IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 7, 2021, accepted March 23, 2021, date of publication March 26, 2021, date of current version April 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3068964

Vehicle Trajectory Reconstruction on Urban Traffic Network Using Automatic License Plate Recognition Data

XINYI QI [1], YANJIE JI [1], WENHAO LI [1,2], AND SHUICHAO ZHANG [3]

[1] School of Transportation, Southeast University, Nanjing 211189, China
[2] School of Informatics, Computing, and Cyber Systems, Northern Arizona University, Flagstaff, AZ 86011, USA
[3] School of Civil and Transportation Engineering, Ningbo University of Technology, Ningbo 315211, China

Corresponding author: Yanjie Ji (jiyanjie@seu.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFE0120100, in part by the 16th Regular Meeting Communication Program of China-Bulgaria Science and Technology Cooperation Committee, and in part by the Fundamental Research Funds for the Central Universities under Grant 2242020K40063.

ABSTRACT Vehicle trajectory data are critical to urban active traffic management and simulation applications. Automatic license plate recognition (ALPR) data can provide partial vehicle trajectory information by matching the detected vehicle license plates through time series. However, the trajectory extracted from ALPR data tend to be sparse and incomplete due to technical and financial constraints. This paper deals with the problem of sparse trajectory reconstruction based on ALPR data. Firstly, the multiple travel activities of the vehicle are divided based on the reasonable travel time threshold, and the incomplete vehicle trajectory is identified. Then, candidate trajectories are generated by an improved K-shortest-path (KSP) algorithm based on space-time prism theory. Finally, the auto-encoder model is utilized to select the candidate trajectory with optimal decision indicators, which realizes the vehicle trajectory reconstruction. The proposed method was implemented on a realistic urban traffic network in Ningbo, China. The verification results show that the proposed method has a comprehensive accuracy of 85% and good robustness. From the comparison with the baseline algorithm, it can be seen that the proposed method still has high accuracy in low ALPR coverage rate, and there exists a minimum required ALPR coverage rate (50% in the test network) for reconstructing trajectories accurately.

INDEX TERMS Automatic license plate recognition, space-time prism, auto-encoder, trajectory reconstruction.

I. INTRODUCTION

Accurate and reliable vehicle trajectory data is of great importance in the intelligent transportation system and urban traffic management [1]. Trajectory data can provide effective information for many application areas, such as travel time estimation, calibration of traffic flow models, crash prediction, vehicle emissions estimation and OD pattern estimation [2]–[7]. Extracted trajectories can not only reflect the characteristics of traffic flow operation in a microscopic level for a particular road segment, but also illustrate traffic demand and spatial-temporal distribution characteristics in a macroscopic level for the whole traffic network [8].

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Awais Javed.

Vehicle trajectory can be collected by mobile sensors and fixed sensors. Mobile sensor data mainly includes probe car GPS data, FRID data and cellular signaling data. And fixed sensor data includes loop detector data and video detection data, etc. According to different data types, the extraction methods of travel trajectory are different. The representative researches on the extraction of travel trajectory by mobile sensor data include: Daganzo [9] and Mehran *et al.* [10] combined the probe car GPS data and signal timing parameters to study a data fusion framework to reproduce vehicle trajectories on urban arterial roads based on the variational theory and kinematic wave theory. The proposed framework was verified on an arterial road in Tokyo. Bachir *et al.* [11] extracted the moving trajectory of travelers from cellular signaling data by semi-supervised learning algorithm and

Bayesian inference. On the other hand, representative studies on the extraction of travel trajectory by fixed sensors are as follows: Lint [12] obtained average vehicle trajectories based on travel time and link flow counts using multiple loop detectors. Ni and Wang [13] proposed a trajectory reconstruction model accounting for consecutive speed variation based on simulated fixed sensors.

However, the penetration rate of GPS devices is relatively low, the continuity of cellular signaling data is poor, and the non-aggregated loop detector data cannot directly reflect the trajectory of individual vehicles. Consequently, the application of these data to extracting vehicle trajectories of the entire traffic network is limited.

Benefiting from the development of automatic license plate recognition technology and the extensive coverage of ALPR facilities on the urban road network, data collected by ALPR devices has provided new possibilities for high-precision vehicle trajectory reconstruction, which is addressed in this paper. In addition, the recognition accuracy of ALPR devices in China can reach 95% in the daytime and 90% at night for the vehicles detected, which means ALPR data can be regarded as an approximation of actual flows on the traffic network [8]. Although ALPR data have high recognition accuracy, the vehicle trajectories extracted from it may still be incomplete due to the limitation on device coverage and data packet dropout.

Therefore, this study aims to reconstruct all the incomplete trajectories extracted from ALPR data on the urban traffic network. Although there are considerable researches about vehicle trajectory reconstruction based on ALPR data, several limitations still need to be tackled: (1) The computational efficiency of path searching on large urban traffic network is low; (2) The existing studies only take static road network characteristics into account and neglect traveler's individual preference when designing decision criteria; (3) To the best of our knowledge, the process of determining weights of the decision criteria is subjective in previous studies. The weight of each decision criterion may be directly determined by researchers or the weight is updated according to a series of formulas put forward by researchers.

In response to the above research goals and challenges, the main contributions of this study are summarized as follows:

(1) We design a P-KSP algorithm which can increase the efficiency of candidate trajectory searching. Compared with the traditional path searching algorithm used in previous researches such as shortest-path (SP) and depth first search (DFS), this method can avoid traversing though all nodes on the traffic network when deriving candidate trajectories which is time-consuming in large-scale network.

(2) We propose six indicators for path decision, i.e., path length, road hierarchy, number of intersections, number of turns, travel time consistency and path preference, which assist in distinguishing the optimal reconstructed trajectory effectively. Previous studies tend to ignore the dynamic state of traffic network and traveler's subjective preference. For instance, travelers may choose the path according to their own inherent habits even if the path is not the optimal. The two dynamic indicators proposed in this paper can correct the bias to some extent.

(3) We apply auto-encoder model to distinguishing the reconstructed trajectory, which can decrease information loss, derive decision weights objectively and capture the nonlinear relationships between different attributes. Some classical multiple attribute decision making methods, such as TOPSIS, GRA and PCA were exploited to accomplish path decision in the existing studies. However, they have the following two limitations: first, the weight determination is relatively subjective; second, some data information may be missing during the decision-making process. Comparatively, the auto-encoder-based method has little information loss and can represent both linear and nonlinear relationships between different indicators.

The remainder of the paper proceeds as follows. Section 2 reviews the literature on vehicle trajectory reconstruction method. Section 3 explains the details of the procedure to reconstruct the vehicle trajectory on urban traffic network. Section 4 illustrates the application of the proposed method in realistic traffic network through a case study. While in Section 5, it presents the performance of the proposed method set under different scenarios. Section 6 concludes this paper and suggests a few future research directions.

