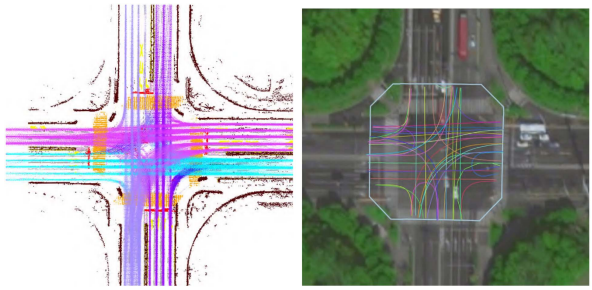
Traffic Flow-Based Crowdsourced Mapping in Complex Urban Scenario

Tong Qin, Haihui Huang, Ziqiang Wang, Tongqing Chen, and Wenchao Ding, *Member, IEEE*

***Abstract*—An accurate road topological structure is of great importance for autonomous driving in complex urban environments. Currently, most autonomous vehicles highly rely on the High-Definition map (HD map) to cruise across the city. Without the prior map, it's hard for vehicles to find right-turning and left-turning ways in large intersections. However, due to the complexity of intersections, producing such a map by human resources is time-consuming and error-prone. In this letter, we proposed a framework to automatically produce the topological map of complicated intersections. This framework adopts the crowdsourcing way to collect semantic information about the environment and traffic flows. The topological structure is inferred from traffic flows correctly and automatically. We highlight that this framework is highly automatic and scalable, which can greatly speed up HD map production and decrease the cost. The proposed system is validated by real-world crowdsourcing data and the result is comparable to the traditional HD maps.**

***Index Terms*—Mapping, crowd-sourcing, traffic flow.**



(a) The crowdsourcing data: semantic map and traffic flows. (b) The generated structure of the intersection.

Fig. 1. Left picture shows the crowdsourcing data, which consists of traffic flows and the semantic map of the environment. The right picture shows the topological structure generated by the proposed system automatically. The road structure is aligned with Google Maps for visualization.

I. Introduction

AUTONOMOUS driving has been among the most popular topics in the past few years. A large number of vehicles with assisted driving capability cruise on the highway automatically, where the road structure is clear and simple. However, it is quite challenging for them to drive in the urban scenario, due to the complicated road structures, such as left turn, right turn, and u-turn. It's difficult to precisely navigate the ego-vehicle and predict the behavior of other vehicles without a clear road structure. Therefore, the High-Definition map (HD map) [1] is crucial for autonomous vehicles in the urban scenario, which helps vehicles recognize road structure.

The HD map is a highly accurate map, including map elements such as road shape, road marking, traffic signs, and barriers in centimeter-level accuracy, which are used for autonomous driving. Maintaining high accuracy is one of the biggest challenges in building HD maps of real-world roads. Hence, producing such a prior map is label-intensive and time-consuming. Since there is no ground marking line at the center area of the intersection, the topological structure of the intersection is difficult to make. It requires a lot of manual annotation and imagination to build the topological structure of a large intersection, which is prone to error. How to build the HD map quickly, correctly, and at a low cost are of great importance to the industry. More and more researches focus on building HD map automatically.

With more and more commercial vehicles equipped with onboard sensors, such as cameras and s, crowdsourcing is a quick and scaleable way to collect road structure information. By leveraging massive vehicles with onboard sensors, extensive road surface information can be gathered in a short time. Besides the stable environment, the dynamic information, traffic flow, can also be collected in this crowdsourced way, as shown in Fig. 1. There is an old saying that the path is shown up only when thousands of people walk through. The traffic flow is effective information for inferring the topological relationship in the intersection. By gathering the driving trajectories of ego and other vehicles, the traffic reflects the topological relationship at the intersection automatically. The major contributions are summarized as follows:

- A crowdsourcing framework for topological map production in complex urban scenarios.
- An algorithm that generates road topology automatically from the traffic flow in intersections.

Manuscript received 8 January 2023; accepted 19 June 2023. Date of publication 3 July 2023; date of current version 12 July 2023. This letter was recommended for publication by Associate Editor S. Thakar and Editor A. Banerjee upon evaluation of the reviewers' comments. *(Corresponding author: Wenchao Ding.)*

Tong Qin is with the Global Institute of Future Technology, Shanghai Jiao Tong University, Shanghai 20001, China (e-mail: qintonguav@gmail.com).

Haihui Huang, Ziqiang Wang, and Tongqing Chen are with the IAS BU, Huawei Technologies, Shanghai 200200, China (e-mail: huanghaihui3@huawei.com; wangziqiang@huawei.com; chentongqingg@huawei.com).

