Background Filtering and Object Detection With a Stationary LiDAR Using a Layer-Based Method

YANJIE SONG [1,2], **HAN ZHANG** [3], **YUANQIANG LIU** [4], **JINZHANG LIU** [5], **HONGBO ZHANG** [1], **AND XIUGUANG SONG** [1]

[1] School of Qilu Transportation, Shandong University, Jinan 250061, China
[2] Suzhou Research Institute, Shandong University, Suzhou 215123, China
[3] Shandong Hi-speed Construction Management Group Company Ltd., Jinan 250001, China
[4] Yantai Water Building Conservancy Investigation and Design Academy, Yantai 264003, China
[5] Yantai Water Industry Group Company Ltd., Yantai 264003, China

Corresponding authors: Hongbo Zhang (zhanghongbo@sdu.edu.cn) and Xiuguang Song (songxiuguang@sdu.edu.cn)

ABSTRACT The connected vehicle environment is significant for the future road network. For constructing the connected vehicle environment, real-time data acquirement is always the prerequisite. Recently, using Light Detection and Ranging (LiDAR)-based roadside infrastructures are becoming a prevalent method of obtaining real-time traffic data. However, the collected raw data from LiDAR cannot usually be used directly. The steps of data processing, like background filtering and object detection, are necessary. The processed data can then be employed in different applications. This paper proposed a novel layer-based searching method that is established with the help of the point distribution features to distinguish moving objects from the point cloud. It aimed to address the unexpected influence of factors such as congested situations and package loss. The new approach was also evaluated compared with the state-of-the-art methods by applying field data. The results showed that the proposed method is more effective than other methods. This method may be applicable to other types of rotating LiDAR for improving the background filtering performance.

INDEX TERMS Background filtering, object detection, stationary LiDAR, layer-based method.

I. INTRODUCTION

Light Detection and Ranging (LiDAR) is capable of measuring distances to objects by emitting and detecting the lasers. Compared to the videos/cameras, LiDAR is more reliable under multiple light conditions [1]. Based on different installation forms, the LiDAR can generally be divided into two types: mobile LiDAR (airborne LiDAR and car-mounted LiDAR) and stationary LiDAR [2], [3]. The stationary LiDAR has recently attracted much attention from researchers and traffic engineers. One primary function of stationary LiDAR is to generate high-resolution micro traffic data (HRMTD) [3]. Specifically, in transportation engineering, the HRMTD means the trajectories of road-users with high frequency (acquisition frequency >10 Hz) [4], [5]. Prevalent traffic applications such as connected-vehicles, near-crash analysis, and dynamic traffic signal timing require the HRMTD as input [6], [7]. Compared to the mobile

LiDAR, the advantage of the stationary LiDAR is attributed to a fixed location. Since most background points under stationary LiDAR can be found in consecutive frames, it allows historical information to be used for background identification. With this feature, the computational load can greatly be decreased compared to the mobile LiDAR.

After data collection, data processing is the next step. Currently, an important application towards the collected data is to identify the objects. Though existing studies related to the object detection have attempted to use the roadside LiDAR data, those studies experienced several drawbacks under the following special conditions: 1) congested situations, 2) package loss, and 3) points drift [8], [9]. Congested situations indicate that the moving objects stop at some locations for a period [10]. The package loss means that some points are missing in the space caused by network overload or unstable connection [11]. Points drift means that the points representing the same stationary object have different coordinates at different frames [12]. Previous research often directly deployed density-based algorithms under these situations to

The associate editor coordinating the review of this manuscript and approving it for publication was Feng Xia [ID].

filter the background and detect objects; however, the existing density-based method may misidentify moving objects as background points (irrelevant points). For this limitation, it is necessary to develop a more advanced method to exclude background points and identify the objects from the roadside LiDAR data.

LiDAR can emit multiple laser beams at a time; each laser beam has a different horizontal angle. Different layers refer to different laser beams (ID). The idea is that it may be possible to improve the background filtering and object detection from data in different layers. Therefore, the authors proposed a novel layer-based method to remove the background and used rotating LiDAR for HRMTD collection [13]. The presented layer-based method can directly identify the background points by analyzing the distance distribution of points in the horizontal angles of each layer. Further, the rest points can be applied by a density-based spatial clustering method (DBSCAN) for object detection.

The remainder of this paper is organized as follows. Section 2 provides a review of related studies. Section 3 comprehensively introduces the roadside LiDAR and elaborates the points distribution feature generated by LiDAR scanning. Section 4 documents the details of the method for background filtering and point clustering. Section 5 evaluates the performance of the proposed method through case studies. Section 6 summarizes the findings of this paper and its future extension.

II. RELATED WORK

The current background filtering approaches for stationary LiDAR can be roughly divided into three groups: rule-based method, volumetric-based method, and point-based method.

The rule-based method is the most prevalent way to filter the background. By this method, the frame without any road-users in the space is usually selected as a target frame. Then the points in the target frame are defined as background points. For the other frames, points located within a pre-defined distance to the points in the target frame are considered as background points and are excluded from the raw LiDAR data. There have been several studies using the rule-based method to achieve background filtering. Zhang *et al.* [14] manually selected one frame without any moving object as a target frame. Points would directly be excluded from the space if their distances to other points in the target frame were less than the threshold. Lee and Coifman [15] developed a similar background subtraction method by giving a pre-defined range to the distance in each detected angle and each laser beam. By aggregating multiple target frames, the acceptable range of the distance regarding the background points in each angle could be obtained. The testing results showed that more than 90% of background points could successfully be excluded from the space. However, the major limitation of the rule-based methods is that it is difficult to obtain the target frame with non-moving objects under heavy traffic conditions [15].

The volumetric-based method converts the LiDAR data into the volumetric space with the rasterization procedure, which is also called the raster-based method [16]. The volumetric-based method usually divides the space into small cubes after aggregating multiple frames. The point density in each cube can then be calculated. Since the density of background points is higher than that of the moving object points after aggregation, the cube can be distinguished into background cube or non-background cube by giving a pre-defined density threshold. Wu *et al.* [18] firstly provided a fixed threshold of point density in the cube for background filtering based on their experience. A dynamic threshold was then developed by considering the point distribution and the mechanical properties of the LiDAR, namely, three-dimensional density-spatial filtering (3D-DSF). The major drawback of 3D-DSF is that the vehicles under congested situations are easily misidentified as background points. Lv *et al.* [16] used the change of point density in the cube at different frames to distinguish background points and non-background points. They considered any cube with a change of point density higher than 2 points in two adjacent frames as the non-background. Results show that more than 98% of background points can be excluded from their experiments. However, those methods did not consider the packet loss issue in their algorithms. As a result, the background points and non-background points might be misclassified under packet loss situations.

