

Article

# Using Probe Counts to Provide High-Resolution Detector Data for a Microscopic Traffic Simulation

Tobias Veihelmann <sup>\*</sup>, Victor Shatov , Maximilian Lübke  and Norman Franchi 

Chair of Electrical Smart City Systems, Friedrich-Alexander-Universität Erlangen-Nürnberg, Cauerstraße 7, 91058 Erlangen, Germany; victor.shatov@fau.de (V.S.); maximilian.luebke@fau.de (M.L.); norman.franchi@fau.de (N.F.)

\* Correspondence: tobi.veihelmann@fau.de

**Abstract:** Microscopic traffic simulations have become increasingly important for research targeting connected vehicles. They are especially appreciated for enabling investigations targeting large areas, which would be practically impossible or too expensive in the real world. However, such large-scale simulation scenarios often lack validation with real-world measurements since these data are often not available. To overcome this issue, this work integrates probe counts from floating car data as reference counts to model a large-scale microscopic traffic scenario with high-resolution detector data. To integrate the frequent probe counts, a road network matching is required. Thus, a novel road network matching method based on a decision tree classifier is proposed. The classifier automatically adjusts its cosine similarity and Hausdorff distance-based similarity metrics to match the network's requirements. The approach performs well with an  $F_1$ -score of 95.6%. However, post-processing steps are required to produce a sufficiently consistent detector dataset for the subsequent traffic simulation. The finally modeled traffic shows a good agreement of 95.1% with upscaled probe counts and no unrealistic traffic jams, teleports, or collisions in the simulation. We conclude that probe counts can lead to consistent traffic simulations and, especially with increasing and consistent penetration rates in the future, help to accurately model large-scale microscopic traffic simulations.

**Keywords:** floating car data; microscopic traffic simulation; probe counts; road network matching; SUMO



**Citation:** Veihelmann, T.; Shatov, V.; Lübke, M.; Franchi, N. Using Probe Counts to Provide High-Resolution Detector Data for a Microscopic Traffic Simulation. *Vehicles* **2024**, *6*, 747–764. <https://doi.org/10.3390/vehicles6020035>

Academic Editor: Elzbieta Macioszek

Received: 26 March 2024

Revised: 26 April 2024

Accepted: 26 April 2024

Published: 28 April 2024



**Copyright:** © 2024 by the authors. 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

Microscopic traffic simulation has become increasingly interesting for the telecommunication networking community [1]. Vehicular ad hoc networks (VANETs), an enabling technology for autonomous driving, road safety, and hazard information services [2], can hardly be tested in the real world since test cases are barely possible to reproduce, large-scale tests are extremely costly, and failures would risk people's physical integrity [3]. Similarly, the additional sensors required for autonomous driving may unfold their total potential only if their information is shared with other vehicles—if they perform collaborative sensing. However, there are some challenges associated with these upcoming technologies: depending on the high relative speed of the vehicles, the concerned V2V communication is complicated [4], and a microscopic traffic simulation is needed as the basis for a network simulator (e.g., [5]). Thus, traffic simulations are an essential tool for researching and developing novel technologies. Nonetheless, the need for accurate and detailed microscopic traffic simulation also poses some challenges. Generally, a traffic simulation requires three kinds of inputs: network data, an additional traffic infrastructure, and traffic demand [6]. Demand modeling is a complex task usually realized with an origin–destination (O-D) matrix [7]. For an area divided into different traffic analysis zones, an O-D matrix contains information on how many vehicles move from one zone to another [8]. Usually, this information is derived from surveys and census data [7]. Still, validating a microscopic traffic simulation resulting from the demographic input data is

difficult [9] since real-world measurements with a high resolution are hard to retrieve and have limited availability [10].

More recent developments have helped alleviate this problem. Advanced communications technologies have enabled the rise of floating car data (FCD), which can provide information on speed, travel time, and route choice proportions [11], so that also traffic flow and traffic congestion information can be derived [12]. With higher penetration rates, more and more FCD products have become available, including traffic density information based on probe counts (e.g., [13]). These offer a much higher spatial resolution than traditional detector data from permanent automatic counting stations. Yet, their penetration rate is limited and often unknown.

This work shows with a case study of a *SUMO* simulation, as introduced in [6], how probe count data and traffic counts from permanent automatic counting stations can jointly serve as input for a large-scale microscopic traffic simulation covering an area of 930 km<sup>2</sup>. The approach consists of three steps: First, the probe counts are extracted, alongside the data from counting stations and upscaled accordingly. Second, we propose a simple, easily applicable, and well-generalizable road-network-matching approach to map the street segments from the network with probe counts to the street segments of the network for the traffic simulation. Third, we provide a qualitative and quantitative evaluation of the microscopic simulation results to understand the consistency of road network matching and the floating car data.

## 2. Related Works

Relevant research branches for this paper are existing works on large-scale microscopic traffic simulations and road network matching approaches. To compare this work to existing solutions, we searched for microscopic simulation scenarios modeling areas greater than 100 km<sup>2</sup>. Because our approach differs in the demand source, which is probe data, it also requires another way to integrate this data source into the simulation. With the demand source given as probe counts for segments in a street network, that network (or at least the street segments with probe counts) needs to be matched with the OpenStreetMap (OSM) road segments serving as a basis for the *SUMO* simulation. The problem is that, although we have the precise coordinates of street segments in both maps, the street segments belonging to the same street often significantly differ in their lengths and, sometimes, also in their shape and exact location. Thus, we also reviewed current road-network-matching methods.

### 2.1. Large-Scale Microscopic Traffic Simulations

Previous research has already provided some examples of large-scale microscopic traffic scenarios, which are listed in Table 1. Except Bieker et al. [14], who used *Vissim*, all simulations were run in *SUMO*. While these have in common that O-D matrices are used for demand modeling, the inputs to create the O-D matrices differ. Uppoor et al. [1], Lobo et al. [3], and Codecá et al. [15] employed demographic data for their city-scale scenarios. Uppoor et al. [1] relied on data on Cologne's population's home locations and socio-demographic characteristics, the city's points of interest, and the inhabitants' time-use patterns, derived from reports from more than 7000 households. Rapelli et al. [16] received an O-D matrix from a traffic operation center whose demographic data resulted from metropolitan surveys. In contrast, Codecá et al. [15] fed more general statistics on the population (e.g., age distribution) into *SUMO*'s *ACTIVITYGEN* tool to model traffic demand. This procedure resembles Lobo et al.'s [3] approach, which collected information on inhabitants, workplaces and locations, employment, vehicles, and householders and used *ACTIVITYGEN* for the final demand modeling.

Other studies leveraged traffic count data from traffic authorities. Those may result from traffic cameras, induction loops, or other traffic sensors and are often enriched by floating car data. Ketabi et al. [17] based their O-D matrix on information from traffic cameras. A simulation without an O-D matrix was performed by Bieker et al. [14]. Their *Vissim* simulation followed the flow measurements from 636 detectors in the City of Bologna.

Gonzalez-Delicado et al. [10] and Guastella et al. [9] realized similar approaches with *SUMO*. Gonzalez-Delicado et al. [10] aligned their *SUMO* simulation's traffic flows with measurements from 99 induction loops with *Cadyts* [18]. Similarly, Guastella et al. [9] created an allowlist of trips with *RANDOMTRIPS*. They applied *ROUTESAMPLER* and traffic count data to model routes and demand. In addition, they evaluated how accurately collaborative IoT agents equipped with sensors and access to historical data could estimate missing detector data.

