within the main body of the interchange, which serves as input for road subnetwork extraction.

2.2. Road Subnetwork Extraction

The spatial distribution of vehicle trajectories is inherently constrained by the geometry of the road network. At road interchanges, the bifurcation and merging processes of the network correspond to the diverging and merging behaviors observed in trajectory data. In this paper, we introduce a novel method for extracting road subnetworks. This approach integrates local directional information with the distribution patterns of forward movement to dynamically detect road diversion events. The method enables the extraction of both the geometric and topological connectivity of road interchanges, forming a road subnetwork represented by nodes and directed edges.

The road subnetwork extraction begins by selecting an entrance/exit (orange/green dots in Figure 6a) as the seed. From this seed point, trajectory bundles (blue lines) flowing into the road interchange are extracted from the high-confidence trajectory set T. A circular

window (yellow dashed line) is then used to continuously slide along the trajectory, updating at each step, until the end of the trajectory is reached. This process models individual substructures as directed graphs. Each center of the sliding window corresponds to a node in the subgraph, which records geometric position information. The relationship between adjacent windows, established by the sliding order, defines the connections between nodes. These connections are represented as directed edges (yellow arrows) between nodes. Transition nodes (pink dots in Figure 6b) associate road segments (blue lines) with intersections, forming a directed graph that topologically links substructures.

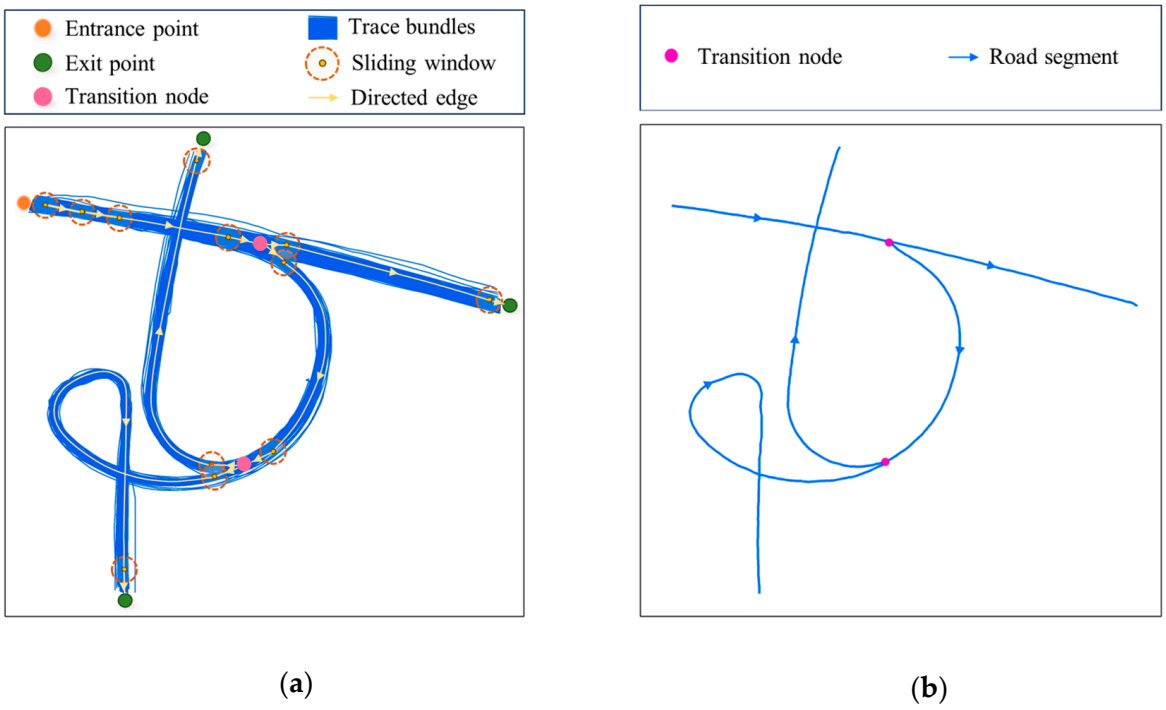


Figure 6. Schematic illustration of road subnetwork extraction. (**a**) Sliding window method for road subnetwork extraction; (**b**) Road subnetwork structure.

After each update of the sliding window, the trajectory points within the window are collected. The peak heading angle of these points is then statistically computed, and potential transition nodes are identified and verified. If the center of the current window is identified as a potential transition node and passes the validation, its location is considered a credible road bifurcation. At this point, trajectories passing through the neighborhood of the transition node are diverted into separate trajectory bundles, which are then tracked recursively. If the center of the current window is not identified as a transition node, the window is updated and the procedure continues. This process continues until the end of the tracking, at which point redundant branches are removed to form the final road subnetwork.

2.2.1. Transition Node Detection

Most existing methods assume that aligned trajectory data of a road are nearly parallel, characterized by both positional and directional proximity. These methods typically set fixed thresholds for distance and heading angle differences to cluster similar trajectory points or to incrementally construct directed graphs. The method presented in this paper integrates heading angle difference and trajectory continuity for improved tracking.

The tracking window with radius R is updated by sliding it along the specified direction and step size d, and connecting edges are generated between the centers of the updated windows. Given that arterials in road interchanges typically have higher road

grades, more lanes, and wider roads than other internal arterials or turn ramps, the tracking window should reflect the real road conditions. To ensure that the local information carried by the track points in the window accurately represents the road, the window's radius is primarily determined by the width of the high-grade arterials. However, a larger window may cause track points to be unevenly distributed, and the tracking direction's deviation will gradually accumulate over time. This leads to the tracked line shifting away from the road's center and may even cause transition nodes to be missed due to insufficient local information. Therefore, after each window update, the center of mass of the track points in the window is used to correct the window's position, ensuring that the tracking path remains aligned with the road's cross-section and stays centered on the road.

Initially, the method utilizes heading angle divergence to identify potential transition nodes. The divergence trend is then verified by pre-tracking the distance difference between two branches, helping to identify high-confidence transition nodes. Specifically, the heading angle space is discretized into 360 intervals. Using kernel density estimation (KDE; Gaussian kernel) [32,33], a normalized probability density curve of the heading angles within the current window is generated. This curve is analyzed to obtain the peak heading angle and the number of peaks within the window. As illustrated in Figure 7a, the center of the next window is marked as a potential transition node if the number of peaks increases compared to the current window.

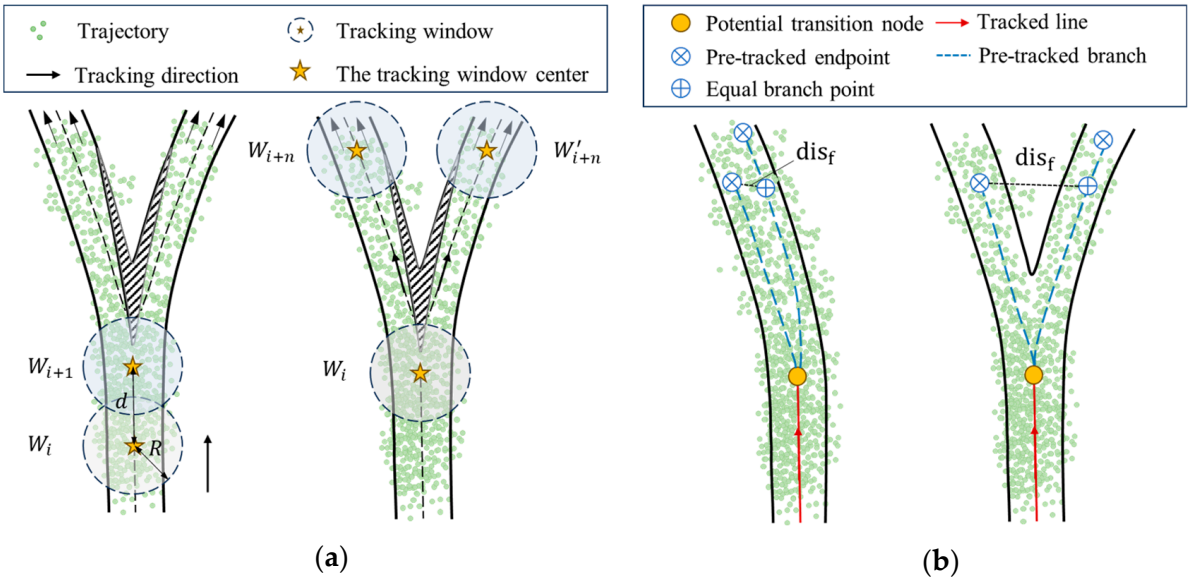


Figure 7. Schematic illustration of transition node detection. (**a**) Node identification and pre-tracking; (**b**) Pre-tracked branch distribution pattern.