The point-based method identifies background points directly based on point distribution without any conversion. The computational load is lower than other methods since the points are stored in the raw LiDAR data and easily accessed. Qi *et al.* [19] used a changeable dimension strategy to store the feature of the points and applied the neural network (NN) to train a classifier to identify background points from the raw LiDAR data. However, the changeable-dimension method needs to be further verified since there is still a lack of systematic introduction to this method's practical application. Zhang *et al.* [20] developed a laser beam-based method for background filtering using roadside LiDAR. It was found that the existence of the moving objects can shorten the distance between the detected points and the LiDAR compared to the situation that no moving objects exist in the space. The previous studies found that the location of the points may drift from the actual location of the object due to the environmental factors or the mechanical properties of the LiDAR sensors [21]. The beam-based method developed by Zhang *et al.* [20] still has the point drift issue or the packet loss issue, and the performance of this method decreased dramatically under the packet loss issue. From the above-mentioned studies, we can find that the existing methods could not effectively subtract the background points yet under complex situations such as congested traffic, package loss, and points drift situations. A more effective method for background subtraction is needed.

Theoretically, data points in the same group should have similar properties or features, while data points in different groups should have different properties or features [22]. Based on the above characteristics, the points belonging

to one object can be grouped into a cluster. The current widely used methods for points clustering include K-means, mean-shift clustering, Gaussian Mixture Models (GMM), and DBSCAN [23], [24]. K-means method usually selects a number of classes and initializes their respective center points. Each data point is classified by computing the distance between that point and each group center. The point can be classified into the point group of the nearest center point. According to the initial classification result, the group center is recalculated by the mean of all the vectors in the group [25]. By repeating the above-mentioned steps, the group centers can eventually become stable. The biggest disadvantage of K-means is that the number of groups needs to be known in advance. The accuracy of the clustering result is significantly influenced by the initially selected points. Mean-shift clustering that attempts to find dense areas of data points is a sliding-window-based algorithm. It is a centroid-based algorithm to locate the center points of each group. It keeps updating the center point to be the mean of the points within the sliding-window. The mean-shift clustering algorithm starts with a circular sliding window centered at a randomly selected point with a radius r as the kernel [26]. Then the algorithm updates the kernel iteratively to a higher density region using a hill-climbing method until convergence. The sliding window proceeds continuously according to the mean until no more points to be updated inside the kernel. In contrast to K-means clustering, there is no need to select the number of clusters as the mean-shift method can automatically discover it. However, the shortcoming of the mean-shift clustering is the low computation speed caused by the sliding window. In GMMs, the data points are assumed to be Gaussian distributed. Two indexes are used to describe the shape of the cluster: the mean and the standard deviation [27]. An optimization algorithm called Expectation-Maximization (EM) is applied to determine the parameters of the Gaussian for each cluster. The Gaussian distribution parameters for each cluster are randomly initialized. The probability of each data point belongs to a cluster can be computed. The closer a point attaches to the Gaussian's center, the more likely it belongs to that cluster. Based on these probabilities, the probabilities of data points within the clusters can be maximized. These parameters using a weighted sum of the data point positions, can then be computed. GMMs are also flexible in terms of cluster covariance. Due to the use of the standard deviation parameter, the clusters can be formed in an ellipse shape, rather than being restricted to circles. The major limitation of the GMM algorithm is the heavy computational load; it is ineffective if the dimensionality of the problem exceeds a threshold. Another disadvantage of the GMM algorithm is that the user needs to set the number of mixture models to satisfy the training dataset. However, in many cases, users do not know how many mixture models should be used. Users may have to generate a number of different mixture models to find the most suitable model set that works for their classification problem. DBSCAN employs two important parameters: epsilon (Eps) and minimum points (MinPts) for clustering

points [28]. Eps represents the radius of the neighborhood, and MinPts is the minimum number of neighbors within Eps. For a point, if the number of its neighbor is greater than or equal to MinPts, this point is marked as a core point. If the number of its neighbor is less than MinPts, but the point belongs to the neighbor of other core points, this point is marked as a border point. If the point does not belong to the core point and border point, this point is called a noise point [29]. Since the LiDAR can provide a higher density for the moving objects after background filtering and the number of moving objects is unknown, the DBSCAN can be an ideal method for LiDAR point density. Several studies have been conducted using different approaches of DBSCAN for points clustering. Cui *et al.* [17] firstly applied the traditional DBSCAN for roadside LiDAR point clustering. It was found that a fixed Eps and MinPts usually could not successfully cluster the points that are far away from the sensor. Chen *et al.* [28] developed a revised DBSCAN for deer identification using the roadside LiDAR. The error rate was less than 8% in their practice. Nevertheless, the time delay for computation remained a problem in collecting real-time HRTMD [29]. Zhao *et al.* [30] developed a revised DBSCAN algorithm by deploying adaptive parameters considering the point distribution in the space and the mechanical features of the LiDAR sensor. It was found that the revised DBSCAN can greatly improve the clustering accuracy compared to the traditional DBSCAN. It should be noted that the DBSCAN suffers the major limitation of the high computational load. Efficiency improvement of clustering is still needed to reduce the time delay further.

III. DATA

This paper employed RS-LiDAR-32 for data collection and algorithm validation. The RS-LiDAR-32 sensor uses 32 infrared (IR) lasers along with IR detectors to measure distances to objects. The device is usually installed on a compact and weather-resistant cabinet. The LiDAR rotates rapidly within its fixed housing to scan the surrounding environment and provide a set of real-time 3D point data. The whole sensing system includes desktop/laptop computers, GPS Antenna (optional), interface boxes, LiDAR sensors, and a DC power supply. The time-of-flight (ToF) methodology is applied in the RS-LiDAR-32. When each IR laser emits a laser pulse, its time-of-shooting and direction are registered. The laser pulse may hit an object, which reflects some of the energy back to the LiDAR. A portion of that energy is received by the paired IR detector, registered as the time-of-acquisition and received power. The sensor's rotation revolutions can be 300 Per Minute (RPM), 600RPM, or 1200 RPM. A two-byte azimuth value (α) appears after the flag bytes at the beginning of each data block. The azimuth is indicated by an unsigned integer. It represents an angle in centesimal measurement. For instance, a raw value of 27742 should be interpreted as $277.42°$. Only one azimuth value is reported per data block. The major features of RS-LiDAR-32 are summarized in Table 1.

TABLE 1. Major features of RS-LiDAR-32.