II. LITERATURE REVIEW

Vehicle trajectory reconstruction method can be classified into two categories: approach based on traffic flow theory (TFT) and statistical modeling approach. Classical TFT-based methods are mostly focused on trajectory reconstruction on freeways. Coifman [14] proposed a TFT-based method for travel time estimation and trajectory estimation exploiting data from a dual loop detector. On this basis, Lint and Hoogendoorn [15] put forward a data fusion algorithm to reconstruct vehicle trajectory. This method allowed data fusion from local sensors, moving vehicles and automatic vehicle identification (AVI) systems, and had been proved to be robust through a micro-simulation tool. Sharma *et al.* [16] reconstructed vehicle trajectories from NGSIM data by obtaining the relationship between trajectory and speed profiles. However, the TFT-based method is applicable to the uninterrupted flow on freeways and is not suitable for the interrupted flow on the urban traffic network with numerous intersections.

In recent years, in order to realize the trajectory reconstruction on urban traffic network, researchers have proposed several statistical modeling approaches such as particle filter (PF), maximum likelihood estimate (MLE) and probability distribution model. For instance, Castillo *et al.* [17] extracted part trajectories from the ALPR data and reconstructed the path flow based on the least quadratic function.

The model was applied on the ND road network, showing its practical value in path flow estimation.

However, with the increase of urban road network scale, the prior data required by MLE model is difficult to collect, thus, Feng *et al.* [18] employed particle filter theory to estimate vehicle trajectories. The particle filter updated the weights of candidate trajectories according to five correction factors in the importance sampling process and the partial trajectory was reconstructed by the particle with maximum weight. This method was validated in the VISSIM model, results showing a reliability of 90% when AVI coverage exceeded 50%. To enhance the estimation precision, Yang and Sun [19] designed an integrated framework with a particle filter at the microscopic level and the stochastic user equilibrium principle at the macroscopic level. The test results on the realistic network showed that this framework improved the accuracy of trajectory reconstruction under low detector coverage. Rao *et al.* [20] studied the trajectory reconstruction accuracy based on particle filter under different sampling rates of ALPR devices and the minimum sampling rate was suggested as 60%. Further, Xie *et al.* [6] utilized a particle filter to assimilate noise of multi-source data, which solved the observation error in trajectory reconstruction. Wei *et al.* [21] proposed a novel initial particle procedure without relying on simulation, which improved Xie's method.

Another way to reconstruct trajectory is treating vehicle travel as a consequence of the choice of different routes [22]. Different decision indicators are used to select the optimal travel path. Yu first used DFS algorithm to find all candidate paths for incomplete trajectories extracted from ALPR data, then proposed four decision attributes (section number, speed match degree, path pattern number, and vehicle turning number), and finally selected the optimal trajectory based on TOPSIS algorithm. The verification results on the realistic network showed that the reconstruction accuracy reaches 85% when missing 30% information. Similarly, Ruan *et al.* [23] exploited KSP algorithm to search candidate trajectories and grey relational analysis (GRA) to select the optimal trajectory. While Zhang *et al.* [24] proposed a data-driven method where the turning probability of a vehicle was calculated from historical trajectory data.

In general, the classical TFT-based method performs well in the case of the uninterrupted flow rather than interrupted flow on the urban traffic network. The PF-based method reconstructs the trajectory by updating the state-space equation. However, it requires a large number of samples to well approximate the posterior probability density. This means that the higher the target accuracy rate of trajectory reconstruction, the higher the complexity of the algorithm. The route-choice-based method is easy to understand and convenient to calculate, but existing researches are limited in computational efficiency and accuracy.

Motivated by the above issues, in this study, we propose a vehicle trajectory reconstruction method based on ALPR data. In this model, the idea of space-time prism is applied to enhance the computational efficiency of path searching. In addition, by combining static and dynamic indicators, the model is expected to reconstruct trajectories accurately for different ALPR device coverage rate.

III. METHODOLOGY

Figure 1 shows the flowchart of the proposed method with four main steps. In Step 1, the trip chain is split according to the travel time threshold and the complete trajectory set and incomplete trajectory set are separated. In Step 2, candidate trajectory set is generated by the path searching algorithm which combines the space-time prism theory with K shortest path algorithm. In Step 3, decision indicators of candidate trajectory are calculated and normalized. In Step4, an autoencoder model is applied to selecting the optimal candidate trajectory, then the incomplete trajectory is reconstructed.

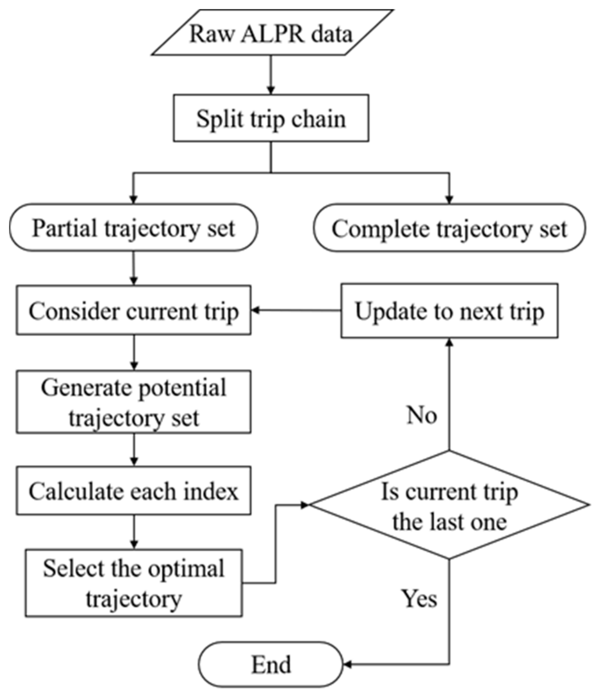


FIGURE 1. Flowchart of the proposed method.

A. TRIP CHAIN DIVISION

The trip chain of most vehicles in a day consists of multiple travel behaviors with different purposes, so it is necessary to divide the trip chain of each vehicle. We split the trip chain based on reasonable travel time between any two travel nodes. If the travel time between two points is not within the reasonable time range, then the two points belong to two different travel activities. It should be noted that we did not take into account the traveler's activity stay time when splitting the trip chain. For example, if a traveler driving a car from home first arrives at a restaurant for dinner and then leaves the restaurant for the supermarket, his trip chain will be divided into two trips: (1) Traveling from home to a restaurant; (2) Traveling from a restaurant to a supermarket.

The dining time in the restaurant will not be counted as travel time.

We set the reasonable travel time interval between any two nodes on the traffic network as $[T_{ij}^{min}, k * T_{ij}^{ave}]$. Where T_{ij}^{min} refers to the shortest travel time between node i and node j; T_{ij}^{ave} refers to the historical average travel time between node i and node j; k is an elasticity coefficient taking a value of 1.5 according to existing research [25]. T_{ij}^{min} and T_{ij}^{ave} can be accessed via the online map API interface. The steps to split the trip chain is described in Table 1.

TABLE 1. Algorithm of trip chain division.