The authenticity of the potential transition node is verified by examining the divergence trend of the subsequent trajectory sequence through pre-tracing (Figure 7a). Specifically, the tracking window is slid n times from the potential transition node along each heading angle peak. Since the center of the sliding window is updated to the center of mass of the trajectory points within the window, and the distribution of branching trajectory points may vary, the spacing between the pre-tracking end windows does not directly reflect the divergence trend of the trajectories. To address this, the side of the two branches closer to the potential transition node is selected, and equal-length points are intercepted at that same distance on the other branch. Next, the distance between the corresponding points of the two branches is calculated. If this distance exceeds a predetermined threshold, the point is considered a valid transition node (Figure 7b). Otherwise, it is deemed a spurious transition node (Figure 7b).

2.2.2. Trajectory Diversion

In road bifurcations, traffic density and speed can vary significantly across different directions. To ensure driving safety and reduce conflicts, buffer zones, such as transition sections, are often implemented. After a bifurcation, the roads may not immediately separate in space, causing trajectory points from different branches to be in close proximity. This overlap can lead to interference in the tracking process. To address this issue, we propose a trajectory diversion method based on K-means++. This method leverages the continuity of vehicle trajectories and the semantic information of local trajectory segments. By recognizing patterns in the directional flow of these segments, we transform the problem of trajectory diversion into the identification of clumping patterns for different road branches. This approach helps separate the trajectories of different branches, reducing interference and improving tracking accuracy.

Distinguishing between trajectory bundles from different branch roads is computationally complex when relying on the similarity of complete trajectories. Vehicle trajectories represent restricted geographic flows [34,35], with turning behaviors strictly constrained by the road network. After passing the transition node, trajectories heading toward the same branch road exhibit similar patterns within a certain distance, forming a characteristic clumping pattern. This method utilizes local semantic information by intercepting a fixed-length (100 m) segment of the trajectory along the tracking direction, with the transition node as the center of the starting neighborhood. We adopt the polar coordinate model of flow clustering to abstractly represent each trajectory segment as a ternary group, consisting of the starting point, ending point, and direction: $((x_o, y_o), (x_D, y_D), \alpha)$. Here, (x_o, y_o) represents the coordinates of the starting point, (x_D, y_D) represents the coordinates of the ending point and α is the direction of the directed line from the starting point to the ending point. The direction α is measured in degrees, with 0° indicating true north.

By reducing the dimensionality of the trajectory representation, we retain its original features while enhancing computational efficiency for subsequent flow pattern recognition. Given that the intercepted trajectory segments have nearly equal lengths and start points near the bifurcation, it can be inferred that the endpoints of segments with the same pattern are adjacent, while endpoints from different patterns are separated. Consequently, the ternary representation can be simplified to a single directional element, and trajectory flow clumps can be identified by recognizing the aggregation of direction angles in angle space, completing the trajectory diversion.

This paper adopts the K-means++ method [36] to identify clustering patterns in the direction of trajectory segments in an unsupervised manner. This approach not only converges quickly but also effectively avoids diversion failure caused by the proximity of randomly initialized clustering centers. The procedure begins by using the peak number of heading angles obtained from transition node detection as the prior number of clusters. Initial clustering centers are then randomly selected from the sample space of trajectory segment direction angles, ensuring that they are as far apart as possible. Next, points in the sample space are assigned to clusters based on minimizing angular differences, and the cluster centers are recalculated. The iteration process continues until the changes in cluster centers are smaller than 1°, at which point the algorithm halts. The trajectories corresponding to the direction angles within each cluster form distinct trajectory bundles after diversion.

2.2.3. Redundant Branch Removal

The trajectory data involves significant uncertainty during the positioning process, resulting in errors in accuracy that prevent full precision. At the initial stage of a road bifurcation, the trajectories of different branch roads are often intertwined. The trajectory

diversion is imprecise, and trajectories from different directions may not be fully separated. However, as the road extends, the trajectories begin to diverge more distinctly. Despite this, transition nodes may be repeatedly detected in the same intersection area, leading to redundant branches. To ensure the accuracy of road subnetwork merging, these redundant branches must be eliminated. The algorithm consists of the following steps:

1. Repeated bifurcation detection: At road interchanges, roads typically do not diverge frequently over short distances. Instead, redundant branches are often associated with repeated transition nodes, with short branches serving as indicators of repeated bifurcation signal, as illustrated in Figure 8a. These short branches are used to detect potential repeated transition nodes.

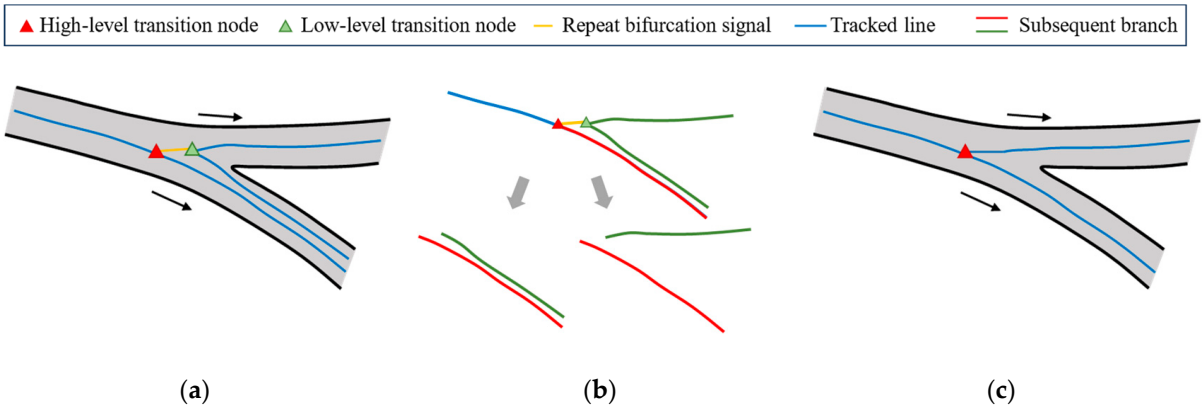


Figure 8. Schematic illustration of redundant branch removal. (**a**) Repeated bifurcation detection; (**b**) Subsequent branch matching; (**c**) Redundant branch pruning.

2. Subsequent branch matching: The two transition nodes associated with a short branch are defined as a high-level transition node and a low-level transition node, based on their occurrence in the direction of branch extension. The matching of subsequent branches at these transition nodes is determined by the maximum overlap of the outer rectangle of the branch R_o, the difference in branch direction $\Delta\omega$, and the projection distance of the branch endpoint dis_e. The branch matching process is illustrated in Figure 8b, with the meaning of each index shown in Figure 9. When the conditions in Formula (3) are met, the two branches are considered successfully matched. Since the matched branches represent the same road section structure, redundant branches are eliminated.

$$\begin{cases} R_o \geq 0.8 \; or \; dis_e < 10m \\ \Delta\omega \leq 20^\circ \end{cases} \tag{3}$$

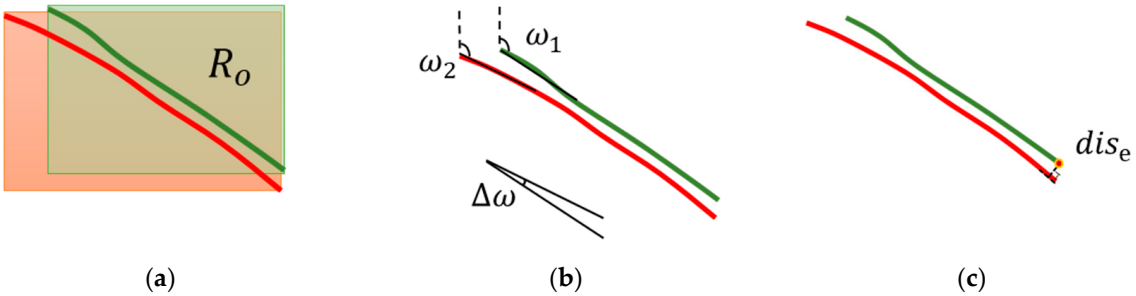


Figure 9. Schematic illustration of branch matching metric. (**a**) Max overlap of branch rectangle; (**b**) Branch direction difference; (**c**) Endpoint projection distance.