**Table 1.** Large-scale microscopic traffic simulations.

City (Year)	Area in km <sup>2</sup>	N Trips	Running Vehicles	Period	Demand Source
Cologne [1] (2014)	400	700,000	up to 15,000	24 h	demographic
Bologna [14] (2015)	N/A	N/A	N/A	1 h	detectors
Luxembourg [15] (2017)	156	N/A	N/A	24 h	demographic
London [17] (2017)	N/A	N/A	N/A	2 h	detectors
Shanghai [19] (2018)	800	N/A	N/A	N/A	taxi fleet
Alicante-Murcia [10] (2021)	N/A <sup>1</sup>	N/A	N/A	9 days	detectors
Turin [16] (2022)	600	2,202,814	up to 20,000	24 h	demographic
Ingolstadt [3] (2022)	N/A	N/A	N/A	24 h	demographic
Brussels [9] (2023)	150 <sup>2</sup>	N/A	N/A	16 h	detectors, HybridIoT
This work	930	18,874	about 4300	1 h	detectors, probe counts

The term "detector" embraces traffic cameras and induction loops in this table. "N Trips" refers to the total number of trips from vehicles in the respective simulation scenario. "Period" denotes the duration of the modeled simulation scenario. <sup>1</sup> Not applicable since the scenario is a freeway scenario. The length of the freeway segment was given as 97 km. <sup>2</sup> No number given by the authors. Estimation based on inspecting the map images they provided.

What differentiates this work from these earlier approaches is that constraints with a very high spatial resolution, i.e., a number for almost every street segment in the simulation, are available through floating car data. While we used this information to model realistic traffic demands, the potential may be even bigger for addressing the problems stressed by [10]: the same products and procedures can be used to validate simulated road occupancy and traffic speeds. This paper proposes a method to include probe counts from floating car data into OpenStreetMap, serving as a basis for a microscopic traffic simulation.

## 2.2. Road Matching

An essential step for integrating probe counts linked to street segments from private companies with OpenStreetMap data is to match the respective road networks. Relevant works are subsumed with the terms road matching, road network matching, or, more generally, spatial conflation. Yu and Liu [20] deliberately integrated human interaction into their matching method. After a similarity-based preliminary matching, potential matches with high uncertainty were passed to humans for manual inspection. The matching parameters were updated based on the subsequently updated pool of labeled samples. Wu et al. [21] formulated the road matching as a minimal cost flow problem [22], minimizing the difference in the features between the two given datasets. After applying the optimization algorithm with only buffer overlap of candidate pairs, they introduced the features angle,

flow, and shortest-path-based constraint to reduce ambiguities. Comparing their method to other approaches applied to the same datasets revealed that their method significantly outperformed previous approaches. Zuo et al. [23] and Guo et al. [24] emphasized the benefit of integrating strokes for road matching. A stroke is the line resulting from concatenating neighboring road segments. Guo et al.'s four-step approach included data preprocessing, entire stroke matching, partial stroke matching, and detection and matching of roundabouts. Hacar and Gökğöz [25] summed up similarity scores from various similarity metrics to determine matching road segments.

Since the authors applied their approaches to different road networks and there were network-specific characteristics such as how cellular the network is, the comparability of the approaches' performances is limited. Even if an approach is compared to other methods on the same network (see [21]), how the comparison would look with other road networks remains unclear. This work proposes an approach that is easily applicable to all types of networks since, after passing a labeled and representative subset, the machine learning algorithm will select features for similarity calculation and the respective features and how they fit best for the networks to be matched.

### 3. Methodology

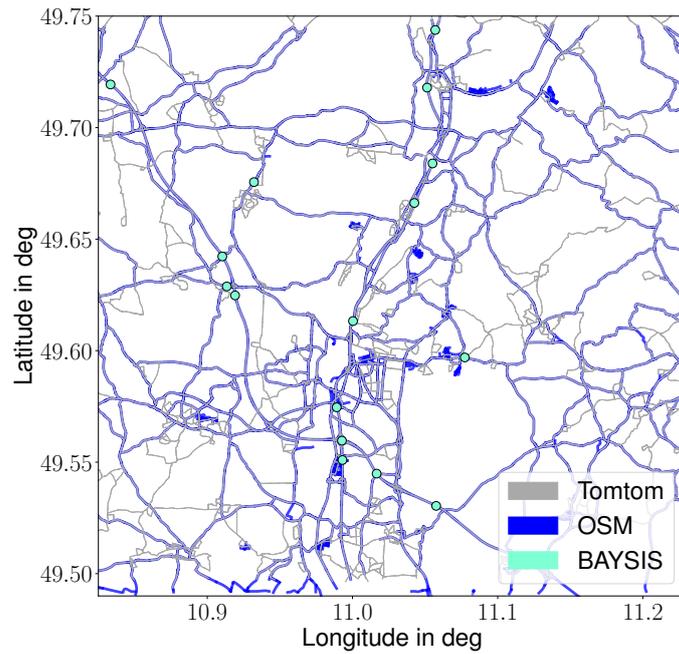
#### 3.1. Overview

The overall idea is to use a high-resolution traffic count dataset to obtain very accurate constraints for traffic generation. The Tomtom traffic stats [13] offer a dataset with probe numbers for each street segment. However, the probe counts do not reflect the total number of vehicles passing specific street segments; only a share is available to Tomtom. Thus, a "true" count from a traffic authority is still required, with the disadvantage of a significantly lower spatial resolution. We, thus, decided to upscale data from Tomtom at each street segment with a constant factor derived from BAYSIS [26] traffic counts from permanent automatic counting stations, which are displayed in Figure 1. Subsequently, the resulting total traffic count numbers for each street segment shall be input for a traffic simulation. We realized this in *SUMO* by creating an edge data file for the *ROUTESAMPLER*. An edge data file is a file specifying the number of cars entering an edge in the simulation in a given period. The following XML code shows an exemplary edge data file structure with an extract from this work's edge data file:

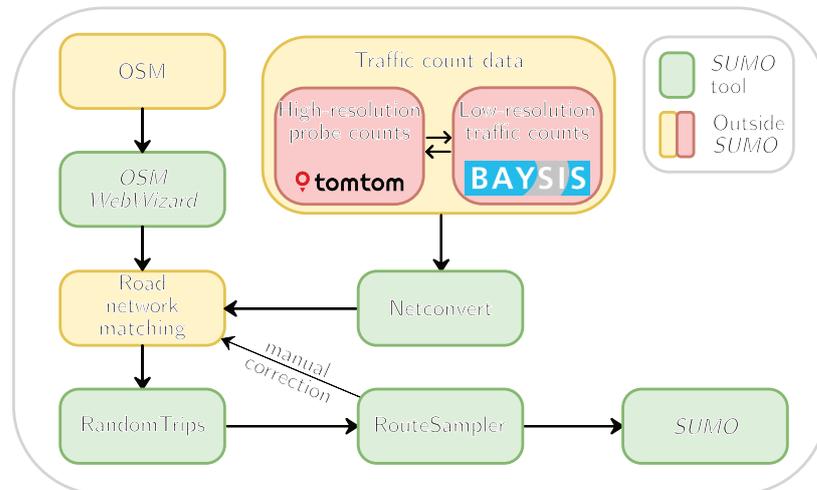
```
<data>
  <interval id="1" begin="0" end="3599">
    <edge id="-1009301209" entered="20"/>
    <edge id="-1010139007" entered="215"/>
    <edge id="-1010774407" entered="145"/>
    ...
  </interval>
  ...
</data>
```

In this paper's understanding, an edge denotes a directed street segment with a defined start and end node or junction, respectively. An edge can include multiple parallel lanes.

A problem to be solved before the edge data file can be created is that traffic counts exist for street segments provided by Tomtom, which are missing information needed as valid *SUMO* input, such as traffic signals or lane numbers. Thus, we used OpenStreetMap data retrieved via *OSMWebWizard* to create a map for *SUMO*. As a result, the OSM street segments, i.e., edges, required a mapping to Tomtom street segments. A custom road-network-matching method yielded the mapping. Figure 2 displays the interaction of all methods.



**Figure 1.** The road network from different perspectives. The turquoise dots show where the BAYSIS [26] counting stations are located. The blue and gray lines display the road network representations after selecting the relevant road categories for the Tomtom map from [13] and OpenStreetMap (OSM).