Algorithm 1: Trip Chain Division

Step1: The passing records derived from ALPR devices are clustered according to the license plate ID, and the record of each car is arranged in chronological order.
Step2: For the mth vehicle (a total of M vehicles), set the location of the first record as the starting point of its first trip.
Step3: Calculate the time interval Δt between the jth and $(j+1)$th passing records of the ith car.
(a) if $\Delta t \in [T_{j,j+1}^{min}, k * T_{j,j+1}^{ave}]$, then the jth and $(j+1)$th crossing records belong to the same trip;
(b) if $\Delta t < T_{j,j+1}^{min}$, then mark the $(j+1)$th record as an error and delete it;
(c) if $\Delta t > k * T_{j,j+1}^{ave}$, then mark the location in the jth record as the end point of the current trip and the starting point of the next trip.
Step4: if $m = M$, stop; if $m < M$, $m = m + 1$, return to **Step2**.

B. CANDIDATE TRAJECTORY GENERATION BASED ON P-KSP ALGORITHM

The K shortest path algorithm is often used to generate a set of candidate paths between two points on urban traffic network, but it is computationally inefficient in large or complex networks. Therefore, we designed a K shortest path algorithm combined with space-time prism theory (P-KSP) to generate the candidate trajectory set.

Space-time prism is a geographic information science method that uses spatial-temporal paths to analyze an individual's activity capabilities. By calculating the spatial projection of space-time prism on the network, the search area for candidate paths can be reduced and computational efficiency can be effectively increased.

Assuming that the traveler's origin point is *point A* and the destination point is *point B*, his travel start time is T_1, and his destination time is T_2. Cone 1 with *point* (A, T_1) as the apex and cone 2 with *point* (B, T_2) as the apex will produce an intersection, which is a space-time prism. The area projected on the plane by the intersection of the two cones is the potential path area (PPA). It can be seen that this potential path area is an ellipse with focal points A and B. When the traveler's moving speed is v and the origin and destination points are points A and B, his range of activity is within this ellipse (PPA) [26]. Figure 2 shows the space-time prism corresponding to a trip.

The steps for solving the potential path region (PPA) are as follows. The urban road network is simplified as a directed graph $G = \{V, E\}$, where $V = \{1, 2, \ldots, n\}$ represents the set of all nodes in the network; $E = \{(i,j)|i,j \in V, i \neq j\}$ represents the set of all sections in the road network. Let the starting point in the two-dimensional plane be (x_o, y_o) and the ending point be (x_d, y_d). Set the coordinates of any point on the PPA to be (x, y), then the expression of the PPA is as follows:

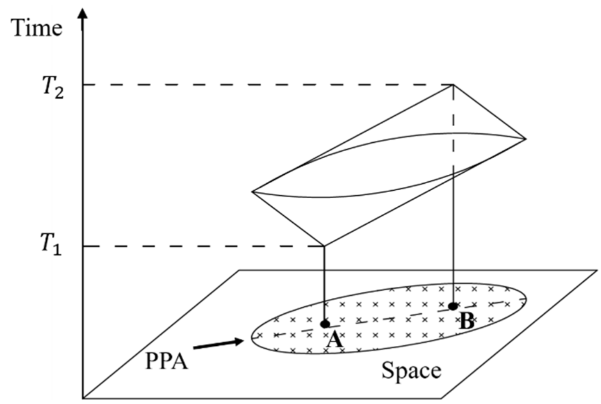


FIGURE 2. Space-time prism.

$$PPA = \{(x, y)\,|T(x_o, y_o, x, y) + T(x, y, x_d, y_d) \leq T_2 - T_1\} \tag{1}$$

where function $T(x_1, y_1, x_2, y_2)$ represents the shortest travel time from point (x_1, y_1) to (x_2, y_2).

Then, we search the candidate path based on KSP algorithm in the possible path area. Path generation in traffic network is a limited K shortest path problem with no closed loop. YEN's algorithm proposed in 1971 is a relatively extensive and effective method currently used to solve the problem of limited K shortest path problem [27]. It first finds the shortest path Q_1 from start point A to end point B using Dijkstra algorithm. The kth $(k > 1)$ shortest path (Q_k) is found using the idea of the deviating path algorithm in recursion: all nodes on Q_{k-1} except the end point are considered deviating nodes and the shortest path from each deviating node to the end point is calculated. Then it is appended to the path from the start point to the deviating node on Q_{k-1}, which forms the final candidate path.

The specific procedure of P-KSP algorithm is presented in Table 2.

C. PATH DECISION INDICATORS

Trajectory reconstruction can be regarded as a process of simulating travelers' path decision. We reproduce the path decision process of travelers by constructing corresponding decision indicators. Main factors considered in previous studies include static information such as path length, road hierarchy, number of intersections and number of turns [28]–[30]. However, the static information may not accurately describe the dynamic decision response of the travelers because the travelers' access to status of the traffic network is lagging and they sometimes choose the path according to their own inherent habits. To solve this problem, we designed two dynamic

TABLE 2. Algorithm of candidate path set generation.

Algorithm 2: Candidate Path Set Generation

Input: The maximum speed limit of the traffic network: V_{max}; the topology of the traffic network: $Graph_{initial}$; the position of $point\ A$ and $point\ B$: ori and des; travel time between $point\ A$ and $point\ B$: T

Output: Candidate path set $Q = \{Q_1, Q_2, \cdots, Q_K\}$ from $point\ A$ to $point\ B$

Process:

```
# obtain the potential path area based on equation(1)
Graph = PPA(Graph_initial, ori, des, V_max, T)
# calculate the shortest path
Q[1] = Dijkstra(Graph, ori, des)
R = [ ]
# calculate the remaining K-1 shortest path
for k from 2 to K do
  for i from 0 to size(Q[k − 1]) − 2 do
    SpNode = Q[k − 1].node(i)
    RtPath = Q[k − 1].nodes(0, i)
    for each path q in Q do
      if RtPath == q.nodes(0, i) then
        remove q.edge(i, i + 1) from Graph
      end
    end
    for each node RtPathNode in RtPath except SpNode do
      remove RtPathNode from Graph
    end
    SpPath = Dijkstra(Graph, SpNode, des)
    TotalPath = RtPath + SpPath
    if TotalPath not in R then
      R.append(TotalPath)
    end
    restore edges to Graph
    restore nodes in RtPath to Graph
  end
  if R is empty then
    break
  end
  R.sort()
  Q[k] = R[0]
  R.pop()
end
return Q
```

indicators, namely travel time consistency and path preference, for correction. The travel time consistency describes the similarity between the travel time of candidate paths and the real travel time recorded by ALPR detection equipment, while the path preference describes the proportion of candidate paths selected in historical data.

Among the above six indicators, path length, road hierarchy, number of intersections, number of turns and travel time consistency belong to the cost-based indicator, while the path preference is a benefit-oriented indicator. Moreover, the driver's sensitivity to the growth of these indicators gradually decreases, so the index-based Max-Min method is used to normalize the indicators.

1) Path length

$$x_1(Q_i) = \begin{cases} e^{-\frac{L_i - \min(L)}{\max(L) - \min(L)}}, & max(L) \neq min(L) \\ 1, & max\ (L) = min(L) \end{cases} \tag{2}$$

where

L_i is the length of path Q_i,

L is the set of path length.

