A Novel Method for Road Intersection Construction From Vehicle Trajectory Data

JIANBO TANG1,2, MIN DENG2, JINCAI HUANG2,3, AND HUIMIN LIU2

1State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, Wuhan 430079, China
2Department of Geo-Informatics, Central South University, Changsha 410083, China
3Shenzhen Key Laboratory of Spatial Smart Sensing and Service, Shenzhen University, Shenzhen 518060, China

Corresponding author: Jincai Huang (huangjincaicsu@163.com)

This work was supported in part by the National Science Foundation of China under Grant 4173000380, Grant 41730105, Grant 41771492, and Grant 41601424, in part by the National Key Research and Development Program of China under Grant 2017YFB0503500, and in part by the Cooperative Research and Development Project of BOTONG Smart City Research Institute under Grant BTZH2018002.

ABSTRACT

Road intersection is an important part of the road network, and it plays a key role in many intelligent transportation applications. Currently, the generation of road maps required expensive field surveying and labor-intensive mapping. With more and more public vehicles equipped with positioning devices, massive trajectory data are available. These trajectory data provides a new way to generate and update urban road networks in real time. Although numerous methods are currently available to reconstruct road maps from GPS trajectories, most of these methods focus on the generation of road segments (i.e., road centerlines or road lanes). Road intersections are simply represented by nodes, the generation of road intersections with detailed structures still remains a challenge. Therefore, this paper proposes a novel method for constructing road intersections from vehicle GPS trajectories. First, the boundaries of road intersections are detected by turning angles clustering. Then, the entrances and exits of each intersection are identified based on a newly developed method. Finally, the detailed geometric structures of intersections are reconstructed based on the detected entrances and exits, and turning rules are further extracted and assigned to each intersection. The experiments on real-world trajectory data show that the proposed method can generate the detailed geometric structures of road intersections as well as the turning rule semantic information successfully.

INDEX TERMS

Intelligent transportation system, GPS trajectory, road intersection construction, turning rules.

I. INTRODUCTION

The road map is a basic part of geographical information and plays a key role in many intelligent transportation applications. Professional road networks are produced in labor-intensive ways (such as field surveying), which face the problems of long update cycles and the lack of up-to-date semantic information (such as traffic turning rules). These became the most important defects in numerous applications such as Cartography, Location-Based Services (LBS), and other intelligent transportation applications [1], [2]. Today, with a large number of floating vehicles installed with GPS recorders, a large number of trajectories are recorded to obtain vehicle location, driving direction, speed, and other information. The recorded GPS trajectories portray numerous geometric and semantic details that are beneficial to construct road networks. Under this premise, researchers have made significant attempts to generate road maps [3]–[9], and extract road semantic information such as traffic modes, road levels, speed limits, and turning rules [10]–[12] from GPS trajectories.

Among these road map construction studies, road intersections are usually not finely mapped. Each road intersection is represented as a single node that connects different road links when these algorithms are implemented [3]–[9], [13]–[16]. However, real-world road intersections have very complicated geometry structures (Figure 1), for example, overpasses (Figure 1-A), radial shaped intersections (Figure 1-B), roundabouts (Figure 1-C). A single node can only represent the location of a road intersection while the detailed structures of
the road intersection are not represented, which will reduce the application range of these methods in fields such as Cartography, and other smart transportation applications.

In summary, not enough attempts have been tried to detect the detailed structures of road intersections. Recently, the development of road intersection has received some attention [17]–[20]. Wang’s research [21] went one step further to detect the traffic rules and travel paths of simple road intersections (such as crossroads, T-junctions, Y-junctions), in their research, high-quality differential GPS trajectory data was used to generate the geometry structures and traffic rules of crossroads, T-junctions, Y-junctions. However, in fact, almost 60% of the floating car trajectories are collected at very low frequency [22], it is still a challenge to make full use of these low-frequency trajectories to construct road intersections with detailed structures. Therefore, in this paper, a novel method for road intersection construction is proposed based on low-frequency GPS trajectories. We aim to construct the geometric structures of inner paths (see Figure 1) of a road intersection as well as the topological information of these paths, i.e. road connection and traffic turning rules.

The following section provides the literature review. Section 3 gives a detailed description of the proposed method. Experimental results and discussions are presented in Section 4. Conclusions are given in Section 5.