Method of Distance Calculation	ToF
Number of Channels	32 (non-linear distribution)
Distance Sensing Range	40 cm to 200 m
Distance Resolution	+/- 3cm
Vertical Field of View	-25°~+15°
Horizontal Field of View	360°
Vertical Angle Resolution	0.33°
Rotation Speed	300/600/1200 rpm
Laser Wavelength	905 nm
Output Point Size	~600k points/second
Power	13.5 w
Weight	1.17 kg
Size	114mm (diam)*108.73mm (height)
Working Temperature Range	-30°C~60°C
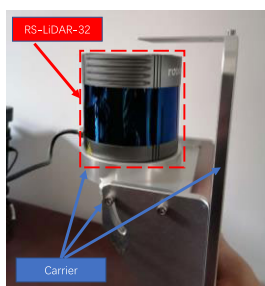
FIGURE 1. RS-LiDAR-32.
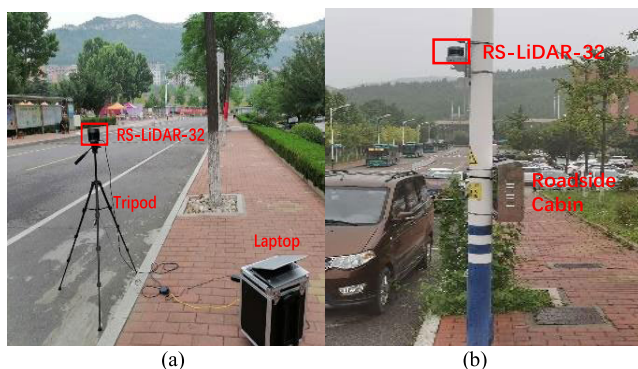
FIGURE 2. Two types of roadside LiDAR installation. (a) LiDAR portable installation; (b) LiDAR fixed installation.

The LiDAR can temporarily be mounted on a tripod for data collection or on a permanently fixed location such as a traffic signal pole for long-term data collection. As for roadside LiDAR implementation, the recommended height is 7-9ft above the ground, considering the horizontal field of view (FOV) and the occlusion issue. Figure 1 shows an example of the RS-LiDAR-32 and its customized carrier.

Figure 2 shows two different data collection methods.

To better express the point distribution feature of the roadside LiDAR, two adjacent frames (Frame 1 and Frame 2) without moving objects were selected, and their distance differences of the scanned points were compared at each
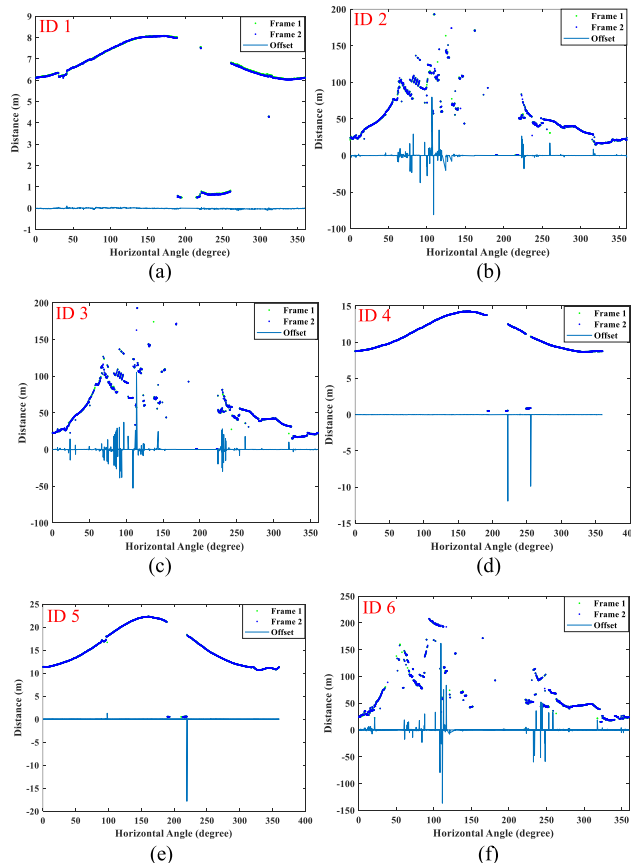

FIGURE 3. Distance Offset Distribution along the horizontal angle for each laser beam: (a) ID 1(Vertical Angle:-10.281), (b) ID 2(Vertical Angle:-6.424), (c) ID 3(Vertical Angle:2.333), (d) ID 4(Vertical Angle:3.333), (e) ID 5(Vertical Angle:4.667), (f) ID 6(Vertical Angle:7.000), (g) ID 7(Vertical Angle:10.333), (h) ID 8(Vertical Angle:15.000), (i) ID 9(Vertical Angle:0.333), (j) ID 10(Vertical Angle:0.000), (k) ID 11(Vertical Angle:-0.333), (l) ID 12(Vertical Angle:-0.667), (m) ID 13(Vertical Angle:1.667), (n) ID 14(Vertical Angle:1.333), (o) ID 15(Vertical Angle:1.000), (p) ID 16(Vertical Angle:0.667), (q) ID 17(Vertical Angle:-25.000), (r) ID 18(Vertical Angle:-14.638), (s) ID 19(Vertical Angle:-7.910), (t) ID 20(Vertical Angle:-5.407), (u) ID 21(Vertical Angle:-3.667), (v) ID 22(Vertical Angle:-4.000), (w) ID 23(Vertical Angle:-4.333), (x) ID 24(Vertical Angle:-4.667), (y) ID 25(Vertical Angle:-2.333), (z) ID 26(Vertical Angle:-2.667), (aa) ID 27(Vertical Angle:-3.000), (bb) ID 28(Vertical Angle:-3.333), (cc) ID 29(Vertical Angle:-1.000), (dd) ID 30(Vertical Angle:-1.333), (ee) ID 31(Vertical Angle:-1.667), (ff) ID 32(Vertical Angle:-2.000).

same horizontal angle. Figure 3 shows the distance offset distribution in the 360-degree horizontal FOV with different laser ID.