3. Redundant Branch Pruning: As shown in Figure 8c, when a branch is successfully matched and the end connected to the low-level transition node does not continue to bifurcate, the branch is directly deleted. If bifurcation continues, the local structure at the branch end is first integrated, and then redundant branches are eliminated. The integration of the local structure also involves sub-structure fusion, which will be discussed in detail in Section 2.3.

2.3. Road Subnetwork Merging

In this paper, we apply the concept of divide and conquer to simplify and decompose the extraction of a complete road network in a road interchange into multiple sub-networks. Each sub-network captures a portion of both the geometric structure and topological connections of the interchange. Specifically, the forward-tracking subnetwork contains road diversion information, while the reverse-tracking subnetwork holds road merging information. To reconstruct the full road interchange structure, we propose a two-stage subnetwork fusion method. This approach fuses subnetworks with the same tracking direction and integrates complementary tracking networks to establish the complete connectivity of the road interchange.

2.3.1. Unidirectional-Tracing Subnetwork Merging

The traffic flow in a road interchange exhibits the characteristics of multidirectional intersections, where traffic from different entrances shares common road sections. As a result, the sub-networks extracted through forward tracking often exhibit varying degrees of information overlap. The merging of unidirectional-tracing sub-networks begins by using the same road diversion location. It then identifies transition nodes within sub-networks that correspond to the same road diversion location, allowing for the discovery and fusion of structural fragments that are spatially and morphologically consistent. Ultimately, the geometric and structural information from sub-networks with the same tracking direction is integrated into a unified whole. The algorithm follows these steps:

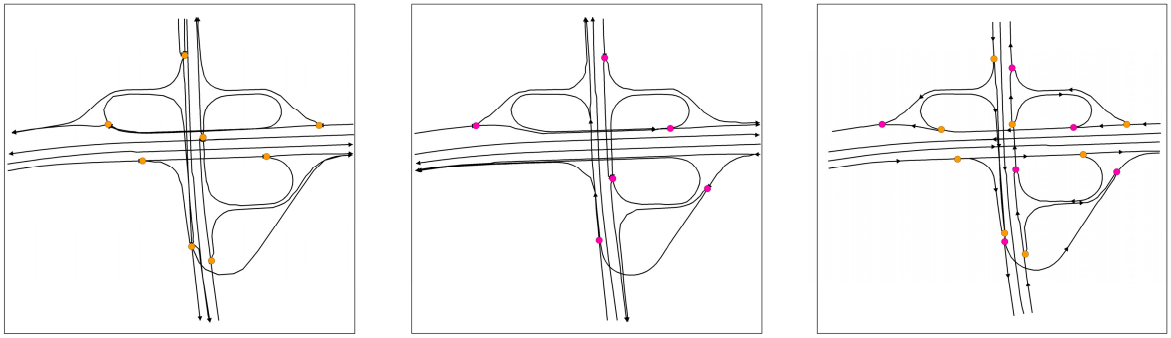
1. Dynamically identify clusters of transition nodes within all sub-networks that are spatially close to each other using the DBSCAN method.
2. For each cluster, select two transition nodes one at a time, and determine whether they represent the same road bifurcation by checking if the direction of the road at both nodes differs by no more than 20°. Once identified, match the subsequent branches of these two points (following the method outlined in Section 2.2.3). The center of these two nodes is then taken as the new transition node after merging. Matched branches are aligned, resampled, and fused. Connection relationships between the fused transition node and branches are updated to preserve topological consistency before and after fusion. This process is repeated until all transition nodes within the cluster that represent the same road bifurcation are merged.
3. Following these steps, transition nodes, branches, and their clusters are updated iteratively. Structural fragment fusion terminates once each cluster retains a single transition node or clusters correspond to distinct road bifurcations.

Due to the heterogeneity of trajectory data distribution, not all transition nodes can be fully extracted from each sub-network, resulting in unmerged structural fragments. To address this, a 5-m buffer is applied around the existing transition nodes. If a branch with a directional deviation $\leq 20°$ from the transition node's road orientation enters this zone, a structural inconsistency is flagged. The integration matching table records the transition node ID, target branch, and its nearest folding point to the node. In the consistency refinement integration phase, branches to be fused are interrupted according to the integration matching table. Structural fragments are then merged using the same method as described

in the previous section. This process results in a geometrically complete and topologically consistent road interchange structure, incorporating information from all subnetworks with the same tracking direction, referred to as the unidirectional-tracing subnetwork.

2.3.2. Bidirectional-Tracing Subnetwork Merging

The unidirectional-tracing subnetwork provides essentially complete geometric structural information but only includes topological details about diversions or mergers. The bidirectional-tracing subnetwork achieves complete structure extraction by complementarily merging the forward-tracing subnetwork and convergence nodes of the reverse-tracing subnetwork, as illustrated in Figure 10. This process can also be described as structural consistency refinement and integration.



(**a**) (**b**) (**c**)

Figure 10. Schematic illustration of bidirectional-tracing subnetwork merging. (**a**) Forward-tracing subnetwork; (**b**) Convergence nodes; (**c**) Results of bidirectional subnetwork merging.

The complementary fusion process centers on the convergence node for buffer analysis. It identifies branches in the forward-tracking structure that pass through the convergence node's vicinity. When detecting structural inconsistencies, both the directional difference between the road at the convergence node and the branch passing through the buffer zone, as well as the degree of trajectory matching, are considered to determine whether a record should be added to the matching table. Once the matching table is constructed, the branches are interrupted sequentially based on the records, and the branch segments on the same road are fused systematically to complete the complementary fusion. This process results in the extraction of a geometrically and topologically complete centerline-level road interchange structure.

3. Experiments and Evaluation

3.1. Experiment Data and Settings

This study focuses on road interchanges in Shenzhen, a city with a complex and interconnected road network featuring various types of interchanges, which allows for a comprehensive evaluation of the proposed method. The dataset comprises crowdsourced trajectories collected from navigation-enabled mobile devices during diverse travel modes (walking, cycling, driving) between 8:00 and 20:00 on 22 December 2021. Each trajectory point includes five attributes: trajectory ID, timestamp, latitude, longitude, and heading angle, with sampling intervals of 0–10 s and positioning accuracy of 1–30 m.

A preprocessing pipeline (Section 2.1) refined the raw data, which initially contained 301,402 trajectories and 2.1 million points. Postprocessing retained 281,657 trajectories and 1.7 million points after noise removal and outlier elimination. As shown in Figure 11, nine representative Shenzhen overpasses were selected as test areas, each defined by a

rectangular boundary encompassing the interchange's main structure. The preprocessed data volume per test area ranges from 17,931 to 39,615 trajectories, averaging 63 points per trajectory, reflecting variations in interchange size and traffic density.