II. LITERATURE REVIEW

Recently, there has been a surge in road map construction [5] from vehicle-generated GPS trajectories. These road map construction algorithms include rasterization-based methods [8], [23], [24], incremental methods [3], [7], [25], [26], clustering-based methods [27], [28] and other hybrid methods [16]. Among these algorithms, road intersections are considered as nodes that connect roadways, and the roadways are generated from trajectories by a wide range of algorithms, such as clustering [27], [28], physical attraction model [7], [25]. Usually, most of the generated road maps have no directions, this means that roadways with opposite directions are expressed as centerlines. To obtain a fine and directed road map, recently, Wang [29] adopted a divide-and-conquer strategy and combine several algorithms to generate road intersections and roadways separately from high-quality GPS trajectories, Tang [30] proposed a road map refinement method based on weighted Delaunay triangulations. In these researches, directed road maps are generated. At the same time, the possibility of more detailed and fine road map (lane-level road map) generation from GPS trajectories has been studied [27], [31]. Uduwaragoda [32] introduced a kernel density estimation model to calculate the density peak of the trajectory across a roadway surface to estimate the number and centerline of lanes. Yang et al. [33] studied the lane changes of road maps from GPS trajectories, especially for road intersections.

Among these road map construction studies, road intersections were not conducted in-depth studies. However, road intersections are the key elements in a road map [34], especially for a directed and detailed road map. The topological consistency of a road map is mainly determined by the structures of road intersections, as they connect different roadways [21], [34]. Early studies mainly considered road intersections as nodes, for example, the rasterization-based methods [8], [23], [24], incremental methods [3], [7], [25], [26], clustering-based methods [27], [28] and other hybrid methods [16]. Some studies attempted to detect the spatial locations and boundaries of road intersections [17], [18], [35] using GPS trajectories. These studies assumed that the turning angles at intersections are larger than those of roadways. Xie et al. [20] proposed a novel method based on the longest common subsequence (LCSS) to identify the connecting points at intersections. There are also researches trying to generate more detailed structures of the road intersections, for example, Deng [36] proposed a low-frequency trajectory hierarchical clustering method to generate detailed road intersections, however, how to select the cluster number still remains a challenge. The most similar to our method is Wang’s research [29]. In their method, the inner path of a road intersection is represented by one raw trajectory and road intersections are mapped based on high-quality GPS trajectories. However, at present, almost 60% of floating vehicle trajectories are collected at low frequency [22]. Using low-frequency GPS trajectories to construct the detailed structures of road intersections (see in Figure 1) is the objective of this study.

In summary, recent studies mainly focused on detecting the spatial locations and boundaries of road intersections. The generation of road intersections with detailed geometry structures and turning rules still remain a big challenge. Therefore, this paper proposes a novel method for detecting the detailed structures of road intersections from low-frequency vehicle trajectories. The following section will introduce the proposed method in detail.

III. METHODOLOGY

A road map consists of intersections and roadways. Each road intersection is connected to several roadways to constitute a routable road network [21], [34]. These connected roadways are usually well separated. When a vehicle passes through an intersection, it will follow specified traffic rules and controlled by traffic lights. Therefore, on the one hand, large turning angles tend to be generated at intersections, which are spatially aggregated in the intersection space, on the
other hand, based on the ‘connecting’ role of intersections, the entrances/exits points where vehicles trajectories pass through intersection boundaries can be detected, afterward, the travel paths and turning rules can be inferred. This paper thus proposes a novel framework for constructing the detailed structures of road intersections, as shown in Figure 2.

A. PRE-PROCESSING
At present, almost 60% of floating vehicle trajectories are recorded at low frequency [22]. These GPS trajectory datasets are unavoidably subject to system errors or random errors owing to multipath effects or signal drifting [37]. Hence, a trajectory filtering process needs to be implemented to eliminate GPS noise.