Wenchao Ding is with the Academy for Engineering and Technology, Fudan University, Shanghai 200001, China (e-mail: dingwenchao@fudan.edu.cn).

This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2023.3291507, provided by the authors.

Digital Object Identifier 10.1109/LRA.2023.3291507

- Comprehensive analysis from real-world experiments with massive crowdsourcing data.

We highlight that the proposed system is a highly automatic framework, which can solve the key problem, recognition of intersection structure, in autonomous driving tasks.

The rest of the letter is organized as follows. The related work is reviewed in Section II. The detailed methodology is presented in Section III. Experimental comparison and validation are conducted in the Section IV. Finally, the letter is concluded in Section V.

II. Related Work

Recognizing the topological structure of the road and lane is of great importance to autonomous driving. Prediction and planning are heavily dependent on such topological structure to infer ego and other vehicles' trajectories. Therefore, many studies have been conducted to extract invisible lanes and connect topology at intersections. These methods can be categorized into three groups.

Pre-made maps (HD Map): By leveraging high-precision sensors, including an industry camera, a Global Navigation Satellite System (GNSS), an Inertial Measurement Unit (IMU), a Light Detection and Ranging(Lidar) sensor, and an onboard computing platform, the lane-level HD map could be produced in [2]. In [3], the topological structure of the intersection was generated by fitting straight lines between entry points and exit points. The geometry of virtual lanes was further modified by cubic spline. Given width, slope, and direction inside the intersection, the geometry of the lane was addressed by piece-wise cubic spline and control points in [4]. The virtual lane generated by these methods was usually different from human driving trajectories.

Online perception-based methods: Recently, Neural networks were widely applied to infer road structure for autonomous driving online. HDMapNet [5] used a neural network to predict the map of the surroundings, which looked like the HD Map. Furthermore, the vector topological relationship could be obtained after post-processing. VectorMapNet [6] and MAPTR [7] were based on HDMapNet to relieve the post-processing. Vectorized results can be obtained directly through the network in an end-to-end way. By incorporating a Standard Definition (SD) prior map, MapLite [8] used CNN and distance transforms to build the HD map online. However, since there was open space without any visible road markings, these methods failed to model the road structure inside the intersection.

Experience (crowdsourcing) based methods: The traffic flow of vehicles has been commonly used for HD map generation. In [9], Joshi et al. used Bayesian Graphical Models and traffic flows to obtain the connection relationship between the entry and exit points in the intersection. However, the open street map was needed to provide the approximate position of the intersection and the center line. In [10], the particle filter was used to construct the geometry and topology of urban roads. In the intersection, the open street map was needed as a prior, and the quadratic Bessel curve was used to fit the road, which was not human-like. in [11], Zhao et al. used traffic flow to correct and update the existing intra-intersection topology of the map. The semantic geometric information of the map within the intersection cannot be obtained by only using the traffic flow. In [12], a crowdsourcing framework was proposed. The semantic road information from multiple vehicles was fused. However, it cannot generate the topological model of the intersection. In [13], Kim et al. proposed a method that can quickly update HD maps by crowdsourced data, but only focus on geometric updates without topological updates. In [14], Zhang et al. combined semantic segmentation and visual SLAM to carry out crowdsourcing updates and achieved vectorization through a traditional clustering method. Because it was based on elements such as road signs and road dividers, it cannot solve the area without road markings in the intersection. Although experience (crowdsourcing) based methods heavily rely on the data, they have the great potential to build the lane-level structure of the intersection.

In this letter, we propose a crowdsourcing framework for topological map production in complex urban scenarios. Compare with traditional HD Map and other crowdsourcing methods, the proposed method is a low-cost solution, which dramatically reduces the cost of map production in the following two aspects:

- Instead of using professional mapping vehicles with expensive sensors, the proposed system relies on mass-productive vehicles, which decrease the cost of hardware and data collection.
- The proposed system is a highly automatic framework to generate topological map in urban scenarios, which relieves the most labor-intensive part of map production.

Furthermore, the proposed system takes advantage of human experiences, which extracts reference lines from humans' trajectories. The output is more reliable and human-like.

III. Methodology

As shown in Fig. 2, the proposed crowdsourcing framework consists of three parts: on-vehicle data collection, on-cloud data alignment, and topology generation.

A. On-Vehicle Data Collection

Nowadays, massive commercial vehicles equipped with onboard sensors, such as IMU, GPS, wheel-encoder, and cameras, drive at any time and everywhere. IMU, GPS, and wheel-encoder can provide localization information. Cameras can provide stable environmental and dynamic object perception. A large amount of data, including road structure and traffic flow, can be obtained in a short time at a low cost. In this work, two types of information are collected from crowdsourcing vehicles. The one is the semantic point cloud of the road surface. The other one is the traffic flow.