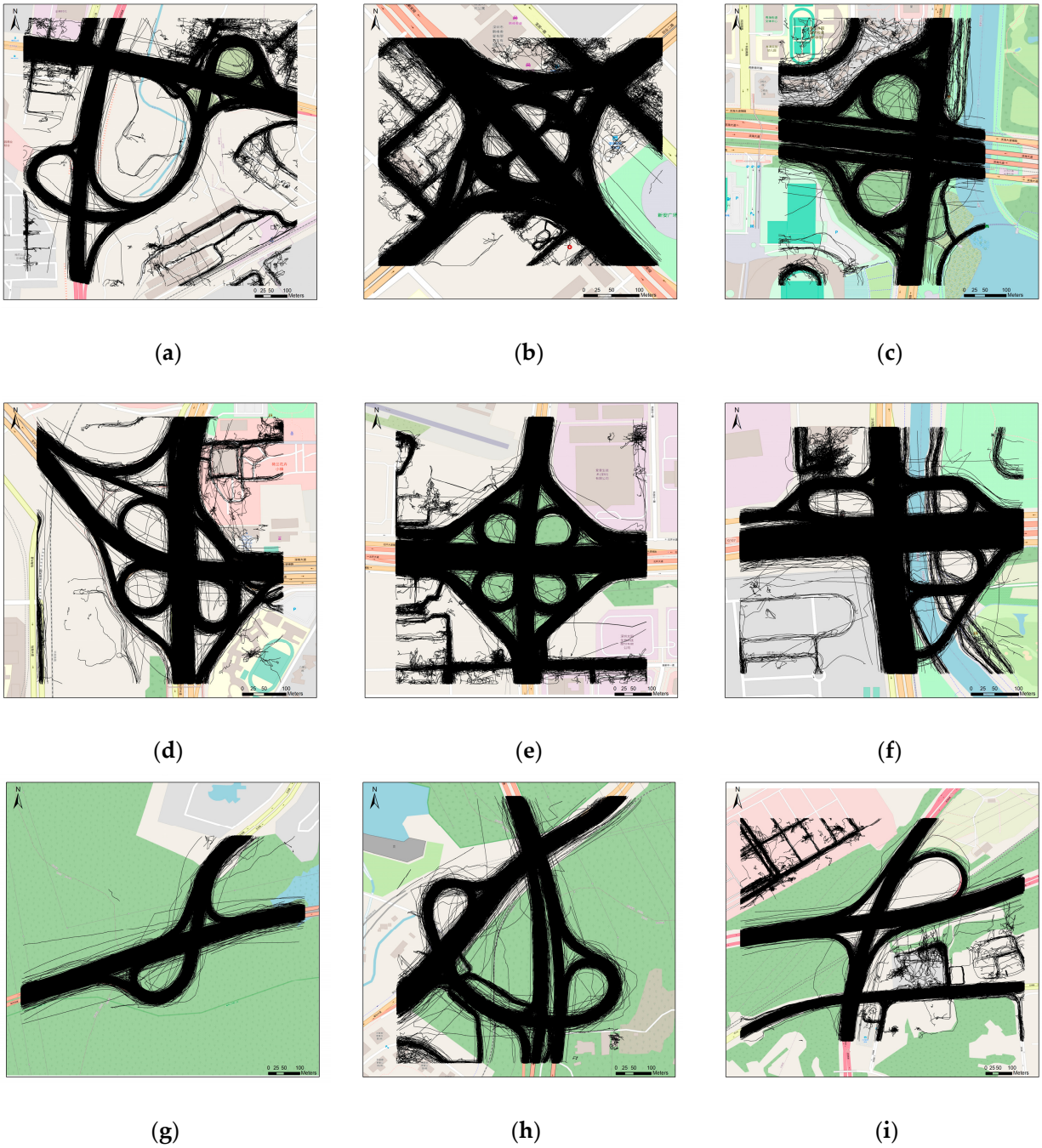


Figure 11. Experiment data. (**a**) Dafapu; (**b**) Chuangye; (**c**) Binhai; (**d**) Baoan; (**e**) Beihuan; (**f**) Shahexi; (**g**) Yulong; (**h**) Liuxian; (**i**) Shuiyan.

The proposed method was implemented in Python 3.8. All experiments were conducted on a desktop computer running Windows 10 and equipped with an AMD Ryzen 7 5800 H processor (3.20 GHz) and 16 GB of RAM.

3.2. Assessment Method

We evaluate inferred road interchange maps on two metrics: GEO and TOPO, which is commonly used in related work [37]. In this study, OSM maps are manually edited with reference to satellite images as the ground truth. The evaluation method involves resampling both the extracted road interchange network and the truth map at the same sampling interval, effectively performing a matching evaluation.

The geometric matching between the extracted road network and the real overpasses is quantified using the GEO index. Specifically, a 10-m sampling interval is used, with "marbles" representing the extracted road network and "holes" representing the true value map. If the distance between a marble and a hole is less than 10 m, and the directional difference between the corresponding road segments is less than 20°, the marble and hole are considered a match. The calculation method is detailed in Equation (1). The topological accuracy of the road interchange network is evaluated using the TOPO index, which focuses on the local structure. Initially, marbles in the generated road network are matched with holes in the truth map. For each successfully matched hole in the truth map, a reachable radius is set, and the F_1 value for the local area is calculated. This process is repeated for all successfully matched holes. The average F_1 value is then used as the overall TOPO evaluation, calculated as shown in the following equation.

$$precision = \frac{\#matched_marbles}{all_marbles} \tag{4}$$

$$recall = \frac{\#matched_holes}{all_holes} \tag{5}$$

$$F_1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{6}$$

where *matched_marbles* refers to the number of marbles that are successfully matched, while *all_marbles* represents the total number of marbles in the set. Similarly, *matched_holes* denotes the number of holes that are successfully matched, and *all_holes* represents the total number of holes in the set.

$$TOPO - F_1 = \frac{1}{n} \sum_{i=1}^{n} F_1(z_i) \tag{7}$$

where n represents the number of holes that are successfully matched one by one by setting marbles and holes in the generated road network and the truth map, respectively, at a 10-m sampling interval. The matching condition is that the distance between the two is less than 10 m, and the directional difference is less than 20°. z_i refers to the circular domain centered on the ith successfully matched hole, with a reachable radius set to 300 m. $F_1(z_i)$ is the F_1 value of the local road network within the circular domain z.

To reduce computational complexity during structural matching, we resample both the extracted and ground-truth road networks at 10-m intervals. As shown in Figure 12 three representative interchanges, including the Beihuan, Chuangye, and Dafapu interchanges (i.e., cloverleaf, turbo, and trumpet types, respectively) are evaluated under matching radii ranging from 5 m to 30 m. The results demonstrate that a 5 m radius leads to underestimated precision due to trajectory positioning errors. When the matching radius is greater than 10 m, the evaluation accuracy will not be significantly improved due to the increase of the matching range, so this paper sets the matching radius to 10 m.

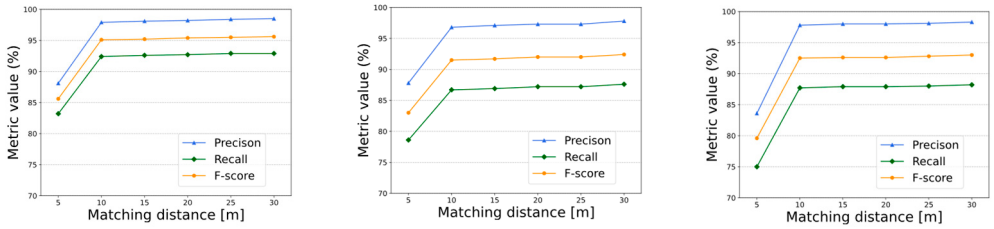


Figure 12. GEO Metric values at different matching distances. (**a**) Beihuan; (**b**) Chuangye; (**c**) Dafapu.

3.3. Experiment Results

3.3.1. Parameter Settings

Crowd-sourced trajectories were preprocessed to retain data localized above overpass main structures. Non-vehicle trajectories were filtered by speed thresholds: pedestrian walking (4–6 km/h), cycling (15–25 km/h), and electric vehicle speeds ($\leq$25 km/h), contrasting with motorized vehicles (60–120 km/h). To distinguish automobile trajectories from others, the parameters δ_k (overall average speed limit) and δ_h (maximum speed limit for trajectory points) were empirically calibrated. Aligning with traffic regulations, road design standards, and experimental validation, this study adopts δ_k = 25 km/h and δ_h = 45 km/h.