**Figure 2.** The methods in this paper: Green boxes represent SUMO functions, red boxes probe or traffic counts, and the yellow boxes data sources or applications outside SUMO.

### 3.2. Road Networks for Simulation

To have both OSM and Tomtom traffic stats data in the same sumolib layout, we created a SUMO map using SUMO netconvert. For OpenStreetMap, we only considered the highway types *motorway*, *trunk*, *primary*, *secondary*, and *tertiary*. The excluded highway types were *unclassified*, i.e., minor roads that serve another purpose than to access properties, and *residential* roads (see [27]). A similar, yet finer Tomtom network layout results from including the functional road classes 0 to 6. Those functional road classes included *motorways*, *freeways*, *major roads* (0); *major roads less important than motorways* (1); *other major roads* (2); *secondary roads* (3); *local connecting roads* (4); *local roads of high importance* (5); and *local roads* (6) (see [28]). *Local roads of minor importance* (7) and *other roads* (8) were excluded. Figure 1 shows the two road networks.

### 3.3. Traffic Count Numbers

The traffic count data for the simulation were from 7 October 2022, a Friday with mixed weather conditions. We took sample counts for one hour from 11:30 a.m. to 12:30 p.m., when the traffic flow between the directions of the highways was roughly even. From the BAYSIS count stations, we took the hourly average from 11:00 a.m. to 1:00 p.m. to extrapolate the Tomtom data. We identified 16 permanent automated count stations, resulting in 32 traffic counts (two directions per count station). For each traffic count, we compared the count with the corresponding Tomtom passenger car probe count(s) of the road section. The median of the 32 factors served as a constant upscaling factor. The distribution of factors suggests that lower Tomtom probe counts (of normal roads as opposed to highways) correspond to a lower proportion of probe counts, i.e., they would require a higher upscaling factor. As our sample size of traffic counts was small and there were only a few count points on ordinary roads (three), we did not have enough data to adequately account for this effect. We accepted the median value for upscaling all road types. However, while our upscaled traffic counts on highways are almost completely correct, vehicles on ordinary roads might be underrepresented in our edge data file.

### 3.4. Road Network Matching

We first mapped the Tomtom street segments to OpenStreetMap street segments to obtain an edge data file. Several approaches exist, and several geometric, semantic, topological, and contextual measures, e.g., strokes, have evolved [24]. While semantic information is available for our maps, its employment is minimal since, in the Tomtom data, highway entrances have the same labels as the highways themselves. At the same time, OpenStreetMap has a separate category for highway entrances, so there is no common differentiation between highways and highway entrances in our OSM and Tomtom data. However, those, in particular, would benefit greatly from this information because the geometric characteristics between a motorway and the connected motorway on-ramp segment can be very similar. Topological measures are also tricky since they are based on node degrees [24], which we cannot correctly retrieve anymore after having excluded less prioritized street types.

Consequently, we decided to only use geometric information, i.e., coordinates, to match Tomtom and OSM street segments. Because the street segments in both data sets offered a wide range of lengths in their street segments, we could not employ a simple distance metric like calculating the distance of the midpoints in both street segments. Also, there was no information on the edge level regarding the direction of a street segment. That information was implicitly given with the ordering of the coordinates defining a road segment. Hence, we derived the necessity of at least two kinds of calculations for an accurate matching of the street segments comprising both road networks: we needed to account for the direction of a street segment and employed a non-trivial distance metric including some kind of maximum distance and shape information to find the corresponding street segments in both data sets. While there have been various similarity and threshold- or score-based approaches to road network matching (e.g., [25,29]), their accuracies vary on different data sets. To have a method that automatically finds optimal threshold values for a given data set, we employed a machine learning (ML) method to be trained and evaluated on a small, manually labeled sample network and then applied to the target network. We displayed nearby Tomtom and OSM street segments for labeling using a modified version of the labeling tool introduced in [30]. As features capturing distance and shape, we selected Fréchet distance [31], Hausdorff distance [31], and modified Hausdorff distance [24]. As solely shape-based metrics, we employed sinuosity similarity [25]. To compare the directions of vectorized line strings, we implemented cosine similarity.

Since multiple metrics targeted the same characteristics (distance, shape, and direction), we assumed redundancy in our features. Thus, we trained the ML model explained in Section 3.4.3 with all features and checked their importance afterward. Only cosine similarity and modified Hausdorff distance remained as selected features.

### 3.4.1. Similarity Metrics

In [32], the Hausdorff distance  $H$  for two point sets  $A = \{a_1, \dots, a_n\}$  and  $B = \{b_1, \dots, b_m\}$  is defined as

$$H(A, B) = \max(h(A, B), h(B, A)), \quad (1)$$

where

$$h(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\|. \quad (2)$$

$\|a - b\|$  represents some norm, for example, the Euclidean norm. The modified Hausdorff distance is designed to be more robust against noise, which is also desired in road-network-matching applications and follows a slightly different definition of  $h(A, B)$  [24], which is

$$h(A, B) = \frac{1}{N} \sum_{a \in A} \min_{b \in B} \|a - b\|, \quad (3)$$

where  $N$  is the number of nodes in  $A$ .

Cosine similarity is a metric that gives the similarity of two vectors. If the two vectors point in the same direction, the cosine similarity is 1; if they point in opposite directions, the cosine similarity is  $-1$ . The formula for calculating the cosine similarity of two vectors  $\mathbf{a}$  and  $\mathbf{b}$  is [33]

$$\text{cosine similarity} = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\| \cdot \|\mathbf{b}\|} = \frac{\sum_{i=1}^n a_i b_i}{\sqrt{\sum_{i=1}^n a_i^2} \sqrt{\sum_{i=1}^n b_i^2}}. \quad (4)$$

Similarity metrics that are not part of the final road-matching model in this paper are presented in Appendix A.

### 3.4.2. Preprocessing

All overlapping Tomtom edges with a buffer of 10 m were considered candidates for each OSM edge. Next, the edges were pruned so that only the line strings' overlapping or parallel parts were considered for further analysis. For those pruned line strings, we calculated the similarity metrics. If there were different numbers of coordinate pairs defining candidate pair line strings, the one with fewer coordinates was interpolated to the same number of coordinates. We only included manually matched Tomtom–OSM pairs with overlap greater than 30% of the shorter edge's length. After this step, 222 non-matches and 2193 matches remained. Since this significant imbalance might interfere with the aim of unbiased ML training, we employed the *imblean* [34] *RandomOverSampler* to train with a balanced dataset.

### 3.4.3. The ML Model

Since we expected straightforward relations between our features, such as “Tomtom edge XYZ matches OSM edge ABC if their cosine similarity is greater than 0.9 and the modified Hausdorff distance is smaller than 10 m”, we considered tree-based classifiers, which represent if–then rules [35], the most natural choice. The specific classifiers of consideration were a single decision tree and a gradient-boosted trees ensemble learner with theoretically better performance, especially if trees grow large (see [36]), but limited explainability due to the high number of trees.