1) Local Semantic Map: Similarly with RoadMap [12], a local semantic map is built on the vehicle. The image captured by the front-view camera is mainly used for semantic feature extraction. The other surrounding cameras can be also used if there are any.

Firstly, a CNN-based segmentation method, such as [15], [16], [17], is adopted to segment the raw image. As shown in Fig. 3(a) and (b), the image is segmented into multiple classes, such as ground, lane line, stop line, road marker, curb, vehicle, bike, and human. The segmentation is robust to the light. Therefore, the

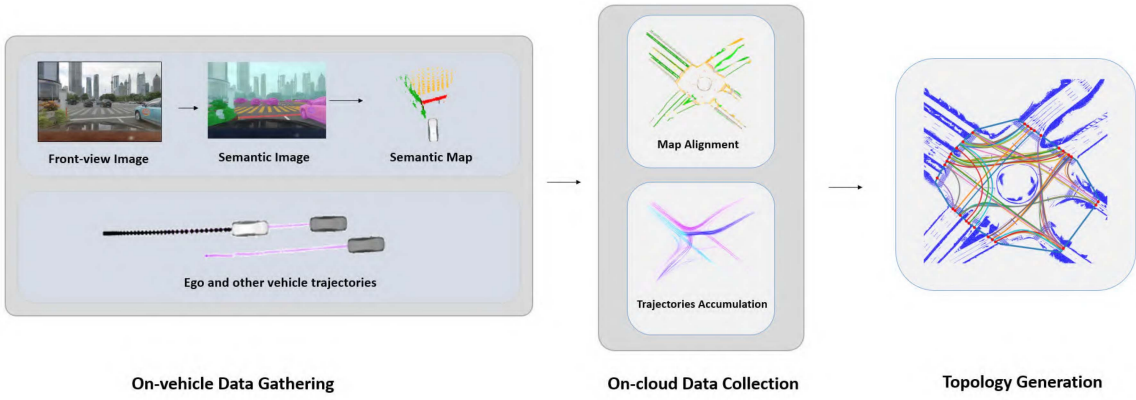
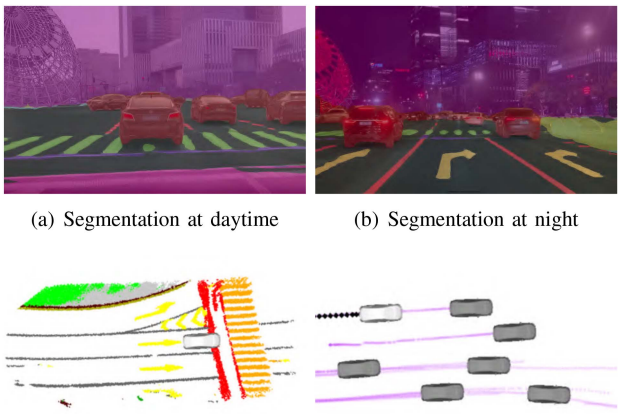


Fig. 2. Illustration of the proposed framework. It consists of three parts: on-vehicle data collection, on-cloud data alignment, and topology generation.



(a) Segmentation at daytime (b) Segmentation at night

(c) Semantic map of surroundings (d) Traffic flows

Fig. 3. Illustration of on-vehicle data collection.

crowdsourcing data captured at any time can be used. Among these classes, ground, lane line, stop line, and road marker are used for semantic mapping. After segmentation, the semantic pixels are inversely projected from the perspective image plane to the ground plane according to the intrinsic and extrinsic calibration. By leveraging the odometry, the semantic points from multiple frames are stacked together to build a local semantic map, with the resolution of 0.1 m, as shown in Fig. 3(c).

2) Traffic Flow Layer: At the same time, onboard sensors such as Radar, and cameras are used to detect and track surrounding vehicles. In detail, some widely-used 3D-detection algorithms, such as Center-net [18], Point-Pillars [19], Detr-3D [20], can provide real-time pose of other vehicles relative to the ego-vehicle. By tracking them for a while, a traffic flow is formed, which contains the trajectories of multiple vehicles, including the ego-vehicle. Since object detection is noisy, we filter out unstable and short-tracking trajectories. A sample of the traffic flow layer is shown in Fig. 3(d). The semantic map and traffic flow layer are uploaded to the cloud for further fusion.