The results of the road interchange network and the efficiency of the algorithm are influenced by the radius of the sliding window R and the step size d. Overpasses typically contain high-grade arterial roads and turn ramps, with road widths ranging from 5 to 20 m. A window that is too small may fail to cover the data of high-grade arterial road cross-sections, resulting in incomplete local information. Conversely, a window that is too large can cause excessive angular variation, making it difficult to accurately capture the angular divergence of local roads. The step size d affects the number of nodes in the final structure. A larger d results in fewer nodes, potentially overlooking small heading divergences, while a smaller d leads to a denser node distribution, which increases redundant pre-tracking calculations and significantly reduces algorithm performance. In this study, the sliding window radius R is set to 20 m, and the sliding step size d is set to 10 m.

3.3.2. Results Analysis

Using crowdsourced trajectory data from Shenzhen, nine overpasses were selected for road network construction testing, covering cloverleaf, trumpet, turbine, and combination types [38]. The results of the road network construction test are shown in Figure 13. By comparing the extracted road interchange network with the corresponding area on the OSM map, we observe that the overall completeness of the road interchange network extracted by our method is high. The network contains no redundant segments and clearly distinguishes between spatially adjacent reverse-parallel roads. The road centerline is continuous, smooth, and closely aligned with the actual road, generally positioned at the center of the real road. Additionally, we successfully extracted nearly all road diverging and merging locations. In the extracted network, orange nodes represent divergence points, while rosy red nodes indicate convergence points, which align with the merging and diverging patterns observed in vehicle travel. The topological connections between road segments are accurate, and no false nodes are generated at separated road overlaps. However, due to the inherent ambiguity in the representation of road diverging and merging locations, the extracted node positions are slightly advanced or delayed compared to the true values. Furthermore, due to the heterogeneity of the trajectory data, some regions with sparse or missing data show road omissions.

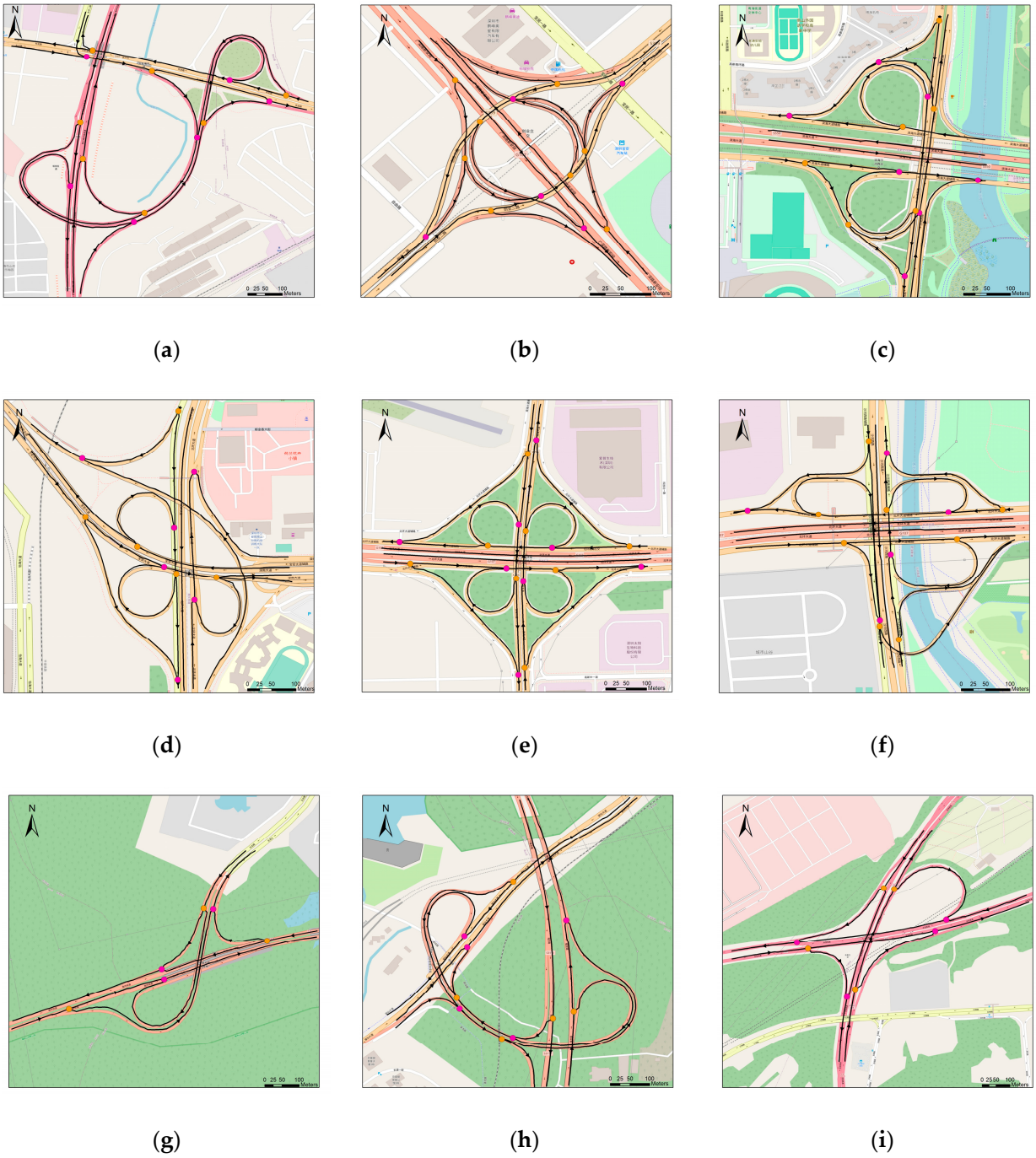


Figure 13. Results from road interchange network construction. (**a**) Dafapu; (**b**) Chuangye; (**c**) Binhai; (**d**) Baoan; (**e**) Beihuan; (**f**) Shahexi; (**g**) Yulong; (**h**) Liuxian; (**i**) Shuiyan.

The accuracy evaluation results for the nine road interchange networks extracted using the method proposed in this paper are shown in Table 2. The GEO accuracy of the road networks for eight of the overpasses in the study area exceeds 92%, with the GEO accuracy of the Shahaxi Overpass slightly lower at 89.34%. The lower GEO accuracy in this case is likely due to the presence of a collector-distributor road with a wider pavement. Because drivers tend to favor one side of the road, the extracted road segments show a consistent offset, failing to match the true road structure. Additionally, the recall rate for all study areas is above 75%, while the F_1-value and TOPO $-$ F_1 score are above 80%. The overall mean GEO precision reaches 95.26%, and the overall mean TOPO $-$ F_1 score is 90.06%. The recall rate is lower than the other metrics, mainly due to two reasons: First,

some overpasses contain auxiliary roads that are adjacent to the main road. Separating the trajectories passing through both roads is challenging, resulting in the omission of auxiliary roads in the extracted network. Second, the use of fixed thresholds to detect road transition nodes causes offsets between the extracted nodes and their true locations. The overpasses have complex diversion structures with varying angles, curvatures, and lengths. The fixed thresholds for extracting diversion points often fail to match the actual locations, leading to inaccuracies. These factors contribute to the lower recall rate compared to the other evaluation metrics.

Table 2. Accuracy evaluation of the results of the proposed method.