Since the loss of GPS signals, some large space and/or time gaps between two continuous sample points exist. In this paper, we divided the single trajectory into different parts according to the sampling interval and the distance between two continuous sample points of the trajectory. Let $T = \{P_1, P_2, \ldots, P_N\}$ denote a GPS trajectory. $P_i$ is a sample point that is a tuple defined as follows:

\[ P_i = (x_i, y_i, t_i, v_i, o_i) \]  

where $(x_i, y_i)$ are the coordinate values of the sample point, $t_i$ is the timestamp, $v_i$ and $o_i$ are the velocity and the vehicle heading at the sample point, respectively. Each trajectory $(T)$ is checked whether it has abnormal edges (i.e. $(T)$ conditions:

1) Time gap:

\[ t_{i+1} - t_i > \sigma_T \]  

2) Space gap:

\[ \| P_i, P_{i+1} \| > k(v_i + v_{i+1})(t_{i+1} - t_i)/2 \]

where $\sigma_T (\sigma_T = 60$s) and $k (k = 1.5)$ are constants. If exists, the trajectory $T$ is divided into two sub-trajectories by cutting off the edge $(P_i, P_{i+1})$, as shown in Figure 3-A.

Additionally, as shown in Figure 3-B, trajectories with abnormal turns are unlikely to reflect the road geometry because of some off-road behaviors (such as passing through gas stations or parking lots, etc.). Therefore, trajectories which contain more than $N_{turn}$ ($N_{turn}$ is set to 4 in this paper) continuous GPS points with an abnormally turning angle larger than $50^\circ$ were eliminated in this paper.

B. DETECTION OF INTERSECTION BOUNDARIES
The difference between road intersections and roadways is that when vehicles pass through a road intersection, relatively larger turning angles are generated. Nevertheless, due to the low-frequency sampling of low-frequency trajectory data, it is inefficient and non-robust to find all the GPS points located at road intersections purely using a specific turning angle threshold. In fact, GPS points with large turning angles tend to congregate at road intersections with strong spatial autocorrelation [29]. Therefore, a hotspot analysis is implemented to detect the spatial aggregation of turning angles. Since the local $G^*$ statistic in [38] is commonly used to detect the statistical significant hotspots of spatial data, in this paper, we use the local $G^*$ statistic to identify the spatial aggregation pattern of GPS points with large turning angles. According to Ord’s research [38], hotspots represent the spatial aggregation of GPS points with high local $G^*$ values and the opposite of cold spots.

In this paper, a hotspot analysis to detect the spatial boundaries of road intersections is implemented according to Deng’s research [36]. Let $h_1 = |o_{i+1} - o_i|$ be the turning angle of a sample point $P_i$. The local $G^*$ statistic of $P_i$ is then calculated as follows:

\[ G^*_i = \frac{\sum_{j=1}^N w_{ij} h_j - \bar{h} \sum_{j=1}^N w_{ij}}{S \sqrt{\left[ N \sum_{j=1}^N w_{ij}^2 - (\sum_{j=1}^N w_{ij})^2 \right] / (N - 1)}} \]

\[ \bar{h} = \frac{1}{N} \sum_{j=1}^N h_j, \quad S = \sqrt{\frac{1}{N} \sum_{j=1}^N h_j^2 - \bar{h}^2} \]

where $w_{ij}$ is the spatial weight, $N$ is the total number of sample points, $d(P_i, P_j)$ is the space distance between points $P_i$ and $P_j$, and $\epsilon$ is a constant ($\epsilon = 0.001$).

A higher positive $G^*$ indicates spatial aggregation of neighboring GPS points of $P_i$ compared with a randomly distributed spatial pattern with a significance-level called $p$-value. The significance level ($p = 0.01, G^*_i > 2.58$) is used to detect the hotspots of GPS points with large turning angles (see Figure 4-A). The point $P_i$ that has a positive $G^*_i$ value larger than 2.58 ($p = 0.01$) is defined as a hotspot point. Then, an adaptive spatial clustering method based on Delaunay triangulations named ASCDT [39] is used to partition the hotspot points into separate clusters (i.e. road intersections) to
capture the spatial aggregation pattern of hotspot points. The ASCDT algorithm is an adaptive spatial clustering algorithm that can detect clusters of different shapes and densities, and it is suitable for detecting road intersections with different shapes and sizes [39]. Firstly, the Delaunay triangulation is constructed, and the global and local long edges are hierarchically removed according to the statistical features of the edges of the Delaunay triangulation. The remaining connected sub-graphs are defined as spatial clusters. In this process, three parameters need to be set to adjust the clustering results: global long edge parameter $\alpha$, local long edge parameter $\beta$, and $minpts$ is the minimum number of the points in a cluster. In general, a larger value of $\alpha$ or $\beta$ will generate more compact clusters. In this paper, $\alpha$ and $\beta$ are both set to 5, and $minpts$ is set to 20 by default. As shown in Figure 4-B, the hotspot point clusters (i.e., road intersections) detected using the ASCDT algorithm are displayed in different colors. Then, the spatial boundaries of road intersections can be estimated with the smallest circumscribed circles of the hotspot point clusters (as the red circles shown in Figure 4-B).