2) Number of intersections

$$x_2(Q_i) = \begin{cases} e^{-\frac{INE_i - \min(INE)}{\max(INE) - \min(INE)}}, & max(INE) \neq min(INE) \\ 1, & max\ (INE) = min(INE) \end{cases} \tag{3}$$

where

INE_i is the intersection numbers of path Q_i,

INE is the set of intersection numbers.

3) Number of turns

$$x_3(Q_i) = \begin{cases} e^{-\frac{T_i - \min(T)}{\max(T) - \min(T)}}, & max(T) \neq min(T) \\ 1, & max\ (T) = min(T) \end{cases} \tag{4}$$

where

T_i is the turning times of path Q_i,

T is the set of turning times.

4) Road hierarchy

$$H_i = \frac{\sum_j^n H_{ij} \times L_{ij}}{\sum_j^n L_{ij}} \tag{5}$$

$$x_4(Q_i) = \begin{cases} e^{-\frac{H_i - \min(H)}{\max(H) - \min(H)}}, & max(H) \neq min(H) \\ 1, & max\ (H) = min(H) \end{cases} \tag{6}$$

where

H_i is the road hierarchy of path Q_i,

n is the number of sections contained in path Q_i,

H_{ij} is the road grade of the jth section in path Q_i,

L_{ij} is the road length of the jth section in path Q_i,

H is the set of road hierarchy.

5) Travel time consistency

$$C_i = \frac{|T_{true}^i - T_{esm}^i|}{T_{true}^i} \tag{7}$$

$$x_5(Q_i) = \begin{cases} e^{-\frac{C_i - \min(C)}{\max(C) - \min(C)}}, & max(C) \neq min(C) \\ 1, & max\ (C) = min(C) \end{cases} \tag{8}$$

where

C_i is the travel time consistency of path Q_i,

T_{true}^i is the true travel time of path Q_i,

T_{esm}^i is the estimated travel time of path Q_i,

C is the set of travel time consistency.

The estimated travel time is obtained by the following steps: (a) The estimation of path travel time is simplified as a regression problem where the path length and the number of intersections are taken as independent variables, and the true travel time is taken as dependent variables; (b) The true travel time, path length and intersection number of each path are extracted from the complete trajectory data set; (c) The data are input into the regression model for fitting so as to get the calculation formula of path travel time; (d) For each candidate trajectory, the estimated travel time is calculated by using the formula.

6) Path preference

$$P_i = \frac{N_{rs}^i}{N_{rs}} \times 100\% \tag{9}$$

$$x_6\left(Q_i\right) = \begin{cases} 1 - e^{-\frac{P_i - \min(P)}{\max(P) - \min(P)}}, & max(P) \neq min(P) \\ 1, & max\ (P) = min(P) \end{cases} \tag{10}$$

where

P_i is the path preference of path Q_i,

N_{rs} is the total travel times between OD pair rs in the historical travel data,

N_{rs}^i is the total travel times of path Q_i between OD pair rs in the historical travel data,

P is the set of path preference.

D. PATH DECISION BASED ON AUTO-ENCODER

We regard the vehicle trajectory reconstruction as the problem of selecting an optimal path from the candidate path set according to the indicator values. Its essence is a multi-attribute decision-making problem. The classical multi-attribute decision-making methods mainly include analytic hierarchy process, principal component analysis, grey relational analysis, TOPSIS method and so on. Their basic idea is to determine the weight of each indicator to get a comprehensive evaluation indicator, then the optimal scheme is selected. Its essence can be understood as the process of mapping multi-dimensional space composed of multiple indicators to low-dimensional space through a certain distance calculation criterion. However, the weight determination of these methods is subjective to some extent, and there are many calculation steps. Therefore, we did not implement path decision based on these classical multiple attribute decision methods. Considering the efficiency and convenience of the auto-encoder model in the field of data dimension reduction and information extraction, we choose it to realize trajectory reconstruction.

Auto-encoder is an unsupervised learning neural network whose target output is the original input. Its purpose is to reconstruct its input to obtain a low-dimensional representation of high-dimensional input. The difference between auto-encoder and PCA is that auto-encoder can represent not only linear transformation but also nonlinear transformation, and it has little information loss in the process of reconstructing input. An auto-encoder consists of two parts, the encoder and the decoder, which is shown in Figure 3. The input variables pass through a small number of neurons in the hidden layer(s), forging a compressed representation of the input (encoding), which is then unpacked and mapped to the output layer (decoding) [31], [32].

The path decision method based on the auto-encoder takes the multi-dimensional vector composed of the indicators of the candidate trajectory as the input, and k-dimensional mapping is carried out through the model. The candidate trajectory with the optimal comprehensive indicator is considered as the reconstructed trajectory. Suppose $\boldsymbol{X}_n$ denotes the n-dimensional decision indicator vector and mapping it to a k-dimensional space includes encoding and decoding processes.

The encoding process can be expressed by (11):

$$Z_i = f(W_e X_i + b_e) \tag{11}$$

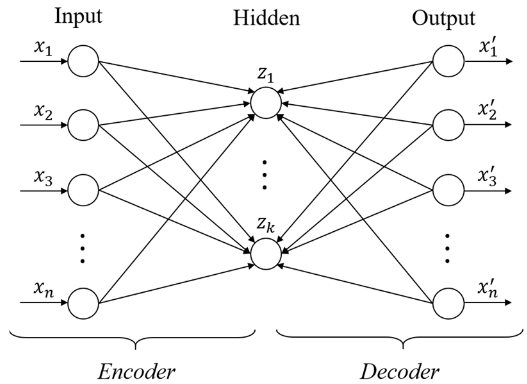


FIGURE 3. **Auto-encoder structure.**

The decoding process can be expressed by (12):

$$\tilde{X}_i = g(W_d Z_i + b_d) \tag{12}$$

where W_e and W_d are encoding weight matrix and decoding weight matrix respectively, b_e and b_d are encoding deviation vector and decoding deviation vector respectively, both $f(\cdot)$ and $g(\cdot)$ are nonlinear activation functions, which are sigmoid function generally.

The loss function of the model is as follows:

$$L\left(x, g\left(f\left(x\right)\right)\right) = ||\tilde{X}_i - X_i||^2 \tag{13}$$

IV. CASE STUDY

A. DATA DESCRIPTION

We selected an urban network in Ningbo, China as a testbed. The road network includes 33 intersections and 108 road sections. A total of 154 ALPR detectors are installed at the intersection of this region. When the vehicle passes through the intersection, these detection devices will record the passing information. Figure 4 shows the study area and the location of ALPR detector. The passing information collected by ALPR equipment includes ALPR device number, detection time, lane number and license plate number. Detailed data

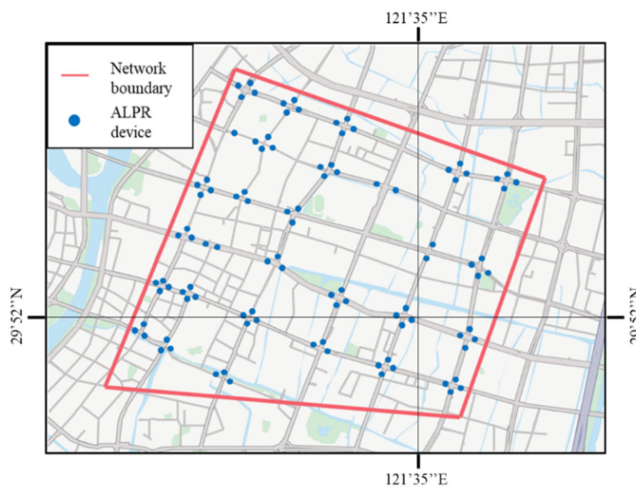


FIGURE 4. **Study area and detector location.**

description is shown in Table 3. The license plate number has been encrypted to protect the driver's privacy.