B. On-Cloud Map Alignment

Since GPS only achieves meter-level accuracy, the data captured by different vehicles can't be aligned perfectly. If we merge them directly under the global coordinate, the map would be blurry. Therefore, a pose alignment procedure is necessary before data fusion. We adopt the popular NN-based method, Geometric Transformer [21], to optimize poses from different vehicles by semantic point cloud matching. This method learns geometric features for robust super point matching, which is robust to low-overlap cases. An illustration of map alignment is shown in Fig. 4.

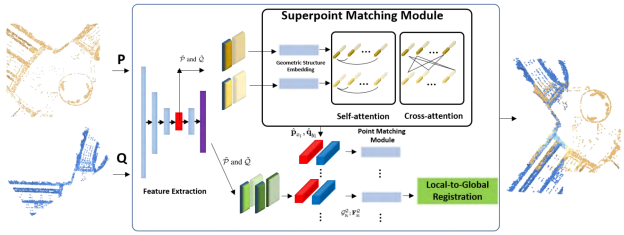


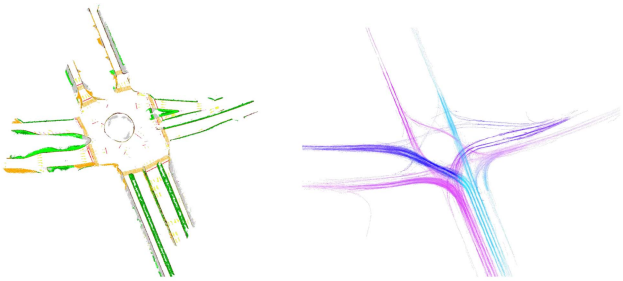
Fig. 4. Local semantic maps collected by different vehicles are aligned together by geometric transformer network.

To be specific, inputs of the transformer are 25 m $\times$ 25 m local semantic map and 50 m $\times$ 50 m global semantic map (thousands of semantic points with a resolution of 0.1 m). The output from the network is the aligned pose between the local coordinate and the global coordinate. Furthermore, we adjust the traffic flow layer according to the aligned pose.

All pieces of the crowdsourcing data are aligned into one global coordinate. The semantic map becomes more complete, while the traffic flows become denser. The perfectly aligned semantic map and traffic flow of one intersection are shown in Fig. 5(a) and (b) respectively.

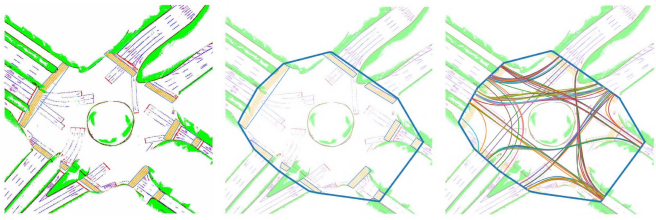
C. Topology Generation for Intersections

In this section, we generate the road topology of intersections automatically. Firstly, we determine the coverage of the intersection, which is represented by polygons. Secondly, the traffic flow is divided into groups according to the same start and endpoints. Finally, the topology is generated from the connectivity of traffic flows. Furthermore, the most human-like reference path is recommended which can guide the vehicle to go cross the intersection.

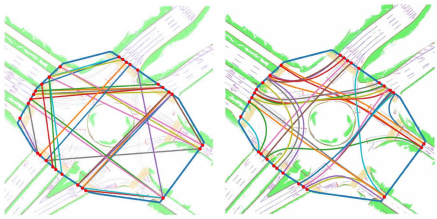


(a) Global semantic map (b) Global traffic flow

Fig. 5. Results of on-cloud map alignment. All pieces of crowdsourcing data are aligned into a global coordinate.



(a) semantic BEV image (b) intersection polygon (c) groups of the traffic flows

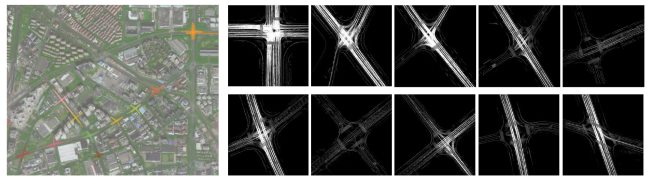


(d) topological structure of the intersection (e) reference lines selection

Fig. 6. Topology Generation for Intersections.

1) Intersection Polygon Generation: The first step is to determine where is the intersection. An approximate seed point can be got from the public navigation map. For each center point, we convert neighboring semantic point clouds to a Bird's Eye View (BEV) image, as shown in Fig. 6(a). There are different types of semantic points on the BEV image. We use DBSCAN to group pixels whose property is the crosswalk and stop line. Furthermore, we use a convex polygon to fit each crosswalk group, and we use the line to fit each stop line group. Finally, the convex hull of these elements is the polygon of the intersection, as shown in Fig. 6(b).