Name of the Road Interchange	Precision	Recall	GEO$-F_1$ Score	TOPO$-F_1$ Score
Dafapu	0.9786	0.8779	0.9255	0.9242
Chuangye	0.9686	0.8675	0.9153	0.9202
Binhai	0.9225	0.7545	0.8301	0.8482
Baoan	0.9600	0.8098	0.8785	0.8774
Beihuan	0.9796	0.9246	0.9513	0.9608
Shahexi	0.8934	0.8778	0.8855	0.8786
Yulong	0.9509	0.8479	0.8964	0.8619
Liuxian	0.9565	0.8826	0.9181	0.9360
Shuiyan	0.9633	0.8856	0.9228	0.8986

3.3.3. Comparison with Other Methods

In this section, we present evaluations of the proposed method and compare it with state-of-the-art (SOTA) algorithms. To ensure representative and relevant comparison, we specifically select baseline methods belonging to the two categories identified as most representative for generating road interchange structures: clustering-based methods and trace-merging methods. Consequently, we compare our proposed road interchange network generation algorithm with Kharita (representing clustering-based methods) and RoadRunner (representing trace-merging methods). We did not include intersection-linking methods (e.g., Wang et al. [20]) as baselines because they fundamentally struggle to distinguish adjacent lanes with identical headings but differing elevations at interchanges, often merging clustered lanes into single segments and generating erroneous intersections where roads cross without physical connectivity. Similarly, entry/exit points-based methods (e.g., Tang et al. [14], Deng et al. [16]) were excluded as they lack explicit mechanisms to extract shared subsequences or topological relationships from turning paths, resulting in incomplete navigation-critical connections that are crucial at interchanges. The baselines are briefly described below:

- Kharita [30]: This algorithm extracts the road network in two phases: node extraction through clustering and edge extraction by connecting nodes based on consecutive trajectory points.
- RoadRunner [19]: This algorithm develops a two-step approach that traces new road segments at path bifurcations, and then merges these segments by aligning the future trajectory flow.

We select representative interchanges, including the Beihuan, Chuangye, and Dafapu interchanges (i.e., cloverleaf, turbo, and trumpet types, respectively), to validate the effectiveness of the proposed method. Table 3 presents the performance results of the proposed method in comparison with SOTA algorithms, evaluated using the GEO and TOPO metrics. The proposed method demonstrates excellent performance, achieving the highest F_1 scores in both GEO and TOPO metrics within the test areas. Specifically, the F_1 scores of the pro-

posed method exceed the best comparative method by 21.7% and 19.9% in GEO and TOPO, respectively, for the Beihuan interchange; 16.5% and 15.9% for the Chuangye interchange; and 1.7% and 0.6% for the Dafapu interchange.

Table 3. Performance comparison of the proposed method and SOTA methods.

Method	Beihuan				Chuangye				Dafapu			
	Prec.	**Rec.**	**GEO** $-F_1$	**TOPO** $-F_1$	**Prec.**	**Rec.**	**GEO** $-F_1$	**TOPO** $-F_1$	**Prec.**	**Rec.**	**GEO**$-F_1$	**TOPO**$-F_1$
Kharita	0.097	0.830	0.174	0.185	0.077	0.778	0.140	0.140	0.124	0.894	0.219	0.213
Road Runner	0.607	0.929	0.734	0.761	0.686	0.827	0.750	0.761	0.896	0.922	0.908	0.918
Our method	0.979	0.924	0.951	0.960	0.968	0.867	0.915	0.920	0.978	0.877	0.925	0.924

Figure 14 illustrates the visual comparison between our method and SOTA algorithms for three representative overpasses, with red arrows highlighting pseudo-road connections. Our approach demonstrates robust adaptability to diverse complex overpass structures. As shown in Figure 13b, the RoadRunner method achieves high geometric completeness in capturing overpass morphology but introduces numerous short pseudo-road connections between densely clustered parallel roads. Furthermore, RoadRunner exhibits limited precision in identifying transition nodes, leading to redundant branching caused by trajectory flow divergence. While merging roads based on trajectory flow trend matching partially mitigates erroneous merging of adjacent roads, it fails to resolve redundant branches arising from multiple divergences in bifurcation zones.

The Kharita method preserves the overall skeleton of road interchanges but generates excessive triangular pseudo-road connections between parallel roads. Although its recall rate is high, the geometric accuracy of extracted roads remains suboptimal. This limitation stems from three factors: (1) the distance metric penalizes directional mismatches, causing points on curved segments to appear distant, leading the algorithm to incorrectly connect them with direct edges instead of representing curvature with intermediate nodes, and (2) the method relies on local trajectory point connectivity without considering long-term trajectory continuity, which results in difficulty distinguishing between roads that are physically close but topologically separate, such as parallel or stacked roads, and (3) GPS noise or sampling rate discrepancies can cause trajectories to "jump" between adjacent parallel roads, resulting in erroneous edges between cluster centroids during edge assignment. Consequently, Kharita struggles to distinguish parallel or stacked roads and produces erroneous topological connections among redundant nodes.

In contrast, our method employs a divide-and-conquer strategy by decomposing road interchange networks into subnetworks. Forward and reverse tracking mechanisms isolate neighboring parallel road structures, while trajectory continuity constraints prevent overlapping errors. Node relationships are dynamically validated during tracing, prioritizing local point features and implementing layer-wise verification to ensure reliability. By prioritizing high-grade roads in parameter design of tracking window size, the method ensures geometric consistency between primary roads and ramps, enabling seamless extraction of structures with varying widths. These innovations collectively yield superior extraction accuracy and topological correctness compared to RoadRunner and Kharita.

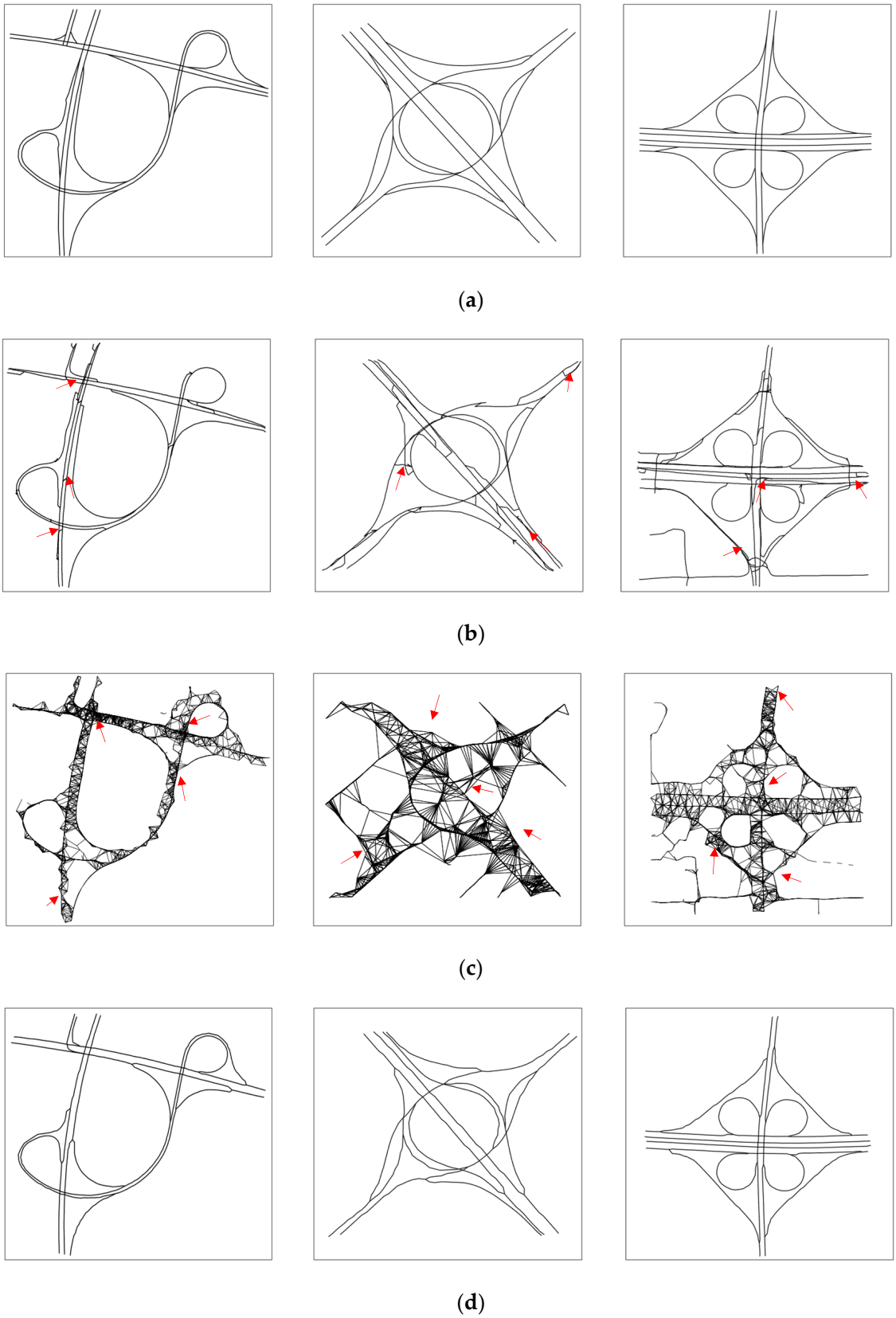


Figure 14. Comparison of road interchange networks generated by our method with SOTA methods. (**a**) OSM; (**b**) RoadRunner; (**c**) Kharita; (**d**) Proposed method.