The goal of a decision tree is to perfectly partition the data into the expected classes, which would be in our case 1 for “Tomtom edge XYZ matches OSM edge ABC” or “Tomtom edge XYZ does not match OSM edge ABC”. A decision tree learns by selecting an attribute based on its information gain, which means the expected reduction of entropy [35] until a perfect partitioning is achieved or the tree has grown to a pre-defined maximum depth. The formula of the information gain for an attribute  $A$  (e.g., cosine similarity) relative to a collection of examples  $S$  (e.g., a set of candidate pairs of OSM: Tomtom edge matches with a known ground truth) is defined as

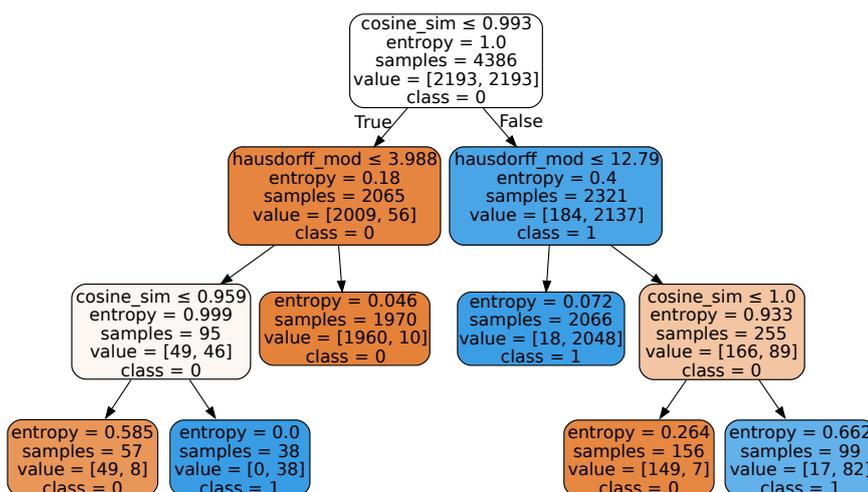
$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{v \in \text{Values}(A)} \frac{|S_v|}{|S|} \text{Entropy}(S_v) \tag{5}$$

with

$$\text{Entropy}(S) = \sum_{i=1}^c -p_i \log_2 p_i, \tag{6}$$

where  $c$  is the number of possible target values for the target attribute (in our case, classes 1 and 0) and  $p_i$  is the proportion of examples in  $S$  belonging to class  $i$  [35]. There is a small extension to this procedure for continuous-valued attributes; still, the attribute with the highest information gain is chosen. However, the rules shall not target specific values, but threshold values. To achieve this, the values for an attribute first are sorted. Then, the gaps between adjacent examples' values with different target classes are considered candidate thresholds. Finally, the highest information gain threshold is selected after calculating the information gain for the candidate thresholds [35].

After some experimentation, an *sklearn* [37] decision tree with a maximum depth of three, min samples split of 10, and entropy criterion served as the classifier. Figure 3 depicts the resulting model. The gradient-boosted trees did not consistently outperform the low-depth decision tree, whereas the decision tree consistently relied on cosine similarity and modified Hausdorff distance as the only features. Five-fold cross-validation evaluates a mean accuracy of 98.2% and an F<sub>1</sub>-score of 98.2% for the decision tree. While the evaluation metrics for the gradient-boosted trees were slightly better, as Table 2 shows, its performance on the subset from the big dataset for the simulation application was slightly worse.



**Figure 3.** Representation of the decision tree model for the road network matching. Nodes with 0 as majority class are colored orange, nodes with 1 as majority class are colored blue. The transparency indicates the impurity in each node. The final model consists of rules based on cosine similarity and modified Hausdorff distance.

**Table 2.** Five-fold cross-validation results for the decision tree and gradient-boosted trees.

	Accuracy	Precision	Recall	F <sub>1</sub> -Score
Decision tree	98.2	97.8	98.7	98.2
Gradient boosting <sup>1</sup>	<b>99.5</b>	<b>100.0</b>	<b>99.0</b>	<b>99.5</b>

<sup>1</sup> The features are modified Hausdorff distance, sinuosity similarity, and cosine similarity. The *sklearn* [37] implementation with the following hyperparameters was employed: *n\_estimators* = 25, *max\_depth* = 3, *min\_samples\_split* = 10, *learning\_rate* = 0.25. The highest score in each column is highlighted in bold.

Thus, we applied the decision tree model to the whole dataset before we scaled up the Tomtom probe counts for the matched Tomtom edges and created an edge data file. When multiple Tomtom edges were mapped to one OSM edge, we assigned the median

probe count of the Tomtom edges to the OSM edge. Further, we calculated for each edge in the edge data file the difference of its count number to its predecessor and successor edge and excluded non-fringe edges where the relative deviation exceeded 35%, and the absolute deviation exceeded 100 vehicles per hour. After this step, 4214 of 5646 counting stations remained.

### 3.5. The SUMO Traffic Simulation

Besides the constraints provided by the edge data file, the *ROUTESAMPLER* requires an allowlist of potential routes before selecting the most appropriate routes for the simulation. Thus, we let *RANDOMTRIPS* generate an allowlist of potential routes. The parameters were set to a fringe factor of 50, meaning routes were 50-times more likely to start with an edge without a predecessor or successor. The random routing factor was 2, meaning completing a route may take at most twice as long as the fastest route with the same origin and destination would take. We further set the minimum route length to 4500 m, the speed exponent to four, favoring edges with a higher speed limit, and the repetition rate to 0.02 to generate many routes.

Before the *ROUTESAMPLER* selected routes from the generated set, we removed obvious U-turns from the route set by removing all routes that included subsequent edges that had the same from- and to-node, but reversed. With 36,817 routes removed, 150,155 routes remained. The *ROUTESAMPLER* then selected the routes to fulfill the 4214 edge data constraints optimally. The parameters included a minimum of three counting stations for each route and an unlimited number of iterations for the optimization. Finally, we executed a *SUMO* simulation with a step length of 1.0 s.

After all the steps described, a first simulation run was executed using the *SUMO* GUI. A qualitative inspection ensured no traffic jams due to implausible routes or vehicle driving behavior. At some places on the street network, there were apparent errors, which we corrected with two measures: First, we extended the share of manually labeled Tomtom-OSM relations with edges with an evident road-network-matching error. These errors especially concerned highway entrances where the road-matching ML model could not correctly distinguish the entrance and the actual motorway. If problems persisted even after this step, we checked the final traffic counts assigned to the edges around the one where a problem had occurred and adjusted them manually to enforce consistency. We applied one iteration of the same procedure to the 50 edge counts with the highest input and output count mismatches presented by the *ROUTESAMPLER* mismatch output file.

### 3.6. Evaluation Metrics

To evaluate the road-network-matching approach, we relied on the metrics accuracy, precision, recall, and  $F_1$ -score (e.g., [38]), which are defined as

$$\text{Accuracy} = \frac{\text{tp} + \text{tn}}{\text{tp} + \text{tn} + \text{fp} + \text{fn}} \quad (7)$$

$$\text{Precision} = \frac{\text{tp}}{\text{tp} + \text{fp}} \quad (8)$$

$$\text{Recall} = \frac{\text{tp}}{\text{tp} + \text{fn}} \quad (9)$$

$$F_1\text{-score} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (10)$$

tp refers to “true positive”, fn refers to “false negative”, tn refers to “true negative”, and fp denotes a “false positive”.

In addition, to analyze to what extent the route sampling after *ROUTESAMPLER* matches the upscaled probe count data, we inspected the mismatches for the respective

edges with an affinity index and the Geoffrey E. Havers (GEH) statistic. The GEH statistic (e.g., [39]) is calculated with the formula

$$GEH = \sqrt{\frac{2 \cdot (M - C)^2}{M + C}} \tag{11}$$

C is the true observed flow, while M is the flow in the simulation model. The units for M and C are usually (also, in this paper) vehicles per hour. This results in a unit for GEH of  $\sqrt{\frac{\text{vehicles}}{\text{hour}}}$  and a metric that is hard to interpret. The most intuitive interpretation is to understand GEH as the geometric mean of the absolute and the relative difference between counted and simulated values (for a deduction, see [40]). Edges considered a good fit have GEH values below five, while edges with GEH values above ten do not fit well [41].

As a further metric, we employed an affinity index resembling the evaluation in [16]. We define the affinity index for our application as the minimum divided by the maximum of upscaled probe count C and simulation count M:

$$\text{Affinity} = \frac{\min(M, C)}{\max(M, C)} \tag{12}$$

### 4. Results

#### 4.1. Road Network Matching

To evaluate the performance of the road-matching ML models, we manually labeled the matching Tomtom edges for 255 randomly selected OSM edges, resulting in 1028 OSM–Tomtom pairs after applying the preprocessing filters. For all metrics except recall in Table 3, the single decision tree slightly outperformed the ensemble of 25 gradient-boosted trees. We, thus, decided to use the single decision tree for our subsequent road-matching activities and as the basis for the following traffic simulation steps.