C. ENTRANCES/EXITS IDENTIFICATION

Based on the detected spatial boundaries of road intersections, the entrances/exits of intersections are further identified. Considering the “connecting” role of road intersections, each intersection connects several roadways, called branch roadways in this paper. Generally, each branch roadway contains an entrance and an exit. Vehicles usually enter an intersection at an entrance of a branch roadway and exit along another branch roadway. Therefore, each vehicle trajectory will generate an entrance and an exit at an intersection. Based on this thought, intersecting points of a vehicle trajectory and the spatial boundary of a road intersection are calculated and defined as the candidate entrance and exit points. When a number of vehicles enter the same intersection, the candidate entrance/exit points will clearly stay together in space since the branch roadways connecting a road intersection are usually well separated with different directions. Therefore, the ASCDT algorithm is used to partition the candidate entrance and exit points into clusters. The centers of entrance and exit point clusters are used to represent the precise locations of entrances/exits of the branch roadways. The detailed implementation of this strategy is described as follows.

1) CANDIDATE ENTRANCE/EXIT POINTS EXTRACTION

Let $T = \{P_1, P_2, \ldots, P_N\}$ be a GPS trajectory, and $P_i$ ($i = 1, 2, \ldots, N$) be the GPS sample point. $\{P_i, P_{i+1}\}$ is defined as a trajectory section. Suppose that the spatial boundaries of road intersections are defined as $C = \{c_1, c_2, \ldots, c_M\}$, with each intersection boundary is a tuple that contains the spatial coordinate sequences. A trajectory $T$ may pass several road intersections and intersect with the spatial boundaries of these intersections. For one trajectory section $\{P_i, P_{i+1}\}$, it may intersect with one or more road intersection boundaries as well. For example, in Figure 5, $\{P_i, P_{i+1}\}$ of $T_j$ intersects with two intersection boundaries. However, for each road intersection, the trajectory section $\{P_i, P_{i+1}\}$ has only three types of intersecting patterns as shown in Figure 5:

1) $\{P_i, P_{i+1}\}$ enters an intersection with a candidate entrance point $e_i$;
2) $\{P_i, P_{i+1}\}$ exits an intersection with a candidate exit point $e_j$;
3) $\{P_k, P_{k+1}\}$ passes an intersection with a candidate entrance point $e_k$ and a candidate exit point $e_l$.

Based on the spatial relationship between $\{P_i, P_{i+1}\}$ and a specific road intersection boundary, for all road intersections in the study area, an iterative procedure is proposed in Algorithm 1 to extract the candidate entrances/exits on a trajectory section $\{P_i, P_{i+1}\}$.

For all trajectory sections, candidate entrances/exits points can be extracted using Algorithm 1. These candidate entrance/exits points are recorded as $E$, which is a tuple to store the information of candidate entrances/exits points defined as follows:

$$E = (e_1, e_2, \ldots, e_K)$$
$$e_i = (tid_i, cid_i, x_i, y_i, tp_i)$$

(5)

where $K$ is the number of candidate entrance/exits points; $tid$ and $cid$ are the index of the trajectory and the index of the road intersection to which $e_i$ belongs, respectively; $x$ and $y$ donate the spatial coordinates of the candidate entrance/exit point; and $tp$ indicates whether $e_i$ is a candidate entrance point ($tp = 0$) or a candidate exit point ($tp = 1$). Moreover, each trajectory can be divided into internal segments (yellow lines in Figure 5) and external segments (cyan lines in Figure 5).
Algorithm 1 Candidate Entrance/Exit Identification Procedure

Input: \( \{P_i, P_{i+1}\}; \ C \) (a set of spatial boundaries of road intersections)
Output: candidate entrances; exits

Step1: For each spatial boundary \( e_i \) in \( C \), if \( e_i \) intersects with \( \{P_i, P_{i+1}\} \), then insert \( e_i \) into a list \( C_m \).