It is shown that for most background points, the offset for one point between two frames was insignificant. But for some laser beams such as ID 17, 20, and 24, there were a lot of randomly located points with large distance offset. By further checking the LiDAR video, it was found that those random points were purely caused by the noise or the dynamic background (such as swinging branches of the trees). To identify the influence of moving objects in the space on the points distribution, a frame without moving objects in the laser beam and another frame with moving objects in the same laser beam were selected for comparison. The point distribution is shown
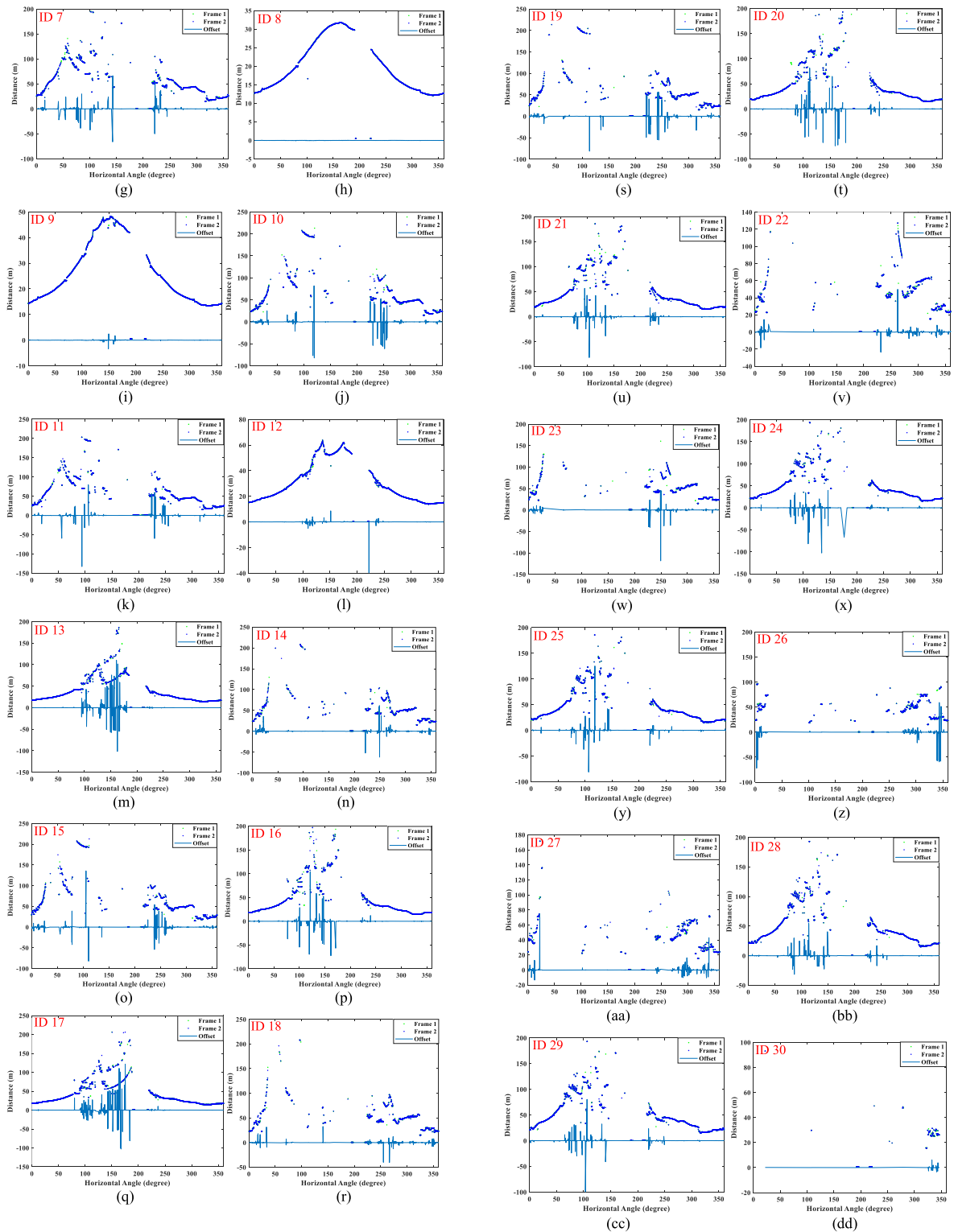
FIGURE 3. *(Continued.)* **Distance Offset Distribution along the horizontal angle for each laser beam: (a) ID 1(Vertical Angle:-10.281), (b) ID 2(Vertical Angle:-6.424), (c) ID 3(Vertical Angle:2.333), (d) ID 4(Vertical Angle:3.333), (e) ID 5(Vertical Angle:4.667), (f) ID 6(Vertical Angle:7.000), (g) ID 7(Vertical Angle:10.333), (h) ID 8(Vertical Angle:15.000), (i) ID 9(Vertical Angle:0.333), (j) ID 10(Vertical Angle:0.000), (k) ID 11(Vertical Angle:-0.333), (l) ID 12(Vertical Angle:-0.667), (m) ID 13(Vertical Angle:1.667), (n) ID 14(Vertical Angle:1.333), (o) ID 15(Vertical Angle:1.000), (p) ID 16(Vertical Angle:0.667), (q) ID 17(Vertical Angle:-25.000), (r) ID 18(Vertical Angle:-14.638), (s) ID 19(Vertical Angle:-7.910), (t) ID 20(Vertical Angle:-5.407), (u) ID 21(Vertical Angle:-3.667), (v) ID 22(Vertical Angle:-4.000), (w) ID 23(Vertical Angle:-4.333), (x) ID 24(Vertical Angle:-4.667), (y) ID 25(Vertical Angle:-2.333), (z) ID 26(Vertical Angle:-2.667), (aa) ID 27(Vertical Angle:-3.000), (bb) ID 28(Vertical Angle:-3.333), (cc) ID 29(Vertical Angle:-1.000), (dd) ID 30(Vertical Angle:-1.333), (ee) ID 31(Vertical Angle:-1.667), (ff) ID 32(Vertical Angle:-2.000).**

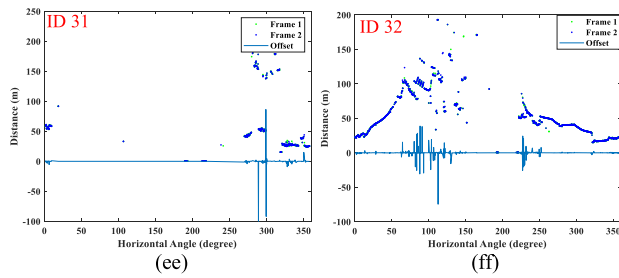**FIGURE 3.** *(Continued.)* Distance Offset Distribution along the horizontal angle for each laser beam: (a) ID 1(Vertical Angle:-10.281), (b) ID 2(Vertical Angle:-6.424), (c) ID 3(Vertical Angle:2.333), (d) ID 4(Vertical Angle:3.333), (e) ID 5(Vertical Angle:4.667), (f) ID 6(Vertical Angle:7.000), (g) ID 7(Vertical Angle:10.333), (h) ID 8(Vertical Angle:15.000), (i) ID 9(Vertical Angle:0.333), (j) ID 10(Vertical Angle:0.000), (k) ID 11(Vertical Angle:-0.333), (l) ID 12(Vertical Angle:-0.667), (m) ID 13(Vertical Angle:1.667), (n) ID 14(Vertical Angle:1.333), (o) ID 15(Vertical Angle:1.000), (p) ID 16(Vertical Angle:0.667), (q) ID 17(Vertical Angle:-25.000), (r) ID 18(Vertical Angle:-14.638), (s) ID 19(Vertical Angle:-7.910), (t) ID 20(Vertical Angle:-5.407), (u) ID 21(Vertical Angle:-3.667), (v) ID 22(Vertical Angle:-4.000), (w) ID 23(Vertical Angle:-4.333), (x) ID 24(Vertical Angle:-4.667), (y) ID 25(Vertical Angle:-2.333), (z) ID 26(Vertical Angle:-2.667), (aa) ID 27(Vertical Angle:-3.000), (bb) ID 28(Vertical Angle:-3.333), (cc) ID 29(Vertical Angle:-1.000), (dd) ID 30(Vertical Angle:-1.333), (ee) ID 31(Vertical Angle:-1.667), (ff) ID 32(Vertical Angle:-2.000).

in Figure 4. From the figure, it can be seen that the detected distances to moving objects were less than the distances to the background (without moving objects) along the direction of the laser beam.