**Table 3.** Evaluation results for the decision tree and gradient-boosted trees.

	Accuracy	Precision	Recall	F <sub>1</sub> -Score
Decision tree	<b>93.5</b>	<b>92.7</b>	98.8	<b>95.6</b>
Gradient boosting	92.9	91.8	<b>99.0</b>	95.3

The highest score in each column is highlighted in bold.

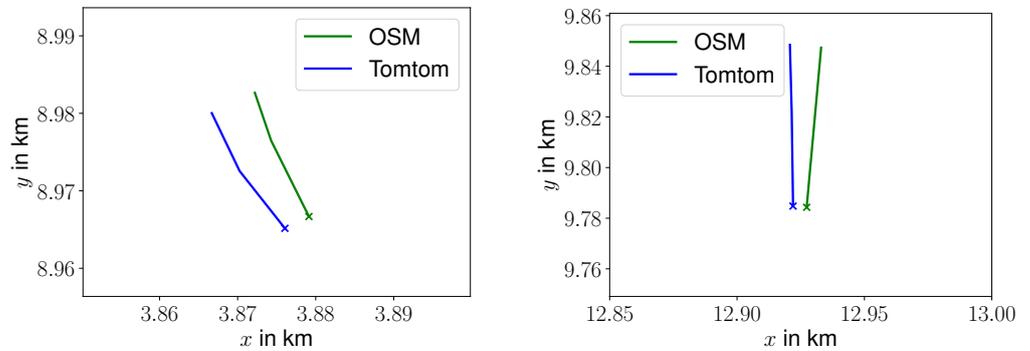
Table 4 displays the confusion matrix for the decision tree predictions. The resulting evaluation metrics are an accuracy of 93.5%, precision of 92.7%, recall of 98.8%, and F<sub>1</sub>-score of 95.6%.

**Table 4.** Confusion matrix for road-network-matching model.

		Ground truth		Total
		Match	Non-match	
Prediction	Match	732	58	790
	Non-match	9	229	238
Total		741	287	1028

Two examples of false predictions of the road-network-matching model are shown in Figure 4. In the first example, the modified Hausdorff distance is 5.05 m, and the cosine similarity is 0.988. The roads are similar, but do not point in the exact same direction. However, they describe the same entity in the two different maps. In the second example, the roads are not parallel and do not describe the same entity because the OSM edge is a road leading into the edge from the Tomtom network. However, only the subsequent OSM edge will describe the same road as the subsequent Tomtom edge. Still, the ML model wrongly predicts the edges to describe the same street segment. The modified Hausdorff

distance for the second example is 11.34 m, and the cosine similarity is 0.994. The decision rules in Figure 3 reveal that candidates with cosine similarity above 0.993 and a modified Hausdorff distance below 12.79 m are predicted as matches. The error happens because both Hausdorff distances are below the threshold. Nonetheless, only the non-matching example has a cosine similarity above the threshold of 0.993.



(a) A false negative.

(b) A false positive.

**Figure 4.** Two incorrectly detected relations. The “x” markers belong to the first coordinate in the respective line string, showing that the edges are oriented in the same directions.

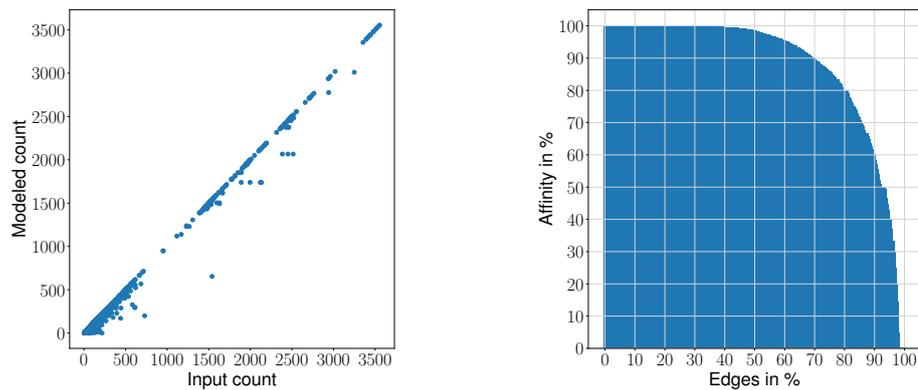
#### 4.2. ROUTESAMPLER

After all manual corrections, the *ROUTESAMPLER* output achieved 95.06% of the input count at 4214 locations. The GEH was below five for 94.49% of the edges with counting information. There were 18 warnings of no routes passing an edge. With the upscaled probe counts as count data, there were 4.7% of the edges with  $5 < GEH \leq 10$  and 0.7% with  $GEH > 10$ . Further summary statistics are shown in Table 5.

**Table 5.** Count summary statistics on *ROUTESAMPLER*.

	Total Count	Achieved	Underflow	Overflow	GEH < 5
Statistics	881,521	838,015	43,838	−332	94.49%

A total of 838,015 vehicle counts were successfully modeled. Visual inspection of the simulation in the *SUMO* GUI did not show any traffic jams or other unwanted effects within the targeted simulation period, i.e., 3600 s. Figure 5a depicts the corresponding modeled count numbers for the input counts. Overall, the plot looks like a 45° straight, which would represent a perfect agreement. However, the agreement appears less obvious for input counts below 900. There was a significant outlier for an input count of 1540, which belongs to a redundant edge that was not appropriately converted from the OSM to the *SUMO* network. For input counts exceeding 2500, there was an excellent agreement. Figure 5b shows a similar pattern. For 80% of the edges, the affinity was at least 80%. For almost 40% of the edges, the affinity was close to 100%. The routes selected by *ROUTESAMPLER*, thus, appear to be in good agreement with the probe counts that served as the input.



(a) Upscaled probe counts vs. simulated counts. (b) Affinity index over edges.

**Figure 5.** Quantitative evaluations of the *ROUTESAMPLER* results.

4.3. *Microscopic Simulation in SUMO*

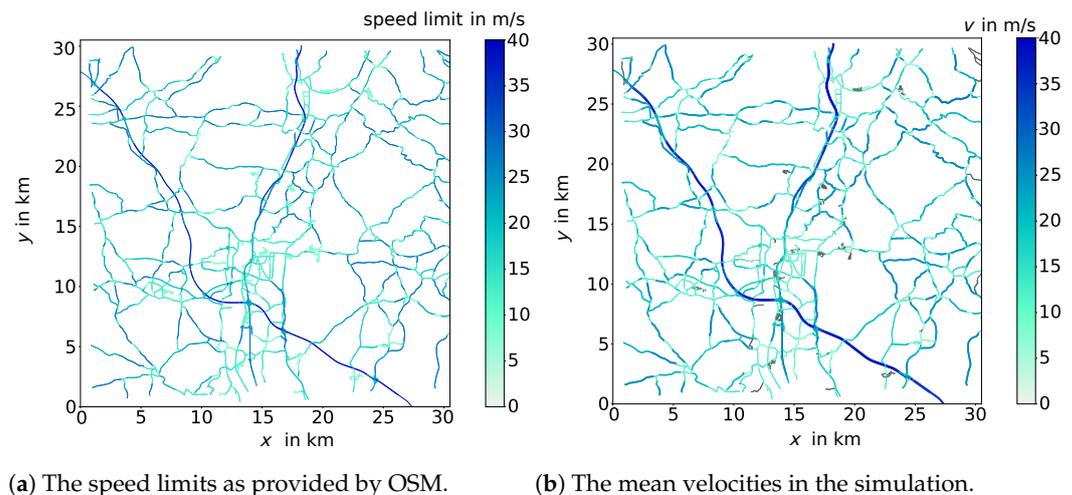
The final simulation showed no warnings regarding emergency braking or vehicles disappearing due to waiting too long. Also, a visual inspection of the cars moving on the map within the *SUMO* GUI showed no traffic jams. Cars were waiting at crossings or traffic signals, which appears reasonable around noon in the selected area. Table 6 gives an overview of final summary statistics regarding the sums of vehicles, trips, and their behavior in the simulation. After a ramp-up phase of approximately 1000 s, the number of running vehicles settled to around 4200 to 4300. This also applies to the number of active vehicles at the end of the simulation. The average trip length is slightly above 14 km, which may seem pretty high, but can be explained by the two prominent highways in the scenario. Similarly, the mean trip duration was just above ten minutes. No teleports, emergency stops, or collisions were happening in the simulation.