The number of spatial boundaries in \( C_m \) is denoted as \( m \).

Step2: if \( m = 1 \)

Step3: Intersecting \( C_m \) with \( \{P_i, P_{i+1}\} \), push the intersecting points into \( e = \{e_1, \ldots, e_n\} \)

Step4: if \( n = 1 \)

Step5: \( P_i \sim \) in \( C_m \) and \( P_{i+1} \sim \) in \( C_m \): entrances = [entrances; \( e \)]

Step6: \( P_i \sim \) in \( C_m \) and \( P_{i+1} \sim \) in \( C_m \): exits = [exits; \( e \)]

Step7: if \( n = 2 \)

Step8: \( \text{dist}(e_1, P_i) < \text{dist}(e_2, P_i) \): entrances = [entrances; \( e_1 \)]; exits = [exits; \( e_2 \)]

Step9: \( \text{dist}(e_1, P_i) > \text{dist}(e_2, P_i) \): entrances = [entrances; \( e_2 \)]; exits = [exits; \( e_1 \)]

Step10: if \( m > 1 \)

Step11: for \( s = 1 \sim m \)

Step12: Repeat Step2 \sim Step8 for \( C_s \)

FIGURE 6. Entrance/exit points of the branch roadways.

FIGURE 7. Turning rules identification.

D. ROAD INTERSECTION CONSTRUCTION

1) TURNING RULES IDENTIFICATION

Generally, vehicles have different turning modes at intersections. Vehicles usually enter a road intersection from the entrance of a branch roadway and then leave at the exit of another branch roadway. The turning behaviors of vehicles at road intersections are constrained by turning rules. Thus, internal trajectories and \( E \) provide valuable information for turning rules identification. In this paper, we extract the turning rules based on the entrances and exits.

For each road intersection, all internal trajectories are classified based on the candidate entrance/exit point clusters. A turning rule is identified when internal trajectories enter an intersection at the same entrance and leave at the same exit. If a road intersection connects \( K \) branch roadways, there are \( K^2 \) possible turning rules. As shown in Figure 7, the trajectories (black lines) enter the intersection at the entrance of branch roadway \( b \) and leave at the exit of branch roadway \( d \), representing a turning rule ‘from \( b \) to \( d \)’.

2) GEOMETRIC STRUCTURES RECONSTRUCTION

Currently, most existing methods to generate the detailed structures of road intersections are based on high-quality GPS trajectories. For example, Wang [29] simply select one trajectory connecting the entrance and the exit to represent the underlying travel path. However, owing to GPS positioning errors and low sampling frequency, one trajectory may not correctly reflect the exact structure of the road path. For example, in Figure 8-A and C, trajectories are sparsely sampled and show geometric inconsistency with the true travel path. It is a challenge to fit the travel paths of different turning modes based on low-frequency GPS trajectories. However, the discrete GPS points of each turning mode seem to demonstrate better geometric characteristics of the road skeleton than the trajectories themselves. In this paper, the principal curve of the discrete GPS points is obtained by using a K-segment curve fitting algorithm in [40]. The K-segment principal curve is defined as a self-consistent smooth curve that passes through the middle of the distribution of an observed point dataset. The curve has the ability to accurately...
to fit a self-intersecting point pattern, as shown in Figure 8-D. It first initializes a line segment as the principle curve $f$ of $3\sigma$ (the standard deviation of points, $k = 1$) and assigns all GPS points to Voronoi regions of these $k$ line segments. Afterward, another line segment is iteratively inserted into $f$ ($k = k + 1$), and $f$ is modified as the $k$ first principle curves fitted by the track points in all Voronoi regions. The objective function is then optimized, and the $k$ line segments are linked as a polyline based on the greedy path search method until $k > k_{max}$ ($k_{max}$ is set to 24 in this paper). Details of the K-segment curve fitting algorithm can be found in [40].