A further investigation examined the distance offset patterns between the moving objects and the background objects in the space. As shown in Figure 5 (b) and (c), when moving objects were scanned by the laser beam, the distance offset was continuous among some adjacent angles. While for fixed or dynamic background, the distance offset patterns appeared as flatten lines or random fluctuation lines. Therefore, the point distribution patterns were different between moving objects and background objects in the space. Generally, the point distribution feature of moving objects is "curve sinking" (because the moving objects are close to the LiDAR) and "continuous" compared with background objects. This paper used this feature to distinguish moving objects and background points.

IV. METHODOLOGY

This section introduces a novel method for background filtering and points clustering. This method can exclude background points and identify the object simultaneously without needing to distinguish background filtering and point clustering strictly.

As mentioned before, the point offset distribution is a major feature used to distinguish background and non-background. The normal distance offset caused by vibration is illustrated as Υ. Υ is determined by the distance resolution of the LiDAR. The number of continuous points (NCP) with offsets higher than Υ is another factor to be considered. A low value of NCP may result in misidentifying the
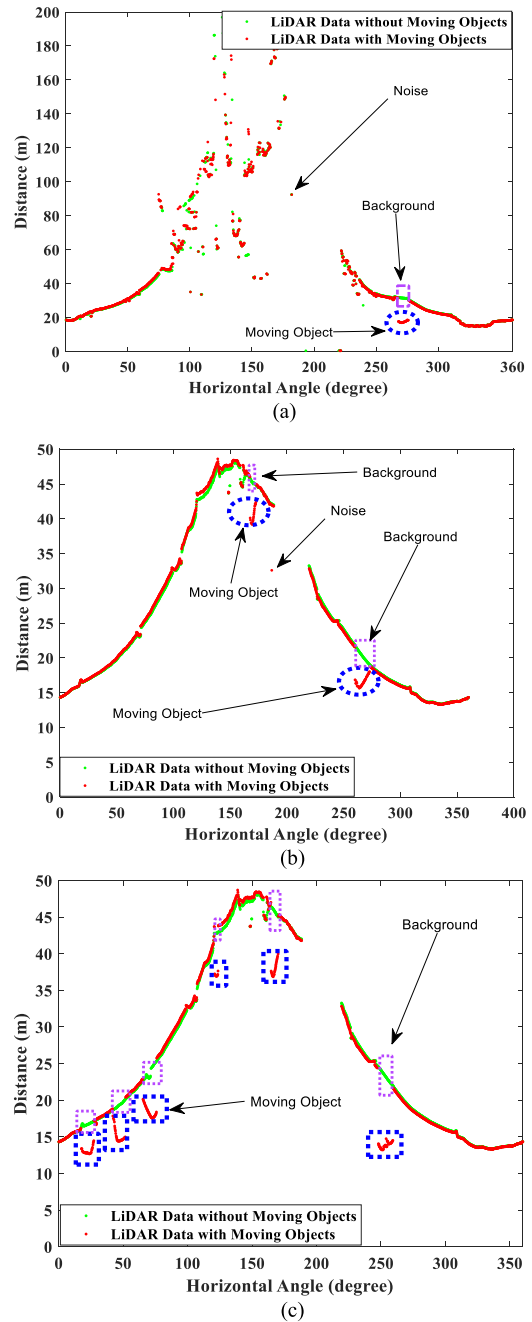

FIGURE 4. Distance distribution along the horizontal angle with moving objects in the space: (a) One moving object in the space, (b) Two moving objects in the space, (c) Six moving objects in the space.

dynamic background points as moving object points under point drifting conditions. A high value of NCP may result in misidentifying some small size moving objects (such as pedestrians) as background points. For RS-LiDAR-32, Υ is 3 cm (representing an offset with ±3cm), meaning that if the offset of one point at two adjacent frames is less than 6 cm, the point will be identified as a dynamic background point. As for NCP, 3 was selected as a threshold based on Lv *et al.*'s analysis [16]. Therefore, a moving object should meet the following criteria—Equation (1) indicating
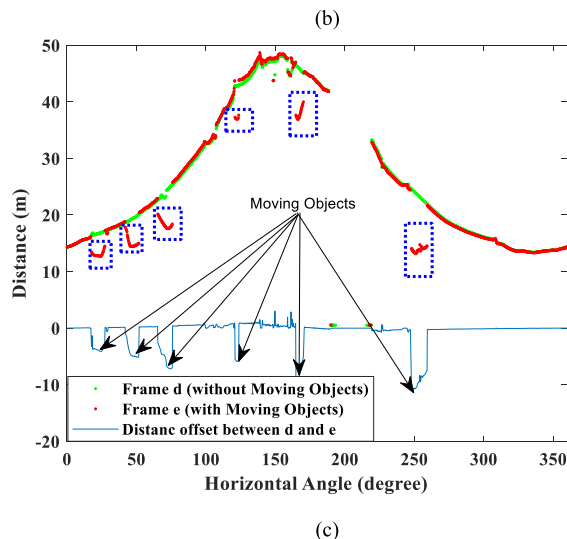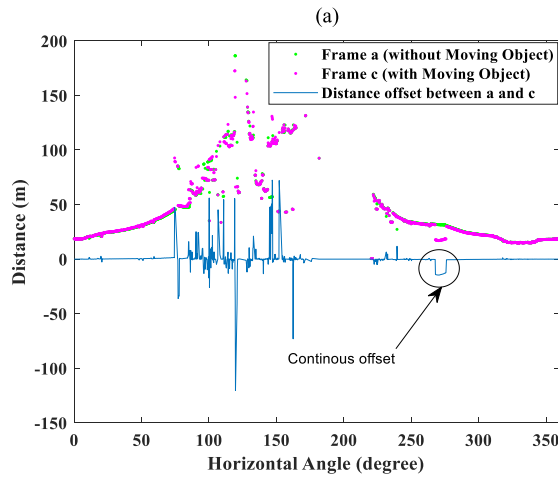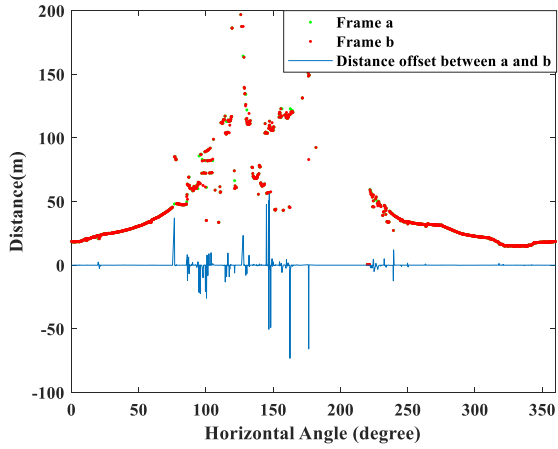
(a)