**Table 6.** Summary statistics on the traffic simulation.

Statistic	Value
Number of vehicles <sup>1</sup>	23,192
Final running vehicles	4318
Final waiting vehicles	3468
Total vehicle trips	18,874
Mean route length in km	14.16
Mean velocity in m/s	24.32
Mean trip duration in s	627.7
Mean waiting time in s	15.68
Teleports <sup>2</sup>	0
Emergency stops	0
Collisions	0

<sup>1</sup> Inserted vehicles. <sup>2</sup> Those happen if there is a too long waiting time (default: >300 s) for a vehicle.

Figure 6b shows the distribution of vehicles' mean speeds on the street network's edges. The two motorways in the network are the roads associated with the highest speeds. However, for the nearly vertical motorway in the map, the velocities diminish where the motorway crosses the city of Erlangen to a level of 20 to 25 m/s. From a simulation perspective, this is comprehensive since there are many on-ramps and speed limits, as Figure 6a reveals. Generally, the simulated velocities match the speed limits very well, which is in line with the observation that there are no traffic jams in the network.



**Figure 6.** The thicker the line strength in (b) and the darker the color, the faster vehicles go on the road. Thin gray lines are edges without data, i.e., without cars.

## 5. Discussion

### 5.1. Road Network Matching

The differences in road networks targeted by prior road-network-matching approaches and their varying data sources make it hard to compare the accuracy of our approach to previous ones. However, we can state that an  $F_1$ -score of 95.6% implies state-of-the-art performance for our road networks to be matched. The precision of 92.7% being lower than the recall of 98.8% is less desirable than vice versa for our application in this paper since false positives imply inconsistencies in the count data and worsen the traffic simulation. At the same time, (some) missing values should not impede the simulation that much. However, we alleviated this issue by visually inspecting the simulation afterward and correcting the most significant underflow edges after executing *ROUTESAMPLER*.

Following the insights from the two exemplary mismatches in the Results Section and further examples we viewed, there may be non-matches above the thresholds in the decision tree and matches below the thresholds. These findings only allow for the conclusion that, for even better road network matching, there would need to be either more or different similarity metrics or other, more complex rules arranging comparisons alongside these metrics. Using gradient-boosted trees, we had already allowed for more complicated rules without significant improvement. More specifically, the better results of the gradient-boosted trees for the training road network and the worse performance on samples from the more extensive target network imply that the gradient-boosted trees model had been overfitted to the data in the small network or there is some concept drift from the small to the extensive network. Furthermore, we presented way more metrics than modified Hausdorff distance and cosine similarity, which the ML model finally considered. Thus, we assumed there is little room for improvement with solely geometric features. Subsequently, further ML-based approaches should include—if, as in our case, there is no consistent semantic information—at least some contextual measures, such as strokes.

When comparing the performance of our road-matching approach to the ones reported in the literature, the judgment very much depends on the network types: For networks with very few nearby and parallel streets, Zuo et al.'s approach [23] outperformed our approach with an  $F_1$ -scores of up to 97.3%, while the more complex networks they tested yielded  $F_1$ -scores of only 85.3%. Juxtaposing our approach with the method from Yu et al. [20], it still remains competitive: they reported a recall of 91.7% and a precision-like metric of 93.3%. The only method with consistently higher  $F_1$ -scores of at least 97% for two test sites and also an extensive comparison to other methods on the same test sites is the min-cost network flow approach by Wu et al. [21]. However, we did not have a chance to apply their method to our road network and see how it would perform there.

A significant strength of our road-matching approach is the ease of applicability. The method does not contain complex pre-processing steps like stroke building or prescribe fixed threshold values for similarity metrics, which would need to be adjusted for another network with systematic differences. No manual steps are required after one initial labeling effort and splitting labeled matched edges from two networks. Further, the machine learning model adapts to each network and, in that way, can learn the most appropriate similarity thresholds for the network. Also, we made the code and the model publicly available for further applications and for using the method as a benchmark.

### 5.2. Traffic Simulation

While the average metrics with a total achieved count of 94.49% of the edge data file count and median underflow and overflow results reflect a decent microscopic modeling of demand in the selected area, some edges still have significant underflow. An inspection of the 50 edges with the highest deviation shows that the multitude of those mismatches results from problems with the road network matching, especially near crossings. A potential reason is that the exact shape of intersections is often not accurately modeled in OSM (compare, e.g., [3]). Edges comprising the intersection, thus, cannot be matched appropriately with edges from a more accurate, private company's street network. Furthermore, we found many missed probe counts to belong to edges near the fringe of the road network without a preceding edge. This can be due to two reasons: First, there may be some probe count "gaps" at OSM edges without a corresponding Tomtom edge resulting from the differences in edge lengths between the networks. While the optimization in the simulation can handle edges without restrictions, having fewer constraints still can impact the quality of modeled traffic. Second, the simulation potentially could not sample enough routes at those fringe edges because even a fringe factor of 50 did not allow sampling enough routes at some fringe edges or, due to time constraints, not enough vehicles could be inserted. Either way, those mismatches disappear when moving away from the fringes. Nonetheless, the final flow count deviation being smaller than the road network matching errors indicates that the steps related to the actual traffic simulation and route sampler are pretty accurate. A near-perfect road network matching with a nearly perfectly consistent edge data file is necessary for achieving even higher agreement of the counting data and the simulation output.

Regarding the traffic simulation, comparisons with other scenarios are problematic, as well, since most scenarios have yet to be objectively evaluated due to missing data. The simulation in this paper stands out in that there are many traffic counts derived from probe counts, and there is a high agreement between simulated traffic flow and those input counts. In comparison, there have been studies with a higher reported agreement of simulation and detector data (e.g., [9] with edge count mismatches of close to zero percent with an insertion rate of 6000 vehicles per hour). However, regarding the percentage of edges with  $GEH < 5$ , our 94.49% exceed the results from [9] for any insertion rate. In addition, their map images with sensors imply that they have much fewer edges with road counts to be met, so it may be easier to fulfill the restrictions with more "random" traffic on other edges.

The statistics of our modeled trips compare well with the simulation scenario from [1]. While their simulation generated approximately 29,000 trips per hour with up to 15,000 running vehicles, it was 18,874 with 4300 running vehicles in our simulation. Their higher number of running vehicles, albeit a smaller simulation area (400 km<sup>2</sup> as opposed to 930 km<sup>2</sup>), can be explained by the city of Cologne being much bigger and having a way higher population density than the city of Erlangen with surrounding rural areas. The simulation scenario introduced by [16] provides significantly more trips per hour, with 91,783, but also has many more cars in the simulation, with up to 20,000 running vehicles. The Turin simulation having slightly disproportionately more trips indicates that the trips in the simulation are shorter than in our simulation, which seems plausible since their scenario is more urban than the one in this paper. In addition, we excluded the least significant

road types from our simulation scenario, which may also have moved the distribution of simulated trips to longer trips.