2) Traffic Flow Grouping: The traffic flow is truncated according to the intersection polygon, as shown in Fig. 6(c). The truncated points that point inward are noted as start points, while the truncated points that point outward are noted as endpoints. Since the start points and end points are as many as the number of traffic flows, we adapt the mean shift grouping method to further cluster start and endpoints. Since the lane width is about 3 meters as usual, we use 3 meters as the parameter for a cluster. The clustered start points set is denote as $\mathscr{S} = \{\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_m\}$ and clustered ending points set is $\mathscr{E} = \{\mathbf{q}_0, \mathbf{q}_1, \ldots, \mathbf{q}_n\}$.



(a) (b)

Fig. 7. Overview of crowdsourcing data. (a) A satellite image of the city block. The colorful areas are the ROI (intersections), where we collect crowdsourcing data. (b) The outlook of traffic flow layer on intersections.

3) Topology Generation: According to the connectivity of the traffic flow, we can build an adjacency matrix $\mathbf{A}_{mn}$ which describes the connection relationships between start points $\mathscr{S}$ and end points $\mathscr{E}$. For example, $\mathbf{A}_{mn}$ looks like:

$$\begin{array}{c} \\ \mathbf{q_0} \\ \mathbf{q}_1 \\ \vdots \\ \mathbf{q}_n \end{array} \begin{array}{c} \begin{array}{cccc} \mathbf{p}_0 & \mathbf{p}_1 & \cdots & \mathbf{p}_m \end{array} \\ \begin{bmatrix} 1 & 0 & \ldots & 1 \\ 1 & 1 & \ldots & 1 \\ \vdots & \vdots & \ddots & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \end{array} \tag{1}$$

$\mathbf{A}_{ij}$ equals to 1 means $\mathbf{p}_i$ and $\mathbf{q}_j$ are connected, otherwise disconnected. A topology of the intersection is shown in Fig. 6(d). Meanwhile, we can keep one trajectory, which has a minimal distance from others. as a recommended path between the connected start point and end point. These recommended paths are smooth and naturally human-like since it comes from human drivers. A sample of the reference line is shown in Fig. 6(e)

IV. Experiment

We conduct experiments with real-world crowdsourcing data within a city block in Shanghai, China, as shown in Fig. 7(a). The crowdsourcing data contains over a hundred trips collected by dozens of vehicles. The data were captured within one week including morning and night. The data contains various intersections in different locations with different numbers of lanes and different shapes, as shown in Fig. 7(b). Among these, there is a super large and complicated intersection with an island in the middle. It's challenging even for humans to label the topology inside the intersection. The manually labeled HD Map is treated as the ground truth. The topology generated by the proposed method is compared with HD Map. Quantitative and qualitative analyses were carried out to verify the effectiveness of the method.

Hardware: The mass-production vehicles we used for crowdsourcing data collection were equipped with front-view cameras and Huawei MDC 610. Only a small part of the computation resource was required for semantic segmentation and data uploading on each vehicle. The semantic segmentation network runs at 20 FPS onboard. On the cloud service, the workstation with Intel Xeon Gold 6134 CPU and Nvidia Quadro P5000 GPU was used, which can perform semantic map registration at 20 FPS. Since the interval of building a local semantic map is

TABLE I
RECALL AND PRECISION COMPARING WITH HD-MAP

Intersection Id	ADE (average)[m]	MDE (max)[m]	precision[%]	recall[%]
0	2.20	3.81	100.00	53.85
1	1.93	2.69	100.00	80.00
2	1.77	3.43	83.33	72.22
3	2.41	4.60	80.00	62.50
4	3.07	4.61	80.48	72.91
5	2.40	3.28	79.16	60.00
6	1.74	2.96	100.00	100.00
7	0.44	0.44	100.00	66.67
8	0.70	1.11	75.00	62.50
9	2.13	3.15	82.35	68.42

around 5 seconds, one workstation can handle the data captured by 100 vehicles in time.

A. Comparing With HD Map

The manually labeled HD Map was treated as the ground truth. Firstly, we compared the completeness and correctness of the topological structure.

We used ADE (Average Displacement Error) and MDE (Maximum Displacement Error) to evaluate correctness compared to the HD map. The ADE and MDE were defined as follows,

$$ADE = \frac{1}{N_{ij}} \sum_{i} \sum_{j \in i} |p^{i,j}_{flow} - p^{i,j}_{hd}|$$
$$MDE = \max_{i,j} |p^{i,j}_{flow} - p^{i,j}_{hd}| \quad (2)$$

where i is iterated over all connection links. For each reference line, we select one point for every 5-meter interval, which j refers to. p_{flow} refers to the point from our method, while p_{hs} refers to the point from HD map.