3.3.4. Sensitivity Analysis

To evaluate how trajectory positioning accuracy affects the algorithm's performance, different levels of Gaussian noise were introduced to the trajectory coordinates. The performance was evaluated at Liuxian interchange, as shown in Figure 15. The results demonstrate that increased Gaussian noise in trajectory coordinates substantially degrades algorithm performance: both recall and F-score exhibit progressive decline as positioning errors grow from 0 to 9 m. Specifically, recall drops from 88.2% to 39.5%, while F-score decreases from approximately 91.8% to 55.6% over this noise range. The decline occurs because trajectory point drift intensifies with higher noise levels, causing points to deviate from the road geometry. Consequently, the algorithm fails to detect trajectory diversion patterns, ultimately leading to omission of interchange branching structures.

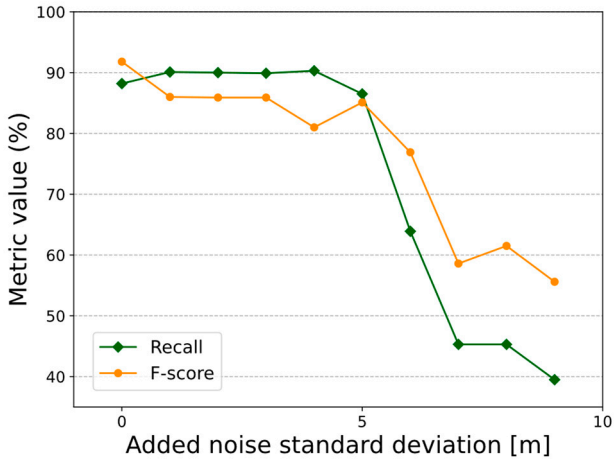


Figure 15. Performance under different levels of added noise.

4. Conclusions

In this paper, we propose a novel algorithm for generating highly accurate, routable road network graphs for highway interchanges. We classify the interference in the study area into four types: circuitous, non-vehicle, low-density, and low-precision trajectories. A high-confidence trajectory screening model is developed, integrating the trajectory morphology index, angular differences, motion features, and neighborhood density. This model forms the basis for the precise extraction of the road interchange network. Next, we introduce a subnetwork tracking algorithm that incorporates both trajectory continuity and local directional information. This method uses overall trajectory continuity to separate multi-level trajectories, enhances the accuracy of road bifurcation identification, and minimizes the risk of generating pseudo-topology at road intersections. Potential transition nodes are verified through pre-tracking to confirm their authenticity. To address network redundancy caused by multiple trajectory diversions at road bifurcations, redundant branches are removed during subnetwork tracking. Finally, a two-stage fusion strategy combines forward tracking with convergence nodes from reverse tracking to create a geometrically and topologically complete road interchange network.

Existing studies focus on road-level networks of typical road structures or 3D road information for highway interchanges but struggle to generate high-accuracy maps for complex interchanges due to underutilized trajectory continuity and flow information. To address this, we propose a divide-and-conquer approach that layers multi-level trajectories using shared entry/exit points and tracks bifurcations via long-trajectory continuity within subnetworks. Unlike conventional methods relying on local distance and heading angle metrics for segment merging, our approach leverages forward movement distribution and subnetwork fusion from multiple entry/exit points. This strategy simplifies the recon-

struction of intricate geometries, eliminates pseudo-intersections by accurately identifying nodes, and distinguishes dense parallel roads. Experimental results demonstrate superior performance in geometric completeness and topological accuracy compared to existing methods. The results indicate the effectiveness of our proposed method in extracting road interchange networks under complex scenarios. Our method generates road networks with higher structural integrity and fewer fragments. The proposed method offers an innovative and practical solution for the automatic generation of road interchange networks, directly supporting efficient navigation systems.

Future work should address two key issues. First, the identification of entry/exit points for automatic road network construction could be improved by using road-level maps of non-interchange areas, combined with road geometry and direction derived from remote sensing images. Second, we will extend this algorithm to evaluate its robustness when applied to interchanges of varying complexity across diverse urban environments.

Author Contributions: Conceptualization, Fengwei Jiao, Longgang Xiang and Yuanyuan Deng; Data curation, Fengwei Jiao; Formal analysis, Fengwei Jiao and Yuanyuan Deng; Funding acquisition, Longgang Xiang; Investigation, Fengwei Jiao; Methodology, Fengwei Jiao; Project administration, Longgang Xiang; Resources, Longgang Xiang; Software, Fengwei Jiao and Yuanyuan Deng; Supervision, Longgang Xiang; Validation, Fengwei Jiao, Longgang Xiang and Yuanyuan Deng; Visualization, Fengwei Jiao; Writing—original draft, Fengwei Jiao; Writing—review & editing, Fengwei Jiao and Longgang Xiang All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (grant numbers 42471460).

Data Availability Statement: The data presented in this study are not publicly available due to commercial restrictions but may be made available from the corresponding author on reasonable request under project collaboration agreements.

Acknowledgments: We express our thanks to the anonymous reviewer for constructive comments.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Nugmanova, A.; Arndt, W.-H.; Hossain, M.A.; Kim, J.R. Effectiveness of Ring Roads in Reducing Traffic Congestion in Cities for Long Run: Big Almaty Ring Road Case Study. *Sustainability* **2019**, *11*, 4973. [CrossRef]
2. Afrin, T.; Yodo, N. A Survey of Road Traffic Congestion Measures towards a Sustainable and Resilient Transportation System. *Sustainability* **2020**, *12*, 4660. [CrossRef]
3. Gu, X.; Abdel-Aty, M.; Xiang, Q.; Cai, Q.; Yuan, J. Utilizing UAV Video Data for In-Depth Analysis of Drivers' Crash Risk at Interchange Merging Areas. *Accid. Anal. Prev.* **2019**, *123*, 159–169. [CrossRef] [PubMed]
4. Ding, Y.; Yao, Y.; Yang, L.; Zhao, X. Study on Safety Features of Road and Traffic Facilities at the Entrance of Interchange Area. In Proceedings of the 2019 5th International Conference on Transportation Information and Safety (ICTIS), Liverpool, UK, 14–17 July 2019; pp. 1469–1475.
5. Jalayer, M.; Zhou, H.; Zhang, B. Evaluation of Navigation Performances of GPS Devices near Interchange Area Pertaining to Wrong-Way Driving. *J. Traffic Transp. Eng. Engl. Ed.* **2016**, *3*, 593–601. [CrossRef]
6. Ekpenyong, F.; Palmer-Brown, D.; Brimicombe, A. Extracting Road Information from Recorded GPS Data Using Snap-Drift Neural Network. *Neurocomputing* **2009**, *73*, 24–36. [CrossRef]
7. Hu, J.; Razdan, A.; Femiani, J.C.; Cui, M.; Wonka, P. Road Network Extraction and Intersection Detection From Aerial Images by Tracking Road Footprints. *IEEE Trans. Geosci. Remote Sens.* **2007**, *45*, 4144–4157. [CrossRef]
8. Zhang, J.; Hu, X.; Wei, Y.; Zhang, L. Road Topology Extraction From Satellite Imagery by Joint Learning of Nodes and Their Connectivity. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–13. [CrossRef]
9. Zhu, Q.; Zhang, Y.; Wang, L.; Zhong, Y.; Guan, Q.; Lu, X.; Zhang, L.; Li, D. A Global Context-Aware and Batch-Independent Network for Road Extraction from VHR Satellite Imagery. *ISPRS J. Photogramm. Remote Sens.* **2021**, *175*, 353–365. [CrossRef]
10. Li, Y.; Xiang, L.; Zhang, C.; Jiao, F.; Wu, C. A Guided Deep Learning Approach for Joint Road Extraction and Intersection Detection from RS Images and Taxi Trajectories. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 8008–8018. [CrossRef]