TABLE 3. ALPR data tags.

Device ID	Snapshot Time	Lane No.	License Plate
11422	2018-06-05 08:33:28	1	025600163
11570	2018-06-05 14:00:05	2	130000130
⋮	⋮	⋮	⋮
11676	2018-06-05 22:15:33	3	015100145

B. IMPLEMENTATION

The following shows the specific process of vehicle trajectory reconstruction. Figure 5 shows the initial trajectory of the vehicle in this example. The vehicle was captured in six locations. We marked these six points in chronological order. Firstly, we compared the interval time between all adjacent recording points and their reasonable travel time intervals to determine whether the trip chain needs to be divided. The results show that the travel time between position 2 and position 3 is two hours, which exceeds the maximum reasonable travel time between position 2 and position 3. According to the trip chain division criterion proposed in Section 2, this trajectory was divided into two travel activities. Position 2 is the end of trip 1 and the starting point of trip 2. Figure 5 shows that the trajectory between intersection 18 and intersection 27 in trip 1 is incomplete and needs reconstruction.

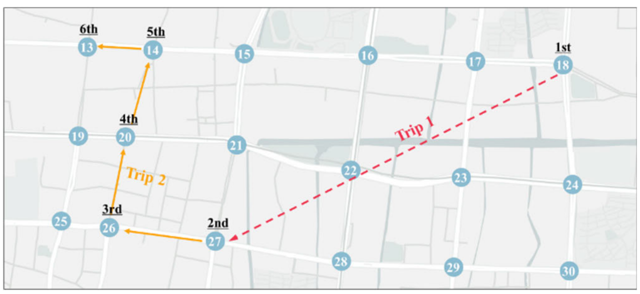


FIGURE 5. Initial trip trajectory.

Considering the actual traffic network scale, we used the P-KSP algorithm proposed above to generate five candidate trajectories for trip 1, as shown in Table 4. Compared with SP algorithm, the proposed P-KSP algorithm saved an average of 19% searching time in this traffic network. After that, we calculated the six indicators of these five paths and normalize the indicator values. The results are shown in Table 5.

Then we established an auto-encoder model with six dimensions in both the input and the output layers. The middle layers of the model are all hidden layers, and the number of neurons is 4, 3, 2, 3 and 4, respectively. The first four layers constitute the encoder, and the last four layers constitute the decoder. Considering the independence between static and dynamic indicators, we map six-dimensional inputs to two-dimensional vectors, namely z_1 and z_2. The activation function of this model is sigmoid function, and the model structure is shown in Figure 6.

TABLE 4. Candidate trajectory set.

No.	Path
1	18-17-16-15-21-27
2	18-17-16-22-28-27
3	18-17-23-22-21-27
4	18-24-30-29-28-27
5	18-24-23-22-21-27

TABLE 5. Normalized indicator value.

No.	L_i	INE_i	T_i	H_i	C_i	P_i
1	0.3678	0.6065	1	0.3678	0.4493	0
2	1	1	0.7165	0.5991	0.6703	0.4865
3	0.6907	1	0.3678	1	1	0.6321
4	0.6658	1	0.7165	0.5991	0.6703	0.3934
5	0.5965	0.3678	0.5134	0.6768	0.3678	0

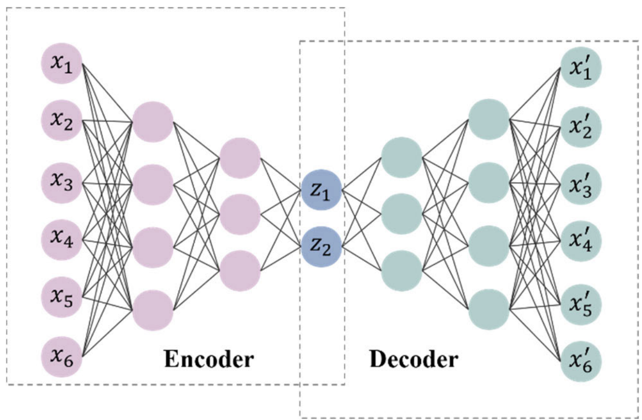


FIGURE 6. Structure of auto-encoder model.

Figure 7 shows the indicator mapping values of five candidate trajectories. According to the proposed indicator value normalization method, the larger the decision indicator value is, the more likely the candidate path is to be the path truly chosen by the traveler. It can be seen that point 3 is farthest from the origin in the two-dimensional coordinate system, which means its indicator value is optimal, so candidate path 3 is selected as the reconstructed trajectory. The complete vehicle trajectory is shown in Figure 8.

V. VERIFICATION

In order to verify the effectiveness of the proposed method, we test the vehicle trajectory reconstruction accuracy for different number of missing nodes and different ALPR device coverage rate, respectively. We randomly select 500 complete paths with the number of nodes greater than

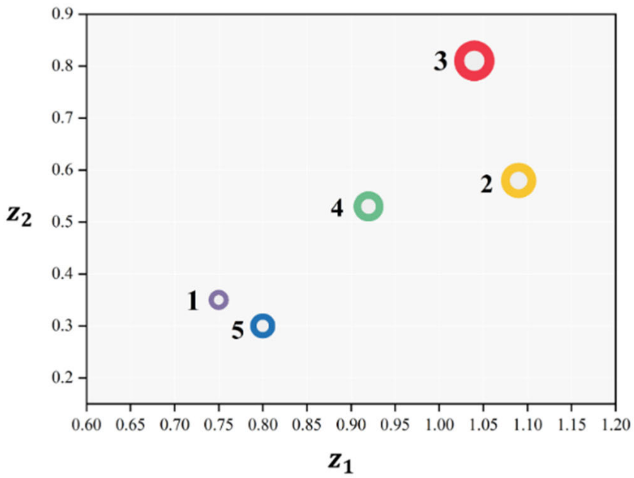


FIGURE 7. Mapped indicator values of five candidate paths.