(b)

(c)

FIGURE 5. Distance offset distribution along the horizontal angle: (a) Distance offset between two frames without moving objects, (b) Distance offset between one frame with one moving object and one frame without moving objects, (c) Distance offset between one frame with six moving objects and one frame without moving objects.
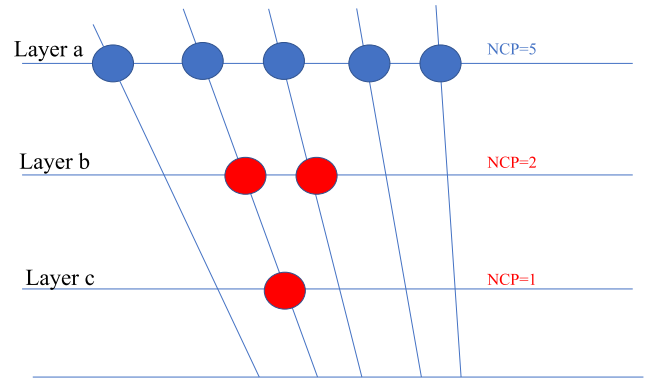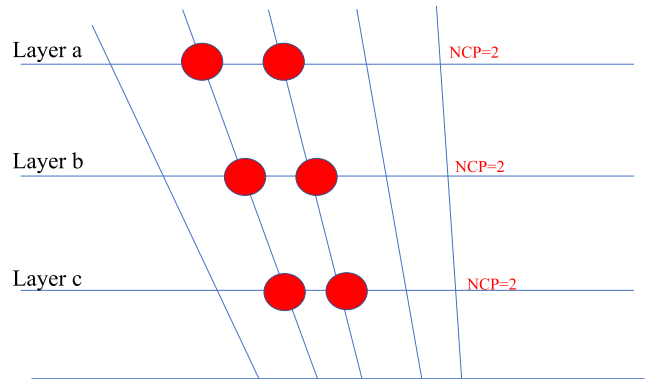

FIGURE 6. Special case 1.

FIGURE 7. Special case 2.

critical information. For example, assuming 5 continuous points exist in one layer from an angle Θ to $\Theta + 0.08$ degree, all points will be identified as moving objects (because the NCP = 5 > 3). At its adjacent layer, only 2 continuous points (NCP = 2 < 3) belong to the same moving object. The reduced NCP may be as a result of occlusion, package loss, or other issues. At another layer, only one point (NCP = 1 < 3) belongs to the same moving object, as shown in Figure 6. If the method in (1) is deployed, those points at layer b and c will be identified as background points.

Another special case is that the NCP value of each layer (a, b, c) is 2, as shown in Figure 7. This NCP distribution pattern can be regarded as a pedestrian with a larger height than width. By the method in Equation (1), the moving object points (pedestrian) are misidentified as background since NCP = 2 < 3 at each layer.

It should be noticed that the layer a, b, and c represent three adjacent laser beams, the dots in each layer shown in Figure 6 and 7 represent consecutive points with an offset greater than Υ. As can be seen from Figures 6 and 7, the number of points at the same angle in adjacent layers (NPAL) should also be considered as a factor for background filtering. Due to occlusion or point drift, the points representing the same object in adjacent layers may not be captured at the same angle. Therefore, the critical problem is how to determine the neighbors of one point. Searching the neighbors in one layer is infeasible. To solve this problem, a layer-based
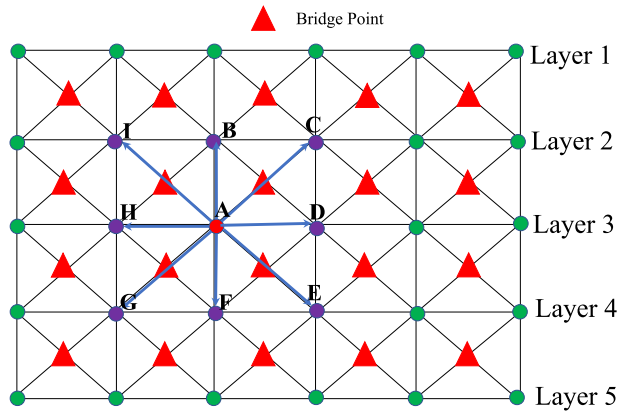
offsets of a consecutive 3 points larger than 6cm.

$$\{\Upsilon_\theta >= 6 \ \& \ \Upsilon_{\theta+0.02} >= 6 \ \& \ \Upsilon_{\theta+0.04} >= 6\} \quad (1)$$

where θ is a horizontal angle at one layer, and Υ is the offset. However, only checking NCP at one layer may miss some

FIGURE 8. NPAL searching procedure.

searching method was invented. The details of the method can be illustrated in Figure 8. The triangles in Figure 8 are dummy points serving as the bridges. If two points can be connected through the bridge, they can be considered as the points in NPAL even they are not at the same angle. Therefore, for one point in one layer, the point can find eight neighbors at most.

The resolution of LiDAR decreases with the increasing distance between the object and the sensor. The searching strategy in Figure 8 can further be illustrated in (2).

$$NCP = n_A^i > 5 \tag{2}$$

where n_A^i is the number of points in point-A's neighbors at layer i.

The DBSCAN is applied for point clustering as follows. The specific application about the DBSCAN method refers to Zhao *et al.* [30]. For one point-A, if it meets (2), then A and its neighbors will be considered as a core point. All core points will be included in one group. With this strategy, all points in Figures 6 and 7 will be grouped correctly. Since all the searching points should meet the threshold that $\Upsilon >= 6$, the number of points used from raw LiDAR data as input of DBSCAN was greatly reduced, which can significantly increase the computation efficiency.