11. Wang, J.; Wang, C.; Song, X.; Raghavan, V. Automatic Intersection and Traffic Rule Detection by Mining Motor-Vehicle GPS Trajectories. *Comput. Environ. Urban Syst.* **2017**, *64*, 19–29. [CrossRef]
12. Gao, L.; Wei, L.; Yang, J.; Li, J. Automatic Intersection Extraction Method for Urban Road Networks Based on Trajectory Intersection Points. *Appl. Sci.* **2022**, *12*, 5873. [CrossRef]
13. Du, J.; Liu, X.; Meng, C. Road Intersection Extraction Based on Low-Frequency Vehicle Trajectory Data. *Sustainability* **2023**, *15*, 14299. [CrossRef]
14. Tang, J.; Deng, M.; Huang, J.; Liu, H. A Novel Method for Road Intersection Construction From Vehicle Trajectory Data. *IEEE Access* **2019**, *7*, 95065–95074. [CrossRef]
15. Huang, J.; Deng, M.; Zhang, Y.; Liu, H. Complex road intersection modelling based on low-frequency GPS track data. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *XLII-2-W7*, 23–28. [CrossRef]
16. Deng, M.; Huang, J.; Zhang, Y.; Liu, H.; Tang, L.; Tang, J.; Yang, X. Generating Urban Road Intersection Models from Low-Frequency GPS Trajectory Data. *Int. J. Geogr. Inf. Sci.* **2018**, *32*, 2337–2361. [CrossRef]
17. Stanojevic, R.; Abbar, S.; Thirumuruganathan, S.; Chawla, S.; Filali, F.; Aleimat, A. Robust Road Map Inference through Network Alignment of Trajectories. In Proceedings of the 2018 SIAM International Conference on Data Mining (SDM), San Diego, CA, USA, 3–5 May 2018; Proceedings. Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2018; pp. 135–143.
18. Cao, L.; Krumm, J. From GPS Traces to a Routable Road Map. In Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Seattle, WA, USA, 4–6 November 2009; Association for Computing Machinery: New York, NY, USA, 2009; pp. 3–12.
19. He, S.; Bastani, F.; Abbar, S.; Alizadeh, M.; Balakrishnan, H.; Chawla, S.; Madden, S. RoadRunner: Improving the Precision of Road Network Inference from GPS Trajectories. In Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Seattle, WA, USA, 6–9 November 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 3–12.
20. Wang, J.; Rui, X.; Song, X.; Tan, X.; Wang, C.; Raghavan, V. A Novel Approach for Generating Routable Road Maps from Vehicle GPS Traces. *Int. J. Geogr. Inf. Sci.* **2015**, *29*, 69–91. [CrossRef]
21. Sun, T.; Di, Z.; Che, P.; Liu, C.; Wang, Y. Leveraging Crowdsourced GPS Data for Road Extraction From Aerial Imagery. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 7501–7510.
22. Qin, J.; Yang, W.; Wu, T.; He, B.; Xiang, L. Incremental Road Network Update Method with Trajectory Data and UAV Remote Sensing Imagery. *ISPRS Int. J. Geo-Inf.* **2022**, *11*, 502. [CrossRef]
23. Cheng, L.; Wu, Y.; Wang, Y.; Zhong, L.; Chen, Y.; Li, M. Three-Dimensional Reconstruction of Large Multilayer Interchange Bridge Using Airborne LiDAR Data. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 691–708. [CrossRef]
24. Zhou, Y.; Huang, R.; Jiang, T.; Dong, Z.; Yang, B. Highway Alignments Extraction and 3D Modeling from Airborne Laser Scanning Point Clouds. *Int. J. Appl. Earth Obs. Geoinf.* **2021**, *102*, 102429. [CrossRef]
25. Li, X.; Liu, C.; Wang, Z.; Xie, X.; Li, D.; Xu, L. Airborne LiDAR: State-of-the-Art of System Design, Technology and Application. *Meas. Sci. Technol.* **2020**, *32*, 032002. [CrossRef]
26. Ren, C.; Tang, L.; Yang, X.; Long, J. Mapping Grade-Separated Junctions in Detail Using Crowdsourced Trajectory Data. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 5552–5561. [CrossRef]
27. Yang, X.; Yang, M.; Cao, Y.; Ren, C.; Zhang, F.; Tang, L. Three-Dimensional Structure Determination of Grade-Separated Road Intersections from Crowdsourced Trajectories. *Int. J. Appl. Earth Obs. Geoinf.* **2023**, *125*, 103598. [CrossRef]
28. Phuengsaeng, P.; Sutthisangiam, N. Automatic Corrective Elevation of GPS Trajectory in Urban Area. In Proceedings of the 2018 18th International Symposium on Communications and Information Technologies (ISCIT), Bangkok, Thailand, 26–29 September 2018; pp. 458–462.
29. Basiri, A.; Haklay, M.; Foody, G.; Mooney, P. Crowdsourced Geospatial Data Quality: Challenges and Future Directions. *Int. J. Geogr. Inf. Sci.* **2019**, *33*, 1588–1593. [CrossRef]
30. Zhang, X.; Yang, Z.; Wu, C.; Sun, W.; Liu, Y.; Xing, K. Robust Trajectory Estimation for Crowdsourcing-Based Mobile Applications. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 1876–1885. [CrossRef]
31. Li, D.; Li, J.; Li, J. Road Network Extraction from Low-Frequency Trajectories Based on a Road Structure-Aware Filter. *ISPRS Int. J. Geo-Inf.* **2019**, *8*, 374. [CrossRef]
32. Scott, D.W. Multivariate Density Estimation and Visualization. In *Handbook of Computational Statistics: Concepts and Methods*; Gentle, J.E., Härdle, W.K., Mori, Y., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 549–569. ISBN 978-3-642-21551-3.
33. Bashtannyk, D.M.; Hyndman, R.J. Bandwidth Selection for Kernel Conditional Density Estimation. *Comput. Stat. Data Anal.* **2001**, *36*, 279–298. [CrossRef]
34. Yan, X.; Pei, T.; Shu, H.; Song, C.; Wu, M.; Fang, Z.; Chen, J. Spatiotemporal Flow L-Function: A New Method for Identifying Spatiotemporal Clusters in Geographical Flow Data. *Int. J. Geogr. Inf. Sci.* **2023**, *37*, 1615–1639. [CrossRef]

35. Pei, T.; Yan, X.; Shu, H.; Wang, X.; Chen, X.; Guo, S.; Jiang, J.; Song, C.; Chen, J. Geographical Flows: A Fresh Perspective on Quantifying Urban Function. *Urban Inform.* **2024**, *3*, 14. [CrossRef]
36. Arthur, D.; Vassilvitskii, S. K-Means++: The Advantages of Careful Seeding. In Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, New Orleans, LA, USA, 7–9 January 2007; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2007; pp. 1027–1035.
37. Biagioni, J.; Eriksson, J. Inferring Road Maps from Global Positioning System Traces: Survey and Comparative Evaluation. *Transp. Res. Rec.* **2012**, *2291*, 61–71. [CrossRef]
38. CD 122—Geometric Design of Grade Separated Junctions. Available online: https://www.standardsforhighways.co.uk/search/3ab9ef31-9880-4e8e-a7eb-f3d218e74ffd (accessed on 14 March 2025).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.