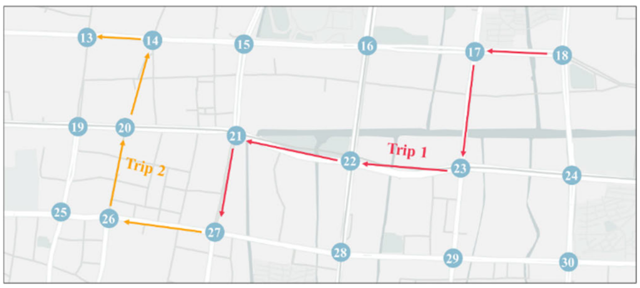


FIGURE 8. Reconstructed trajectory.

seven from the historical data set with the verification ratio of 15%, which constitute the real complete path set $QT = \{QT_1, QT_2, \cdots, QT_i, \cdots QT_{500}\}$.

And for the verification of reconstruction accuracy for different number of missing nodes, we randomly deleted $j(j = 1, 2, 3, 4, 5)$ continuous nodes of each path QT_i in set QT to obtain the incomplete path set $QU = \{QU_1, QU_2, \cdots, QU_i, \cdots QU_{500}\}$. We reconstruct the trajectories in the set QU using principal component analysis (PCA), the shortest path (SP) algorithm, and the auto-encoder method proposed in this study, and the accuracy results are shown in Figure 9.

It can be seen from Figure 9 that the accuracy of the three methods decreases with the increase of the number of missing nodes. The accuracy of the SP algorithm is still acceptable when only one and two nodes are missing, and declines rapidly since missing three nodes. And the accuracy is less than 40 % when there are five nodes. The possible reason is that when there are few missing nodes, the real path is close to the shortest path. And when the number of missing nodes increases, the SP algorithm only considers the path length as an influencing factor, which does not correspond to the actual choice of the traveler.

Comparing the performance of the proposed method in this study and the PCA algorithm, it can be seen that when the number of missing nodes is 1 and 2, the accuracy of the two methods is very close, which is maintained at

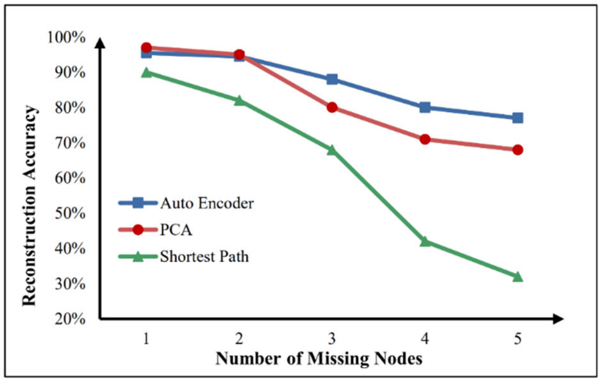


FIGURE 9. Reconstruction accuracy for different number of missing nodes.

more than 90%. However, when the number of missing nodes reaches 3 or more, the accuracy of the auto-encoder is higher, maintaining above 80%, and the comprehensive accuracy reaches 85%. While the accuracy of principal component analysis decreased to 70%. The possible reason is that as the number of missing nodes increases, there are several candidate paths with close path length, and the traveler will consider several influencing factors when choosing the path. There is a non-linear correlation between these influencing factors. Principal component analysis can only represent linear transformation in data mapping, while auto-encoder can also represent nonlinear transformation, and its information loss is much smaller than that of PCA. Therefore, when the number of missing nodes increases, the proposed method in this study has higher accuracy.

On the other hand, since the research area selected in this study is an open road network, it will cause some fixed errors. We found that the origin and destination points of most trajectories are located at the boundary of the road network, which means that these paths may be truncated by the boundaries of the study area in reality, and the state of the peripheral road network is not considered in the trajectory reconstruction process, so the accuracy of trajectory reconstruction will be affected.

Further, we tested the accuracy of vehicle trajectory reconstruction at different ALPR coverage rates. We randomly deleted 10 %, 20 %, 30 %, 40 %, 50 % and 60 % of the vehicle passing information recorded by ALPR device in the complete path set QT to simulate the cases where the ALPR coverage rates are 90 %, 80 %, 70 %, 60 %, 50 % and 40 %, respectively. Figure 10 shows the reconstruction accuracy of each path at different ALPR coverage rates. Figure 11 shows the average trajectory reconstruction accuracy at different coverage rates.

It can be seen from Figure 10 that the accuracy of trajectory reconstruction decreases with the decrease of ALPR coverage rate. When the coverage rate is between 60% and 90%, the fluctuation of accuracy is small, indicating that the algorithm is stable and reliable. When the coverage rate is between 40% and 50%, the accuracy fluctuates greatly, indicating that the algorithm performs poorly at a low detector

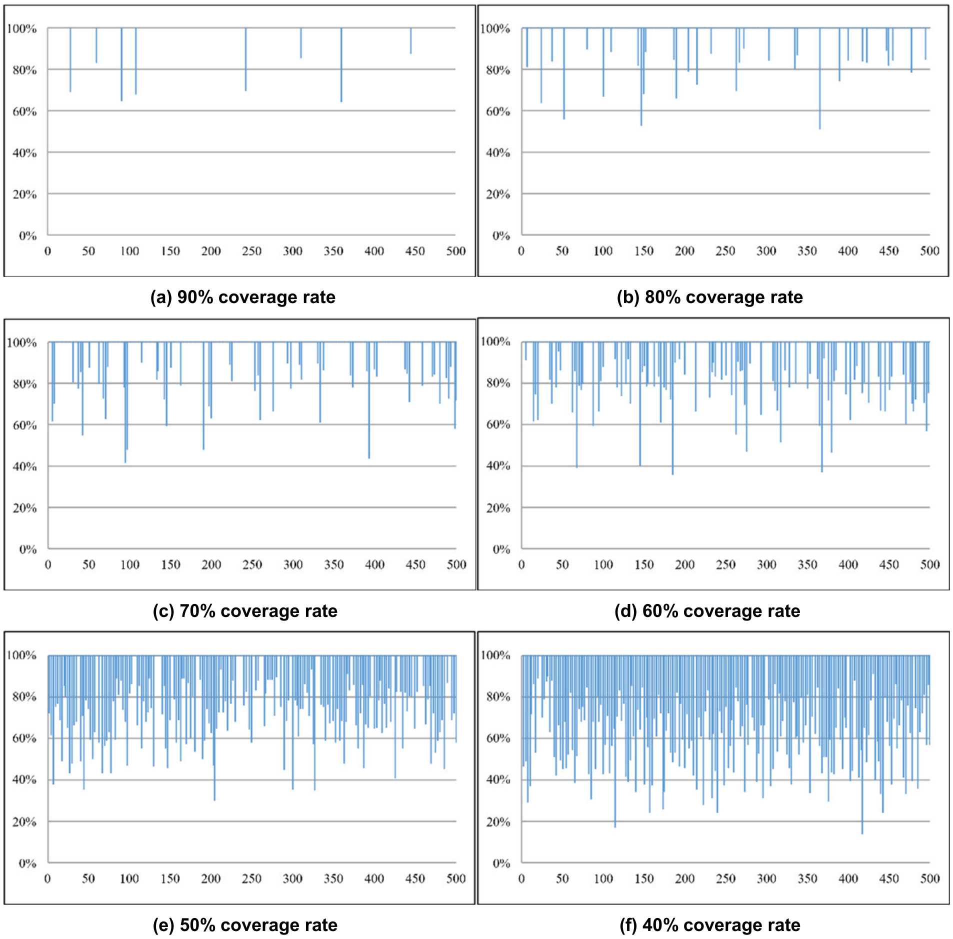


FIGURE 10. Reconstruction accuracy at different ALPR coverage rate. (The x-axis represents 500 trajectories, and the y-axis represents reconstruction accuracy.)