### 5.3. Limitations

A limitation of this work is the penetration rate of the probe counts. While the overall penetration rate is significantly higher than reported in earlier works relying on or integrating FCD, the assumption of a constant upscaling factor from probe counts to actual traffic counts is not (yet) realistic. We used the median as an upscaling factor for a more robust metric to avoid skewing our upscaling factor because of outliers. Still, a non-constant, for example, road-type-dependent upscaling might be more accurate. Due to too few real detectors in the area of investigation, we did not have enough data to derive a finer upscaling model. Less data on minor roads should be bearable for the use cases targeted with this simulation framework. These aim was at vehicle-based collaborative sensing only, where a reliably high number of vehicles is permanently available within at least one hour. Thus, replication with richer detector data will be needed to have more refined upscaling approaches prospectively. In addition, increasing penetration rates in the future may alleviate these concerns.

Furthermore, the work shows a specific one-hour scenario on a particular day. We only have the probe counts available today, so this paper cannot extend the proposed methods to other scenarios. However, the strategies should still be generalizable with the following considerations: First, in this work, we already figured that the ratio of probe counts to public traffic counts varies over time and different street types. Thus, for any scenario where probe counts are upscaled to meet public traffic counts, this shall happen on the finest spatial and temporal scale possible. Second, the upscaling factors seemed relatively constant for motorways and significant streets, especially applicable to large-scale scenarios focusing on essential streets. If those constraints are acceptable, the methods proposed in this scenario should work well for other scenarios, i.e., times or areas. Since the ML approach for road network matching will automatically detect the most appropriate threshold values for pre-defined similarity metrics for a pre-labeled dataset, we recommend manually labeling a small, but representative subset of a street network of interest. After some manual corrections, the parameters for the traffic simulation can then be selected to meet the specific scenario.

## 6. Conclusions

This work proposed a new method to model a microscopic traffic simulation for 930 km<sup>2</sup> areas: high-resolution probe counts from a navigation service provider were upscaled with low-resolution traffic counts from a traffic authority. After introducing a simple, yet generalizable road network matching approach, we mapped these data with 4214 counting locations to an OSM map suitable for a *SUMO* traffic simulation. The simulation results achieved a 94.49% agreement of output traffic flow and input counting data. Further evaluation metrics indicated a good agreement between the input and modeled traffic counts. Furthermore, visually inspecting the traffic simulation, we did not observe any inconsistent behavior.

Thus, including probe counts can yield fairly consistent microscopic simulations whose flow values' realism approximates the measured probe counts. The simulation's results could be even more realistic with a more accurate road-network-matching approach. The one proposed in this work, with the strength of being easy to apply and easily adjusted to other networks and different kinds of road networks without the necessity of experimenting on different threshold values for similarity metrics, is limited by its precision of 92.7% on the evaluation data.

The encouraging results imply that road-network-matching research should make its most promising approaches available to obtain an even better agreement of the input and modeled counts. An implication for the industry is that probe counts could also serve as the input for high-quality and large-scale traffic simulations with associated business

models. Progress in that field could accelerate traffic simulation evaluation in many ways since, in future works, edge counts, vehicle velocities, and routes in large-scale microscopic simulations could be validated reliably.

**Author Contributions:** Conceptualization, T.V.; methodology, T.V.; validation, T.V.; investigation, T.V.; resources, V.S. and T.V.; data curation, T.V.; writing—original draft preparation, T.V.; writing—review and editing, M.L., N.F., and V.S.; visualization, T.V.; supervision, M.L. and N.F. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The code for the road-network-matching approach, the resulting machine learning model, and the code for processing intermediate simulation results, the OSM-based SUMO network file, hence the information needed to comprehend the simulation, is available online at <https://github.com/19veihito95/trafficpubl> (accessed on 26 March 2024). The publication of the source data of the probe counts and any derived data is prohibited by contract.

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

## Abbreviations

The following abbreviations are used in this manuscript:

BAYSIS	BAYerisches StraßenInformationsSystem
CAV	Connected and Automated Vehicle
FCD	floating car data
GEH	Geoffrey E. Havers
ML	machine learning
O-D	origin–destination
OSM	OpenStreetMap
SUMO	<i>Simulation of Urban Mobility</i>
VANET	vehicular ad hoc network

## Appendix A. Similarity Metrics

### Appendix A.1. Sinuosity Similarity

Our implementation of sinuosity similarity is

$$\text{Similarity}_{\text{Sin}} = \frac{\min(\text{Sinuosity}(A), \text{Sinuosity}(B))}{\max(\text{Sinuosity}(A), \text{Sinuosity}(B))}, \quad (\text{A1})$$

where sinuosity as described in [42] is defined as

$$\text{Sinuosity} = \frac{\text{Length of road segment}}{\text{Distance between the road segment's start and end point}}. \quad (\text{A2})$$

### Appendix A.2. Discrete Fréchet Distance

The Discrete Fréchet distance [43] is defined as:

$$\delta_{dF}(P, Q) = \min \left\{ \|L\| \mid L \text{ is a coupling between } P \text{ and } Q \right\}, \quad (\text{A3})$$

where a coupling  $L$  is a sequence

$$(u_{a_1}, v_{b_1}), (u_{a_2}, v_{b_2}), \dots, (u_{a_m}, v_{b_m}) \quad (\text{A4})$$

and  $\|L\|$  is

$$\|L\| = \max_{i=1, \dots, m} d(u_{a_i}, v_{b_i}), \quad (\text{A5})$$

where  $P$  and  $Q$  are polygonal curves and  $\sigma(P) = (u_1, \dots, u_p)$  and  $\sigma(Q) = (v_1, \dots, v_q)$  are the corresponding sequences [43], or points representing a line string in the scope of this paper.