V. EVALUATION

Data collected at three sites under different scenarios were used as examples to illustrate the performance of the proposed layer-based algorithm. Those sites include a freeway segment, an intersection, and a rural arterial segment. The major features of the three sites are shown in Table 2.

Figure 9 shows the results of a frame before and after data processing on a freeway segment (Site 1). There were ten vehicles in the selected frame. After applying the layer-based method, all ten vehicles were kept in the space, and background points were successfully excluded. The point clouds were plotted in the Cartesian coordinates.
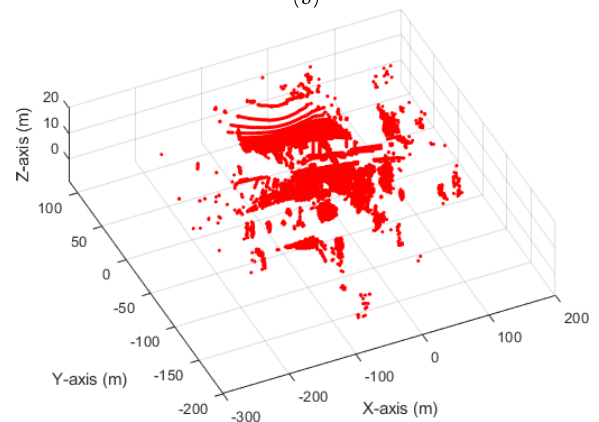
Figure 10 shows the results of a frame before and after data processing at a signalized intersection (site 2). There were six
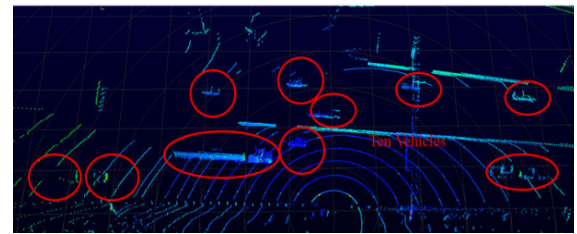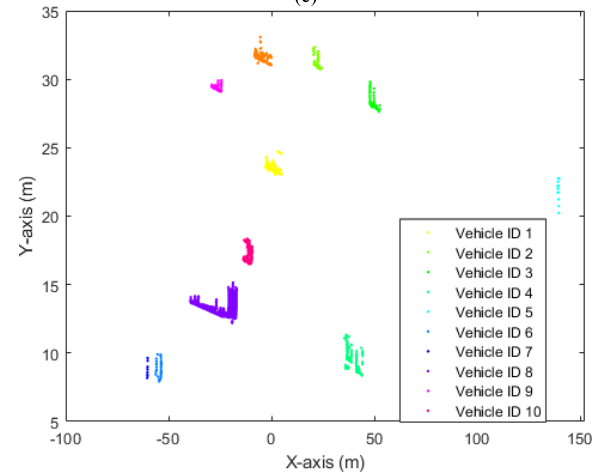

(a)
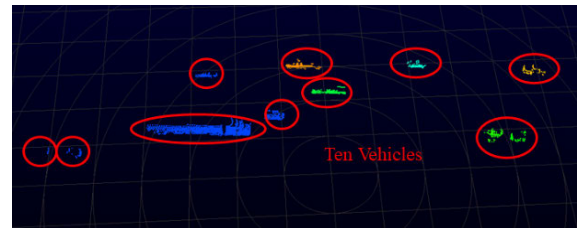
(b)

(c)

(d)

FIGURE 9. Before-and-after laser-based processing on a freeway segment: (a) Raw data (visualized in RSview), (b) Data after applying the layer-based Method, (c) Raw point cloud, (d) Data point cloud after applying the layer-based method.

vehicles at different directions in the selected frame, representing a congested situation. After applying the layer-based method, all six vehicles were kept in the space, and all background points were successfully excluded.

TABLE 2. Major features of selected sites.

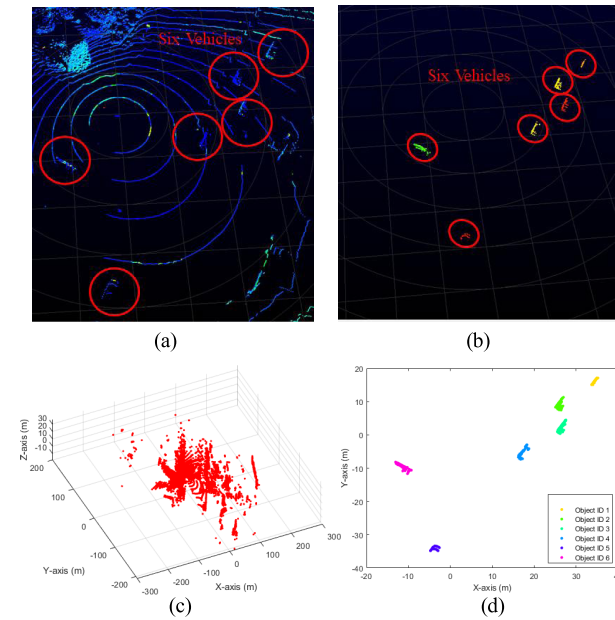| Site ID	Type	Speed limit	AADT	Number of through lanes
Site 1	Freeway	120 km/h	13000	2
Site 2	Urban Intersection	70 km/h	9000(major road)	3(major road)
Site 3	Rural Arterial	70 km/h	5000	1

FIGURE 10. Before-and-after laser-based processing at an intersection. (a) Raw data (visualized in RSview), (b) Data after applying the layer-based Method, (c) Raw point cloud, (d) Data point cloud after applying the layer-based method.

Figure 11 shows the results of a frame before and after data processing at a rural arterial segment (site 3). Seven vehicles were detected in the selected frame. After applying the layer-based method, all vehicles were kept in the space, and background points were successfully excluded.