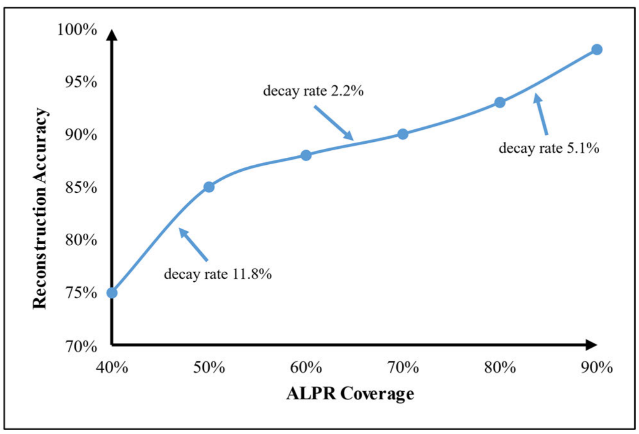


FIGURE 11. Average accuracy at different ALPR coverage rate.

coverage rate. Figure 11 illustrates that when the detector coverage is equal to or greater than 50%, the average accuracy exceeds 85%, and when the coverage is 40%, the accuracy decreases to 75%. Moreover, when the coverage rate is reduced to 50 %, the attenuation accelerates. Overall, the verification results show that the proposed trajectory reconstruction method has high accuracy and good stability.

VI. CONCLUSION

This study proposed a new method of vehicle trajectory reconstruction on urban traffic network based on license plate recognition data. Firstly, we split the trip chain extracted from ALPR data based on travel time threshold. Then we designed an improved KSP algorithm to generate candidate paths for incomplete vehicle trajectory, which improved the searching efficiency. Finally, on the basis of six decision indicators, we utilized auto-encoder model to select the optimal trajectory.

This method was tested on a realistic traffic network for different number of missing nodes and different ALPR coverage rates. The verification results show that the comprehensive accuracy of the method exceeds 85% and the algorithm is robust and reliable when the ALPR coverage rate exceeds 60%. Thus, the proposed method can realize high-precision trajectory reconstruction and perform well in realistic traffic scenarios.

Although the verification method can effectively evaluate the accuracy and robustness of the trajectory reconstruction algorithm, the limitation of it is that the complete trajectory as ground-true data cannot include the intersections on the road network that are not equipped with ALPR device. Therefore, if the probe car GPS data collected during the same period with ALPR can be used as ground-true data, it can serve as a better and more reasonable verification. Because probe car GPS data records the trajectories of constantly moving vehicles, it can cover almost all intersections on the road network.

In future work, more explorations can be made from as follows: (1) This study will use multi-source data as a supplement to the limited observations of ALPR devices. For instance, taxi GPS trajectory data and mobile phone signaling data can be exploited to provide more consecutive initial vehicle trajectory and more precise travel time [33], [34]. (2) We will improve the algorithm to enhance the accuracy of vehicle trajectory reconstruction especially for low ALPR coverage rate: a) More activation function will be tried to improve the model performance such as tanh function; b) The actual meaning of the hidden layer of the auto-encoder model still needs further study. Meanwhile, we will try more auto-encoder model structures, such as introducing regularization. (3) Since ALPR equipment does not fully cover all intersections of the road network, we cannot obtain the average travel time of all paths to estimate the travel time of candidate trajectories. In this paper, we exploited a regression model to simplify the problem. And we will adopt GAN and LSTM model based on the combination of vehicle GPS data and ALPR data to improve the accuracy of travel time estimation. (4) It is noted that the time-saving rate of the proposed P-KSP algorithm is 19% compared with SP algorithm, which seems not a great improvement. The reason is that the scale of the selected traffic network is not large enough and the network conditions are not complex enough. We will apply the algorithm to a more large-scale and complex network to validate its performance. (5) The ALPR device can identify which approach the vehicle is going through. This characteristic will be used to reduce the candidate trajectory set and then improve the calculation efficiency. (6) The reconstructed vehicle trajectory will be used to estimate the path flow and OD demand to realize real-time traffic management.

REFERENCES

[1] Q. Gao, F. Zhang, R. Wang, and F. Zhou, "Trajectory big data: A review of key technologies in data processing," *J. Softw.*, vol. 28, no. 4, pp. 959–992, 2017.

[2] M. Li, Z. Li, C. Xu, and T. Liu, "Short-term prediction of safety and operation impacts of lane changes in oscillations with empirical vehicle trajectories," *Accident Anal. Prevention*, vol. 135, Feb. 2020, Art. no. 105345.

[3] K. Tang, S. Chen, Z. Liu, and A. J. Khattak, "A tensor-based Bayesian probabilistic model for citywide personalized travel time estimation," *Transp. Res. C, Emerg. Technol.*, vol. 90, pp. 260–280, May 2018.

[4] C. Wang, C. Xu, and Y. Dai, "A crash prediction method based on bivariate extreme value theory and video-based vehicle trajectory data," *Accident Anal. Prevention*, vol. 123, pp. 365–373, Feb. 2019.

[5] C. Wang, C. Xu, J. Xia, Z. Qian, and L. Lu, "A combined use of microscopic traffic simulation and extreme value methods for traffic safety evaluation," *Transp. Res. C, Emerg. Technol.*, vol. 90, pp. 281–291, May 2018.

[6] X. Xie, H. van Lint, and A. Verbraeck, "A generic data assimilation framework for vehicle trajectory reconstruction on signalized urban arterials using particle filters," *Transp. Res. C, Emerg. Technol.*, vol. 92, pp. 364–391, Jul. 2018.

[7] C. Xu, J. Zhao, and P. Liu, "A geographically weighted regression approach to investigate the effects of traffic conditions and road characteristics on air pollutant emissions," *J. Cleaner Prod.*, vol. 239, Dec. 2019, Art. no. 118084.

[8] Z. Liu, Y. Liu, Q. Meng, and Q. Cheng, "A tailored machine learning approach for urban transport network flow estimation," *Transp. Res. C, Emerg. Technol.*, vol. 108, pp. 130–150, Nov. 2019.

[9] C. F. Daganzo, "A variational formulation of kinematic waves: Basic theory and complex boundary conditions," *Transp. Res. B, Methodol.*, vol. 39, no. 2, pp. 187–196, Feb. 2005.

[10] B. Mehran, M. Kuwahara, and F. Naznin, "Implementing kinematic wave theory to reconstruct vehicle trajectories from fixed and probe sensor data," *Transp. Res. C, Emerg. Technol.*, vol. 20, no. 1, pp. 144–163, Feb. 2012.