We used recall and precision to evaluate completeness metrically. The recall and precision were defined as follows,

$$recall = \frac{N_{true\_proposals}}{N_{HD}}$$
$$precision = \frac{N_{true\_proposals}}{N_{true\_proposals} + N_{false\_proposals}} \quad (3)$$

where N_{HD} meant the number of all connection links inside the HD Map, $N_{true\_proposals}$ and $N_{false\_proposals}$ meant the number of right and wrong connection links from the proposed method respectively.

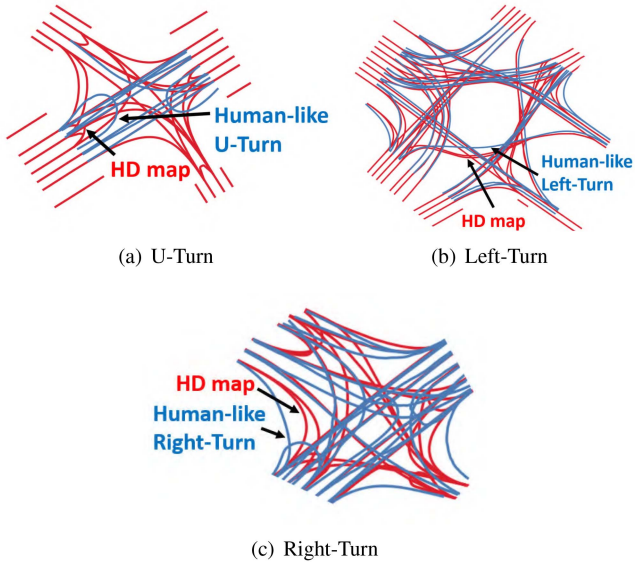
The result was shown in Table I. It can be seen that the proposed method achieved high precision and not-very-high recall. Reasons for the not-very-high recall were that:

- The limited scale of crowding sourcing data failed to cover everywhere inside the intersection.
- There were some unreasonable links that human drivers never passed. For example, changing multiple lanes when going straight at the intersection.

Furthermore, we compared the geometric accuracy of the proposed reference line against the HD Map. We evaluated the Average Distance Error (ADE) and Max Distance Error (MDE) between the proposed reference lines and HD Map lines. The result was shown in Table I. It can be seen that in most cases, the automatically generated reference line from traffic flow was close to the HD map (the average distance below 3 m), which validated the reliability of the proposed method.

TABLE II
COMPARING WITH ORIGINAL TRAFFIC FLOW

Intersection Id	ADE[m]		MDE[m]	
	Hd map	proposed	Hd map	proposed
0	3.53	**0.7**	6.77	**1.14**
1	2.33	**0.26**	4.97	**0.53**
2	1.97	**0.34**	3.97	**1.12**
3	4.80	**0.27**	9.71	**0.80**
4	3.17	**0.41**	6.69	**0.93**
5	2.19	**0.48**	4.43	**1.06**
6	1.82	**0.36**	3.74	**0.88**
7	2.06	**0.36**	4.09	**0.65**
8	2.30	**0.28**	4.45	**0.78**
9	2.54	**0.25**	4.53	**0.55**



(a) U-Turn (b) Left-Turn

(c) Right-Turn

Fig. 8. Samples of human-like reference lines. The red lines are the reference line from HD Map, while the blue lines come from the proposed system.

Fig. 9. U-turn shape is special from human drivers, which is bulb shape.

B. Comparing With Traffic Flow

For each connected link, we recommend one reference line between the start point to the endpoint. In this section, we compare how well it fits the traffic flow against the reference line from HD Map. We evaluated the average distance between the recommended line to the traffic flows (multiple human-driving