Finally, by fitting the trajectories of each turning rule using the K-segment principal curve fitting, the detailed geometric structures of road intersections are obtained. Through the above steps, road intersections with detailed geometric structures as well as the topological information (i.e. road connection and traffic turning rules) can be constructed based on the low-frequency trajectory data.

IV. EXPERIMENTAL ANALYSIS

A. DATASETS

To evaluate the effectiveness of the proposed method, an area on the south bank of the Yangtze River (about 13.0 km × 11.0 km) in Wuhan, China, was selected (Figure 9). The area is densely populated and contains complicated types of road intersections which are suitable for our experiment. Floating car trajectories were collected from May 1, 2014 to May 3, 2014 and the data contains 1,129,466 GPS sample points. After the data pre-processing, the input trajectory data consists of 28,354 tracks with a total length of 58,305km (average: 2.06km and standard deviation: 1.45km). The trajectories comprise from 5 up to 160 GPS sample points with a sampling rate of 1-300s (average: 49.24s and standard deviation: 27.82s). Figure 9 shows the study area and the spatial distribution of the trajectories (gray lines).

![FIGURE 9. Study area and the spatial distribution of the trajectories.](image-url)

B. INTERSECTION BOUNDARIES DETECTION

1) DETECTING INTERSECTION BOUNDARIES

The hotspot analysis and point clustering are implemented to obtain the spatial boundaries of road intersections in the study area. As shown in Figure 10-A, hotspots of sample points with large turning angles are detected with a significance level of $p = 0.01$. It can be seen that the hotspot analysis can successfully distinguish intersections and roadways. From Figure 10-A, it can be seen that hotspots are well recognized, but outliers (i.e. some sparse sample points with large turning angles) are also detected. Figure 10-B shows the hotspot point clustering result by the ASCDT algorithm. Each cluster corresponds to an individual road intersection, and the smallest circumscribed circle (the red circle) of the cluster represents the spatial boundary of the road intersection. By employing the point clustering, the outliers are removed finally and the underlying road intersections where hotspot points are aggregated are identified. As we can see, there are various types of road intersections in the study area such as T-junctions, overpasses, Y-junctions, roundabouts, radial intersections, and other intersections with complex structures. The shapes and geometric structures of these intersections are complicated and diverse, and the sizes and distances between pairs of intersections are varied. Road intersections with different shapes, sizes, and distances are successfully detected by the proposed method.

2) QUANTITATIVE EVALUATION

Nevertheless, there are also some intersections that are incorrectly detected as road intersections, as shown in Figure 10 (yellow circles), and some others are not detected (blue circles) because of the spatiotemporal heterogeneity of low-frequency GPS trajectories. To quantitatively evaluate the effectiveness of intersection detection by the hotspot analysis and point clustering, the precision, recall, and F-value are calculated:

$$\text{Precision} = \frac{\text{Correctly detected}}{\text{Correctly detected} + \text{Incorrectly detected}}$$

$$\text{Recall} = \frac{\text{Correctly detected}}{\text{Correctly detected} + \text{Not detected}}$$

$$F - \text{Value} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$
By manually observing the underlying satellite images and the road network data obtained from OpenStreetMap (OSM), 162 road intersections were detected in the study area. By comparing the detected road intersections using the proposed method with the manually recognized intersections, 136 intersections are correctly detected among all 157 detected road intersections, 21 are incorrectly detected, and 26 are undetected. The precision, recall, and F-value are 86.6%, 84.0%, and 85.3%, respectively. The precision, recall, and F-value indicate the validity of intersection detection by using the hotspot analysis and point clustering.

To construct the detailed structures of road intersections from the trajectories, the number of trajectory tracks that are available for each intersection is very relevant, and generally, more trajectory tracks are available for an intersection, the more detailed and precise structures can be reconstructed. Figure 11 shows the distribution of the number of trajectory tracks that are available for each intersection detected by our method.

From Figure 12, it can be seen that most of the road intersections are correctly reconstructed in spite of the complexity and diversity of these intersections. In Figure 12, the entrances are dotted as green boxes, the exits are dotted as blue boxes, and the yellow line represents the travel path of one turning rule from the entrances to the exits. By overlay analysis with satellite imagery, the detailed structures of the road intersections are all correctly reconstructed from the GPS trajectories, the shapes and positions are consistent with the underlying roads in satellite images.