[11] D. Bachir, G. Khodabandelou, V. Gauthier, M. El Yacoubi, and J. Puchinger, "Inferring dynamic origin-destination flows by transport mode using mobile phone data," *Transp. Res. C, Emerg. Technol.*, vol. 101, pp. 254–275, Apr. 2019.

[12] J. W. C. Van Lint, "Empirical evaluation of new robust travel time estimation algorithms," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2160, no. 1, pp. 50–59, Jan. 2010.

[13] D. Ni and H. Wang, "Trajectory reconstruction for travel time estimation," *J. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 113–125, Aug. 2008.

[14] B. Coifman, "Estimating travel times and vehicle trajectories on freeways using dual loop detectors," *Transp. Res. A: Policy Pract.*, vol. 36, no. 4, pp. 351–364, May 2002.

[15] J. W. C. Van Lint and S. P. Hoogendoorn, "A robust and efficient method for fusing heterogeneous data from traffic sensors on freeways," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 25, no. 8, pp. 596–612, Nov. 2010.

[16] A. Sharma, Z. Zheng, and A. Bhaskar, "A pattern recognition algorithm for assessing trajectory completeness," *Transp. Res. C, Emerg. Technol.*, vol. 96, pp. 432–457, Nov. 2018.

[17] E. Castillo, J. M. Menéndez, and P. Jiménez, "Trip matrix and path flow reconstruction and estimation based on plate scanning and link observations," *Transp. Res. B, Methodol.*, vol. 42, no. 5, pp. 455–481, Jun. 2008.

[18] Y. Feng, J. Sun, and P. Chen, "Vehicle trajectory reconstruction using automatic vehicle identification and traffic count data," *J. Adv. Transp.*, vol. 49, no. 2, pp. 174–194, Mar. 2015.

[19] J. Yang and J. Sun, "Vehicle path reconstruction using automatic vehicle identification data: An integrated particle filter and path flow estimator," *Transp. Res. C, Emerg. Technol.*, vol. 58, pp. 107–126, Sep. 2015.

[20] W. Rao, Y.-J. Wu, J. Xia, J. Ou, and R. Kluger, "Origin-destination pattern estimation based on trajectory reconstruction using automatic license plate recognition data," *Transp. Res. C, Emerg. Technol.*, vol. 95, pp. 29–46, Oct. 2018.

[21] L. Wei, Y. Wang, and P. Chen, "A particle filter-based approach for vehicle trajectory reconstruction using sparse probe data," *IEEE Trans. Intell. Transp. Syst.*, early access, Mar. 9, 2020, doi: 10.1109/TITS.2020.2976671.

[22] H. Yu, S. Yang, Z. Wu, and X. Ma, "Vehicle trajectory reconstruction from automatic license plate reader data," *Int. J. Distrib. Sensor Netw.*, vol. 14, no. 2, p. 13, Feb. 2018.

[23] S. Ruan, F. Wang, D. Ma, S. Jin, and D. Wang, "Vehicle trajectory extraction algorithm based on license plate recognition data," *J. ZheJiang Univ. (Eng. Sci.)*, vol. 52, no. 5, pp. 836–844, 2018.

[24] Z. Zhang, H. Liu, L. Rai, and S. Zhang, "Vehicle trajectory prediction method based on license plate information obtained from video-imaging detectors in urban road environment," *Sensors*, vol. 20, no. 5, p. 1258, Mar. 2020.

[25] X. Feng, H. Sun, J. Wu, Z. Liu, and Y. Lv, "Trip chain based usage patterns analysis of the round-trip carsharing system: A case study in Beijing," *Transp. Res. A: Policy Pract.*, vol. 140, pp. 190–203, Oct. 2020.

[26] H. J. Miller, "A measurement theory for time geography," *Geographical Anal.*, vol. 37, no. 1, pp. 17–45, Jan. 2005.

[27] J. Y. Yen, "Finding the K shortest loopless paths in a network," *Manage. Sci.*, vol. 17, no. 11, pp. 712–716, 1971.

[28] S. Zhong, L. Zhou, S. Ma, and N. Jia, "Effects of different factors on drivers' guidance compliance behaviors under road condition information shown on VMS," *Transp. Res. A: Policy Pract.*, vol. 46, no. 9, pp. 1490–1505, Nov. 2012.

[29] D. Li, C.-J. Jin, M. Yang, and A. Chen, "Incorporating multi-level taste heterogeneity in route choice modeling: From disaggregated behavior analysis to aggregated network loading," *Travel Behav. Soc.*, vol. 19, pp. 36–44, Apr. 2020.

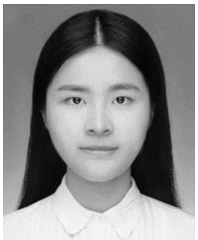
[30] C. Sun, L. Cheng, S. Zhu, F. Han, and Z. Chu, ''Multi-criteria user equilibrium model considering travel time, travel time reliability and distance,'' *Transp. Res. D, Transp. Environ.*, vol. 66, pp. 3–12, Jan. 2019.

[31] G. E. Hinton, ''Reducing the dimensionality of data with neural networks,'' *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[32] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, ''Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion,'' *J. Mach. Learn. Res.*, vol. 11, no. 12, pp. 3371–3408, Dec. 2010.

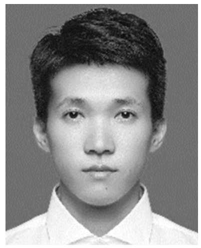
[33] P. Wang, J. Lai, Z. Huang, Q. Tan, and T. Lin, ''Estimating traffic flow in large road networks based on multi-source traffic data,'' *IEEE Trans. Intell. Transp. Syst.*, early access, Apr. 29, 2020, doi: 10.1109/TITS.2020.2988801.

[34] P. Wang, C. Wang, J. Lai, Z. Huang, J. Ma, and Y. Mao, ''Traffic control approach based on multi-source data fusion,'' *IET Intell. Transp. Syst.*, vol. 13, no. 5, pp. 764–772, May 2019.



XINYI QI received the bachelor's degree in traffic engineering from Southeast University, in 2020, where she is currently pursuing the master's degree in transportation engineering. Her research interests include traffic safety, travel behavior analysis, and public transportation.

YANJIE JI received the Ph.D. degree from Southeast University, in 2007. She is currently a Professor with the School of Transportation, Southeast University. Her research interests include transportation planning and control, intelligent transport systems, travel behavior analysis, and data-driven transit operations.



WENHAO LI received the M.S. degree in transportation engineering from the Guilin University of Electronic Technology, in 2019, and the M.S. degree in computer information technology from Northern Arizona University. He is currently pursuing the Ph.D. degree in transportation engineering with the School of Transportation, Southeast University. His research interests include transportation planning and control, data analytics, and intelligent transport systems.

SHUICHAO ZHANG received the Ph.D. degree from Southeast University, in 2012. He is currently an Associate Professor with the School of Civil and Transportation Engineering, Ningbo University of Technology. His research interests include transportation planning and management and transportation design.