TABLE III
EVOLUTION OF TOPOLOGICAL RESULTS WHEN THE NUMBER OF CROWDSOURCING TRIPS INCREASES FROM 5 TO 100

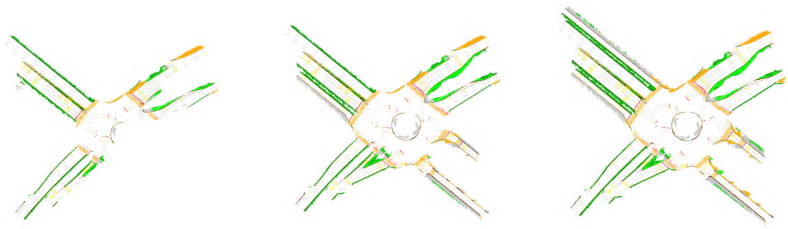
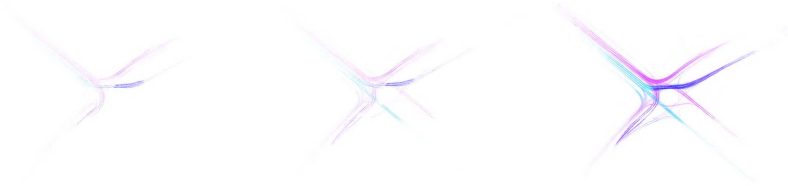
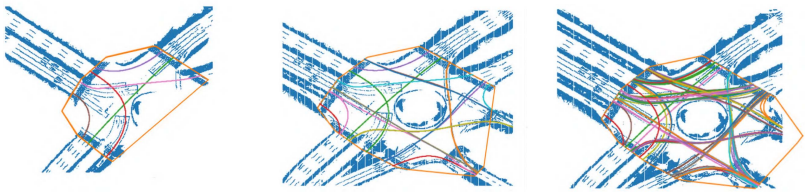
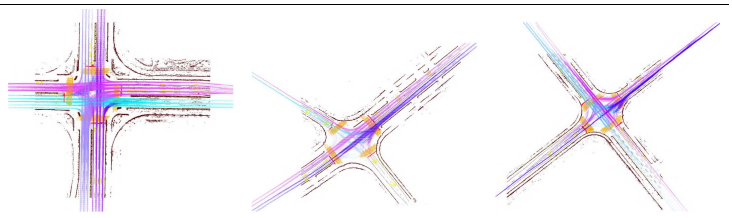
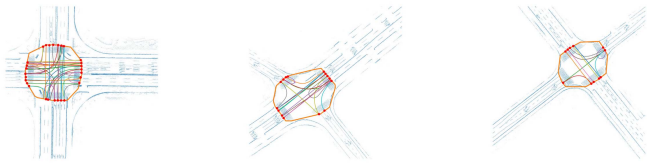
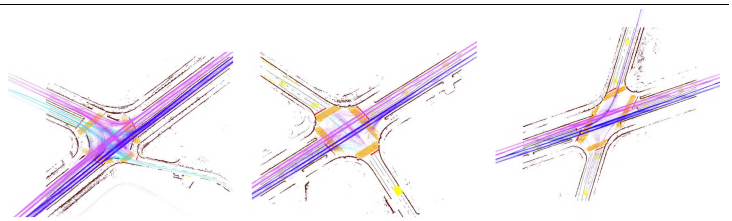
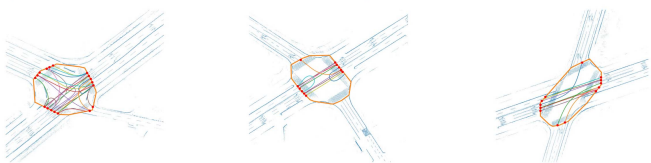
	5 trips	50 trips	100 trips
Semantic point cloud layer			
Traffic flow layer			
Intersection topology			

TABLE IV
TOPOLOGICAL STRUCTURE RESULTS FROM THE PROPOSED FRAMEWORK

index	0	1	2
semantic map and traffic flow			
topology			

index	3	4	5
semantic map and traffic flow			
topology			

trajectories). The smaller value denoted it fitted the human-driving trajectory better. The result was shown in Table II. It can be seen that the proposed reference line was closer to the traffic flows than HD Map, indicating that our results were more human-like.

Qualitatively, we can see some samples in Fig. 8, where reference lines generated by rigid rules in the HD map were far away from human driving trajectories. In Fig. 8(a), there was a U-shape turn, that human drivers couldn't follow because it was too narrow. As shown in Fig. 9, due to the limitation of the turning radian, the human drivers need to borrow more space to perform a U-turn, which leads to the U-turn reference lines in a bulb shape. Using rule-based U-turn reference lines in traditional HD maps is far from reality. The reference trajectory from the human driver themselves is more meaningful compared with the HD map. The same situation happened in the right turn and left turn in Fig. 8(b) and (c) receptively.

More topological structure results from the proposed framework are shown in Table IV.

C. Completeness Discussion

With the continuous increase of the crowdsourcing data, the semantic map becomes more complete and the traffic flow becomes denser. The topological structure evolves over time. Different from the HD Map, which is constant, our system has great growth potential.