The constructed detailed structures of road intersections (A–F in Figure 12) are shown in Figure 13. We can find that the reconstructed road intersections help to build a routable road map.

Table 1 lists the detected turning rules of each intersection in Figure 13. Taking into account the diversity of the shapes of intersections, traffic turning rules are not expressed...
**TABLE 1.** Turning rules of road intersections A-F.

<table>
<thead>
<tr>
<th>Intersections</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Turning rules</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. c→e</td>
<td>5. e→c</td>
<td>13. a→d</td>
<td>15. c→b</td>
</tr>
<tr>
<td>2. e→b</td>
<td>6. b→e</td>
<td>7. c→a</td>
<td>10. c→b</td>
</tr>
<tr>
<td>3. a→d</td>
<td>8. a→e</td>
<td>8. d→a</td>
<td>14. c→d</td>
</tr>
<tr>
<td>4. c→a</td>
<td>9. d→c</td>
<td>9. b→c</td>
<td>9. e→b</td>
</tr>
<tr>
<td>5. a→c</td>
<td>10. d→a</td>
<td>10. b→c</td>
<td>10. d→b</td>
</tr>
<tr>
<td>6. d→b</td>
<td>11. a→e</td>
<td>11. b→a</td>
<td>11. d→b</td>
</tr>
<tr>
<td>7. b→a</td>
<td>12. a→b</td>
<td>12. b→a</td>
<td>12. c→a</td>
</tr>
</tbody>
</table>

**D. RESULTS COMPARISON AND EVALUATION**

To further evaluate the effectiveness of our method, the results of Wang's method [29] is also compared, as shown in Figure 14.

**FIGURE 14.** Comparison of the proposed method and Wang’s method.

From Figure 14, we can see that Wang’s method correctly reconstructed most of the road intersections. However, for several complicated intersections, the travel paths are incorrectly extracted owing to GPS errors and the low sampling rate, for example, path c→b of intersection A, path a→b of intersection D, and path c→b of intersection E, as shown in Figure 14. For these intersections, our proposed method generates more robust results, as the blue lines indicate in Figure 14.

Further, we evaluate the proposed method by calculating the geometric accuracy with the maps derived from OSM. This is actually a graph comparison problem for calculating the distance between two graphs. Recently, the path-based distance [4], [5], [35] has been popular in evaluating the quality of the reconstructed road maps. Our goal is to compute the geometrical deviations (distances) between each point of the generated travel paths and the corresponding point on OSM (not the distance between paths). Therefore, a modified path-based distance is used to calculate the distances between each pair of travel paths in OSM and the generated map. First, the distance between a point \( p_i \) of a travel path and the corresponding point on OSM is defined as the minimum of \( d_j, d_{j+1}, \) and \( d_p \), as shown in Figure 15. \( O_j O_{j+1} \) is the nearest line segment of \( p_i \) on OSM, and \( d_p \) is the distance between \( p_i \) and the perpendicular foot on \( O_j O_{j+1} \). Then, the geometrical deviation of each point of the generated travel paths is calculated as a preliminary evaluation of the experimental results.

Finally, the travel paths of both Wang’s method and the proposed method are resampled at 1 m because the points of the travel paths in these two methods are not the same. A linear interpolation algorithm is employed to generate the dense resampled points from the input travel paths. Then, the geometrical deviations of both methods are calculated, as shown in Table 2.

**TABLE 2.** Geometrical deviations of Wang’s method and the proposed method.

<table>
<thead>
<tr>
<th>Intersection</th>
<th>The proposed method</th>
<th>Wang’s method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>min (m)</td>
<td>max (m)</td>
</tr>
<tr>
<td>a</td>
<td>0.0034</td>
<td>33.1573</td>
</tr>
<tr>
<td>b</td>
<td>0.0004</td>
<td>26.4409</td>
</tr>
<tr>
<td>c</td>
<td>0.0006</td>
<td>26.0450</td>
</tr>
<tr>
<td>A</td>
<td>0.0002</td>
<td>20.9647</td>
</tr>
<tr>
<td>D</td>
<td>0.0005</td>
<td>138.2070</td>
</tr>
<tr>
<td>E</td>
<td>0.0005</td>
<td>31.4276</td>
</tr>
</tbody>
</table>