## References

- Uppoor, S.; Trullols-Cruces, O.; Fiore, M.; Barcelo-Ordinas, J.M. Generation and Analysis of a Large-Scale Urban Vehicular Mobility Dataset. *IEEE Trans. Mob. Comput.* **2014**, *13*, 1061–1075. [CrossRef]
- Miucic, R. Introduction. In *Connected Vehicles*; Miucic, R., Ed.; Wireless Networks Series; Springer: Cham, Switzerland, 2019.
- Lobo, S.; Neumeier, S.; Fernandez, E.M.G.; Facchi, C. InTAS—The Ingolstadt Traffic Scenario for SUMO. In Proceedings of the SUMO User Conference 2020, Virtual, 26–28 October 2020; pp. 73–92. [CrossRef]
- Wang, Y.; de Veciana, G.; Shimizu, T.; Lu, H. Deployment and Performance of Infrastructure to Assist Vehicular Collaborative Sensing. In Proceedings of the 2018 IEEE 87th Vehicular Technology Conference (VTC Spring), Porto, Portugal, 3–6 June 2018; pp. 1–5. [CrossRef]
- Bajpai, S.; Sahoo, G.K.; Kumar Das, S.; Singh, P. An Efficient Inter-Vehicle Communication Framework on Road Traffic Accident Detection using OMNET++ and SUMO. In Proceedings of the 2022 IEEE 2nd International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC), Gunupur, India, 15–17 December 2022; pp. 1–6. [CrossRef]
- Lopez, P.A.; Behrisch, M.; Bieker-Walz, L.; Erdmann, J.; Flötteröd, Y.P.; Hilbrich, R.; Lücken, L.; Rummel, J.; Wagner, P.; Wiessner, E. Microscopic Traffic Simulation using SUMO. In Proceedings of the IEEE Intelligent Transportation Systems Conference, Maui, HI, USA, 4–7 November 2018; pp. 2575–2582. [CrossRef]
- Tsanakas, N.; Gundlegård, D.; Rydergren, C. O–D matrix estimation based on data-driven network assignment. *Transp. B Transp. Dyn.* **2023**, *11*, 376–407. [CrossRef]
- Abrahamsson, T. *Estimation of Origin-Destination Matrices Using Traffic Counts—A Literature Survey*; Technical Report IR-98-021; IIASA: Laxenburg, Austria, 1998.
- Guastella, D.A.; Cornelis, B.; Bontempi, G. Traffic Simulation with Incomplete Data: The Case of Brussels. In Proceedings of the 1st ACM SIGSPATIAL International Workshop on Methods for Enriched Mobility Data: Emerging Issues and Ethical Perspectives 2023, Hamburg, Germany, 13 November 2023; pp. 15–24. [CrossRef]
- Gonzalez-Delgado, J.J.; Gozalvez, J.; Mena-Oreja, J.; Sepulcre, M.; Coll-Perales, B. Alicante-Murcia Freeway Scenario: A High-Accuracy and Large-Scale Traffic Simulation Scenario Generated Using a Novel Traffic Demand Calibration Method in SUMO. *IEEE Access* **2021**, *9*, 154423–154434. [CrossRef]
- Dabbas, H.; Fourati, W.; Friedrich, B. Floating Car Data for Traffic Demand Estimation—Field and Simulation Studies. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–8. [CrossRef]
- Zhang, Y.-C.; Zuo, X.-Q.; Zhang, L.-T.; Chen, Z.-T. Traffic Congestion Detection Based On GPS Floating-Car Data. *Procedia Eng.* **2011**, *15*, 5541–5546. [CrossRef]
- Tomtom. Traffic Stats. Available online: <https://www.tomtom.com/products/traffic-stats/> (accessed on 22 January 2024).
- Bieker, L.; Krajzewicz, D.; Morra, A.; Michelacci, C.; Cartolano, F. Traffic Simulation for All: A Real World Traffic Scenario from the City of Bologna. In *Modeling Mobility with Open Data*; Behrisch, M.; Weber, M., Eds.; Springer: Cham, Switzerland, 2015; pp. 47–60.
- Codeca, L.; Frank, R.; Faye, S.; Engel, T. Luxembourg SUMO Traffic (LuST) Scenario: Traffic Demand Evaluation. *IEEE Intell. Transp. Syst. Mag.* **2017**, *9*, 52–63. [CrossRef]
- Rapelli, M.; Casetti, C.; Gagliardi, G. Vehicular Traffic Simulation in the City of Turin From Raw Data. *IEEE Trans. Mob. Comput.* **2022**, *21*, 4656–4666. [CrossRef]
- Ketabi, R.; Alipour, B.; Helmy, A. En route: Towards vehicular mobility scenario generation at scale. In Proceedings of the 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Atlanta, GA, USA, 1–4 May 2017; pp. 839–844. [CrossRef]
- Flötteröd, G. A search acceleration method for optimization problems with transport simulation constraints. *Transp. Res. Part B Methodol.* **2017**, *98*, 239–260. [CrossRef]
- Xia, F.; Rahim, A.; Kong, X.; Wang, M.; Cai, Y.; Wang, J. Modeling and Analysis of Large-Scale Urban Mobility for Green Transportation. *IEEE Trans. Ind. Inform.* **2018**, *14*, 1469–1481. [CrossRef]
- Yu, W.; Liu, M. An iterative framework with active learning to match segments in road networks. *Cartogr. Geogr. Inf. Sci.* **2023**, *50*, 333–350. [CrossRef]
- Wu, H.; Xu, S.; Huang, S.; Wang, J.; Yang, X.; Liu, C.; Zhang, Y. Optimal road matching by relaxation to min-cost network flow. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *114*, 103057. [CrossRef]
- Klein, M. A Primal Method for Minimal Cost Flows with Applications to the Assignment and Transportation Problems. *Manag. Sci.* **1967**, *14*, 205–220. [CrossRef]
- Zuo, Z.; Yang, L.; An, X.; Zhen, W.; Qian, H.; Dai, S. A Hierarchical Matching Method for Vectorial Road Networks Using Delaunay Triangulation. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 509. [CrossRef]
- Guo, Q.; Xu, X.; Wang, Y.; Liu, J. Combined Matching Approach of Road Networks Under Different Scales Considering Constraints of Cartographic Generalization. *IEEE Access* **2020**, *8*, 944–956. [CrossRef]

25. Hacar, M.; Gökğöz, T. A new approach for matching road lines using efficiency rates of similarity measures. *Int. J. Eng. Geosci.* **2021**, *6*, 146–156. [[CrossRef](#)]
26. BAYSIS. Dauerzählstellen. Available online: <https://www.baysis.bayern.de/internet/verdat/dauz/index.html> (accessed on 14 February 2024).
27. OpenStreetMap. OpenStreetMap Wiki: Key:Highway. Available online: <https://wiki.openstreetmap.org/wiki/Key:highway> (accessed on 8 March 2024).
28. Tomtom. Traffic Stats Documentation: Product Information—FAQ. Available online: <https://developer.tomtom.com/traffic-stats/documentation/product-information/faq> (accessed on 8 March 2024).
29. Sun, Y.; Lu, Y.; Ding, Z.; Wen, Q.; Li, J.; Liu, Y.; Yao, K. Multi-Scale Road Matching Based on the Summation Product of Orientation and Distance and Shape Descriptors. *ISPRS Int. J. Geo-Inf.* **2023**, *12*, 457. [[CrossRef](#)]
30. Veihelmann, T.; Seifaei, M.; Lübke, M.; Franchi, N. Leveraging Gradient-Boosted Trees to Combine FFT, Correlation, and Fréchet-distance Features for Clear-sky Detection. In Proceedings of the 40th European Photovoltaic Solar Energy Conference and Exhibition, Lisbon, Portugal, 18–22 September 2023; pp. 020458-001–020458-005. [[CrossRef](#)]
31. Zhang, M. Methods and Implementations of Road-Network Matching. Ph.D. Thesis, Technische Universität München, Munich, Germany, 14 October 2009.
32. Huttenlocher, D.P.; Klanderman, G.A.; Rucklidge, W.J. Comparing images using the Hausdorff distance. *IEEE Trans. Pattern Anal. Mach. Intell.* **1993**, *15*, 850–863. [[CrossRef](#)]
33. Turney, P.D.; Pantel, P. From Frequency to Meaning: Vector Space Models of Semantics. *J. Artif. Intell. Res.* **2010**, *37*, 141–188. [[CrossRef](#)]
34. Lemaître, G.; Nogueira, F.; Aridas, C.K. Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *J. Mach. Learn. Res.* **2017**, *18*, 1–5. [[CrossRef](#)]
35. Mitchell, T.M. *Machine Learning*; McGraw-Hill: New York, NY, USA, 1997.
36. Friedman, J.H. Greedy Function Approximation: A Gradient Boosting Machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [[CrossRef](#)]
37. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830. [[CrossRef](#)]
38. Powers, D.M.W. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *Int. J. Mach. Learn. Technol.* **2011**, *2*, 37–63. [[CrossRef](#)]
39. Department for Transport. TAG UNIT M3.1: Highway Assignment Modelling. Available online: [https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment\\_data/file/938864/tag-m3-1-highway-assignment-modelling.pdf](https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/938864/tag-m3-1-highway-assignment-modelling.pdf) (accessed on 27 February 2024).
40. Friedrich, M.; Pestel, E.; Schiller, C.; Simon, R. Scalable GEH: A Quality Measure for Comparing Observed and Modeled Single Values in a Travel Demand Model Validation. *Transp. Res. Rec. J. Transp. Res. Board* **2019**, *2673*, 722–732. [[CrossRef](#)]
41. Schweizer, J.; Poliziani, C.; Rupi, F.; Morgano, D.; Magi, M. Building a Large-Scale Micro-Simulation Transport Scenario Using Big Data. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 165. [[CrossRef](#)]
42. Stolum, H.H. River Meandering as a Self-Organization Process. *Science* **1996**, *271*, 1710–1713. [[CrossRef](#)]
43. Eiter, T.; Mannila, H. *Computing Discrete Frechet Distance: CD-TR 94/64*; Technical report; Christian Doppler Labor für Expertensysteme, Technical University of Vienna: Vienna, Austria, 1994.

**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.