As shown in Table III, we use a larger and more complex intersection to highlight the ability of our method. We demonstrated the evolution of topological results when the number of crowdsourcing trips increased from 5 to 100. At the very beginning, the data is too insufficient to recover a complete intersection structure. When the data grew to 50 trips, the semantic information and topological relationship of the intersection were mostly complete.

The amount of required data depends on the scale of the intersection. Taking a normal intersection with four roads, and each road with two forward and two reverse lanes as an example, it requires at least 48 different trajectories to recover the topological structure. Due to the noise and uncertainty issues, it usually needs 100 trips from ego-vehicle and other vehicles to fully recover the structure. The detailed time depends on the scale of vehicle fleets. The more vehicles, the shorter time will be.

V. Conclusion

In this letter, we propose a method that leverages crowdsourcing data to construct the topological structure within intersections, which solves the difficult problem to build the map of intersections and decreases the cost. The generated topological relationship is close to the HD Map annotated manually and can be constructed completely automatically without manual intervention. The generated reference path is more human-like than HD Maps.

Currently, the proposed system can not deal with construction changes. In the future, we will focus on identifying construction changes and updating the map automatically and timely. In addition, we will take different sizes of vehicles into consideration to recommend different reference lines.

References

[1] R. Liu, J. Wang, and B. Zhang, "High definition map for automated driving: Overview and analysis," *J. Navigation*, vol. 73, no. 2, pp. 324–341, 2020.

[2] C. Guo, K. Kidono, J. Meguro, Y. Kojima, M. Ogawa, and T. Naito, "A low-cost solution for automatic lane-level map generation using conventional in-car sensors," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2355–2366, Aug. 2016.

[3] C. Ye, J. Li, H. Jiang, H. Zhao, L. Ma, and M. Chapman, "Semi-automated generation of road transition lines using mobile laser scanning data," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 5, pp. 1877–1890, May 2020.

[4] L. Ma, Y. Li, J. Li, Z. Zhong, and M. A. Chapman, "Generation of horizontally curved driving lines in HD maps using mobile laser scanning point clouds," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 5, pp. 1572–1586, May 2019.

[5] Q. Li, Y. Wang, Y. Wang, and H. Zhao, "HDMapNet: An online HD map construction and evaluation framework," in *Proc. Int. Conf. Robot. Automat.*, 2022, pp. 4628–4634.

[6] Y. Liu, T. Yuan, Y. Wang, Y. Wang, and H. Zhao, "Vectormapnet: End-to-end vectorized HD map learning," in *Proc. Int. Conf. Mach. Learn.*, 2023, pp. 22352–22369.

[7] L. Bencheng et al., "MapTR: Structured modeling and learning for online vectorized HD map construction," 2022, *arXiv:2208.14437*.

[8] T. Ort, J. M. Walls, S. A. Parkison, I. Gilitschenski, and D. Rus, "Maplite 2.0: Online HD map inference using a prior SD map," *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 8355–8362, Jul. 2022.

[9] A. Joshi and M. R. James, "Joint probabilistic modeling and inference of intersection structure," in *Proc. IEEE 17th Int. Conf. Intell. Transp. Syst.*, 2014, pp. 1072–1078.

[10] Y. Zhou, Y. Takeda, M. Tomizuka, and W. Zhan, "Automatic construction of lane-level HD maps for urban scenes," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 6649–6656.

[11] L. Zhao et al., "Automatic calibration of road intersection topology using trajectories," in *Proc. IEEE 36th Int. Conf. Data Eng.*, 2020, pp. 1633–1644.

[12] T. Qin, Y. Zheng, T. Chen, Y. Chen, and Q. Su, "A light-weight semantic map for visual localization towards autonomous driving," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 11248–11254.

[13] K. Kim, S. Cho, and W. Chung, "HD map update for autonomous driving with crowdsourced data," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 1895–1901, 2021.

[14] P. Zhang, M. Zhang, and J. Liu, "Real-time HD map change detection for crowdsourcing update based on mid-to-high-end sensors," *Sensors*, vol. 21, no. 7, 2021, Art. no. 2477.

[15] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Interv.*, Springer, 2015, pp. 234–241.

[17] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian SegNet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," 2015, *arXiv:1511.02680*.

[18] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, *arXiv:1904.07850*.

[19] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 12689–12697.

[20] Y. Wang, V. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. M. Solomon, "DETR3D: 3D object detection from multi-view images via 3D-to-2 D queries," in *Proc. Conf. Robot Learn.*, 2021, pp. 180–191.

[21] Z. Qin, H. Yu, C. Wang, Y. Guo, Y. Peng, and K. Xu, "Geometric transformer for fast and robust point cloud registration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 11133–11142.