From Table 2, we can see that our method outperforms Wang’s method as the mean and the maximum geometrical deviations are smaller than in Wang’s method. In general,
our method has a mean geometry precision of 4–10m compared with underlying roads from OSM, and the maximum geometrical deviation is mostly between 20m and 30m. However, the maximum geometrical deviation of Wang’s method reaches from 28 up to 50m. This is caused by the sparseness of the low-frequency trajectory sampling. The geometrical precision of intersection D is unusual because the underlying OSM roads in the black box in Figure 14 are missing. In addition, it should be noted that there is no left or right travel path in the OSM road network data, which leads to the geometrical deviation calculated in Table 2 being relatively higher than the actual value.

Figure 16 shows a geometric deviation histogram of the intersection A, which clearly shows the advantages of the proposed method in terms of geometric accuracy compared to Wang’s method. This indicates that the proposed algorithm has the advantage of extracting the principal travel paths from low-frequency trajectories.

![FIGURE 16. Geometric deviation with OpenStreetMap of intersection A.](image)

**E. DISCUSSION**

From the experimental results, we can see that the proposed method can effectively construct the detailed structures and turning rules of the road intersections from the input low-frequency trajectory data. However, in this paper, the K-segment principal curve fitting algorithm is implemented to fit the travel paths of the road intersection from trajectories, and the number of trajectory tracks that are available for each intersection is very relevant. Generally, more trajectory tracks are available for an intersection, the more detailed and precise structures can be reconstructed by the proposed method. Figure 17 shows some constructed results when a different number of trajectory tracks for a road intersection is used.

It can be seen that when the trajectories are too sparse at a road intersection, the constructed paths of the road intersection may be not correct because of the large curve fitting errors by employing the principle curve fitting algorithm. Therefore, in order to ensure the effectiveness of the proposed method, the number of trajectories that are available for a road intersection should not be too small (e.g. less than 10 tracks). Through the experimental analysis on two typical intersections with a different number of trajectories for intersection construction (see Figure 17), when more than twenty trajectories are available for each path of the road intersection, the proposed method can reconstruct the paths consistent with the underlying roads. The quantitative relationship between the quality of the intersection details and the minimum number of trajectories needs further investigation. In addition, benefiting from the fact that trajectory data is a typical stream data, the amount of trajectory data accumulates over time. Therefore, by collecting the available GPS trajectory data at intersections with a longer time period (e.g. a week or a month), the detailed geometric structures of the travel paths of the road intersections could be well reconstructed using the proposed method.

![FIGURE 17. Geometry structure generating results using different number of trajectories.](image)

**V. CONCLUSIONS**

Road intersection is an important part of the road network. For constructing the detailed structures of road intersections for intelligent transport applications (such as car navigation and route planning), this paper proposed a novel method for road intersection construction based on low-frequency trajectory data. Low-frequency GPS trajectory data is currently widely collected by vehicles equipped with GPS receivers. Since the low sampling rate and noise of low-frequency GPS trajectory data, it still remains a challenge to make full use of this data to construct road intersections with detailed structures. In the proposed method, the hotspot analysis and point clustering are first employed to detect the spatial boundaries and the entrance/exit points of intersections. Then, based on the detected entrance/exit points, turning rules of the intersections are identified, and the detailed geometric structures of travel paths of the intersection are reconstructed from trajectories passing through the entrance/exit points. Experimental results on real-world floating car trajectories showed that the proposed method can successfully reconstruct the road intersection details. Compared with the state-of-the-art alternative algorithm (i.e. Wang’s method),
the proposed method generated better results. Our contributions are: 1) the proposed method detects the detailed structures of road intersections using common low-frequency trajectory data with low sampling rate and noise; 2) an automatic algorithm is proposed to detect the entrance and exit points of the road intersections; 3) the proposed method can construct both the geometric structures as well as the topological information (including road connection and the traffic turning rules) of road intersections; 4) a modified path-based distance measure is defined for quantitative evaluation of the reconstructed road maps.

The proposed method is based on the low-frequency trajectory data, which is not able to generate high-detailed road structures such as the lanes. The detection of lanes at the road intersections will be our future attention and as well as the detection of the travel time, traffic congestions and traffic events for dynamic transportation mapping.

REFERENCES