To further evaluate the performance of the developed method, the layer-based algorithm was compared to the state-of-the-art methods. The density-based method (3D-DSF proposed by Wu *et al.*) [18] and the raster-based method (RA developed by Lv *et al.*) [16] were used for comparison. The results are shown in Table 3. It was found that the proposed layer-based method (LB) performs better than the other two methods at the three sites. The computational time of the algorithm is important for traffic behavior and safety analysis [31]. Therefore, the filtering consumed time was also compared between this method and the 3D-DSF method. After further statistical analysis, the filtering speed of this method is only about 3% slower than the DSF method and can meet the actual real-time requirement. It should be noted that the 3D-DSF needs more pre-processing time compared with this LB method.
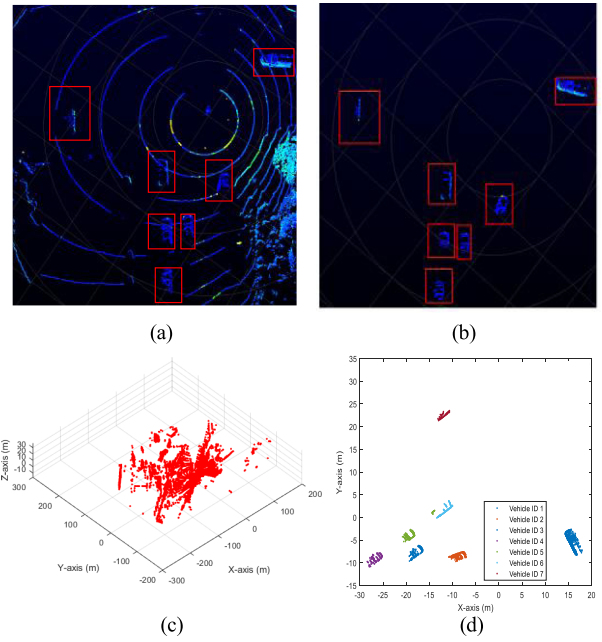

FIGURE 11. Before-and-after laser-based processing at an intersection: (a) Raw Data (Visualized in RSview), (b) Data after Applying the Layer-based Method, (c) Raw Point Cloud, (d) Data Point Cloud after applying the layer-based method.

TABLE 3. Performance evaluation.

Site	Method	Object points* (Before processing)	Object points (After data processing)	Object points excluded percentage
	LB		**3352**	**2.07%**
Site 1	3D-DSF	3423	2109	38.39%
	RA		2875	16.01%
	LB		**8992**	**2.13%**
Site 2	3D-DSF	9188	8816	4.05%
	RA		8971	2.36%
	LB		**6989**	**3.89%**
Site 3	3D-DSF	7272	6605	9.17%
	RA		6804	6.44%

VI. CONCLUSION

This paper developed a new method to exclude background points and identify the objects from the roadside LiDAR data. The proposed method used the point distribution pattern at different layers as a novel feature for moving object identification. Based on the feature, a layer-based searching method is then proposed to address the unexpected influence of factors such as congested situations and package loss. The evaluation showed that the proposed method could provide higher accuracy compared to the state-of-the-art methods. The presented method can keep more than 96% of moving object points and effectively excluded all the background points from the space. The computational time of the proposed method can meet the actual application needs (only about 3% slower than 3D-DSF). Furthermore, it can also be extended applied to other types of rotating LiDAR after properly calibrating the parameters.

More data are required to further investigate the performance of the developed method. The computation time of the proposed method needs to be recorded and quantized. The next step of the research is to analyze the influence of different weather conditions on the effectiveness of the proposed method.

REFERENCES

[1] Z. Zhang, J. Zheng, H. Xu, and X. Wang, "Vehicle detection and tracking in complex traffic circumstances with roadside LiDAR," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2673, no. 9, pp. 62–71, Sep. 2019, doi: 10.1177/0361198119844457.

[2] J. Wu, H. Xu, Y. Tian, Y. Zhang, J. Zhao, and B. Lv, "An automatic lane identification method for the roadside light detection and ranging sensor," *J. Intell. Transp. Syst.*, vol. 24, no. 5, pp. 467–479, Sep. 2020, doi: 10.1080/15472450.2020.1718500.

[3] Y. Cui, J. Wu, H. Xu, and A. Wang, "Lane change identification and prediction with roadside LiDAR data," *Opt. Laser Technol.*, vol. 123, Mar. 2020, Art. no. 105934, doi: 10.1016/j.optlastec.2019.105934.

[4] J. Wu, Y. Zhang, and H. Xu, "A novel skateboarder-related near-crash identification method with roadside LiDAR data," *Accident Anal. Prevention*, vol. 137, Mar. 2020, Art. no. 105438, doi: 10.1016/j.aap.2020.105438.

[5] J. Wu, H. Xu, Y. Zhang, and R. Sun, "An improved vehicle-pedestrian near-crash identification method with a roadside LiDAR sensor," *J. Saf. Res.*, vol. 73, pp. 211–224, Jun. 2020, doi: 10.1016/j.jsr.2020.03.006.

[6] J. Wu, H. Xu, Y. Zhang, Y. Tian, and X. Song, "Real-time queue length detection with roadside LiDAR data," *Sensors*, vol. 20, no. 8, p. 2342, Apr. 2020, doi: 10.3390/s20082342.

[7] Y. Ma, Y. Zheng, J. Cheng, and S. Easa, "Real-time visualization method for estimating 3D highway sight distance using LiDAR data," *J. Transp. Eng., A, Syst.*, vol. 145, no. 4, Apr. 2019, Art. no. 04019006, doi: 10.1061/JTEPBS.0000228.

[8] K. Dai, Y. Wang, Q. Ji, H. Du, and C. Yin, "Multiple vehicle tracking based on labeled multiple Bernoulli filter using pre-clustered laser range finder data," *IEEE Trans. Veh. Technol.*, vol. 68, no. 11, pp. 10382–10393, Nov. 2019, doi: 10.1109/TVT.2019.2938253.

[9] J. Wu, H. Xu, B. Lv, R. Yue, and Y. Li, "Automatic ground points identification method for roadside LiDAR data," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2673, no. 6, pp. 140–152, Jun. 2019, doi: 10.1177/0361198119843869.

[10] J. Wu, Y. Zhang, Y. Tian, R. Yue, and H. Zhang, "Automatic vehicle tracking with LiDAR-enhanced roadside infrastructure," *J. Test. Eval.*, vol. 49, no. 1, Jan. 2021, Art. no. 20190859, doi: 10.1520/JTE20190859.

[11] J. Wu, H. Xu, and J. Zhao, "Automatic lane identification using the roadside LiDAR sensors," *IEEE Intell. Transp. Syst. Mag.*, vol. 12, no. 1, pp. 25–34, Oct. 2020, doi: 10.1109/MITS.2018.2876559.

[12] Z. Ren, L. Wang, and L. Bi, "Robust GICP-based 3D LiDAR SLAM for underground mining environment," *Sensors*, vol. 19, no. 13, p. 2915, Jul. 2019, doi: 10.3390/s19132915.

[13] J. Wu, H. Xu, Y. Tian, R. Pi, and R. Yue, "Vehicle detection under adverse weather from roadside LiDAR data," *Sensors*, vol. 20, no. 12, p. 3433, Jun. 2020, doi: 10.3390/s20123433.

[14] Z. Y. J. Y. X. Zhang Zheng Wang and X. L. Fan, "Background filtering and vehicle detection with roadside lidar based on point association," in *Proc. 37th Chin. Control Conf. (CCC)*, Jul. 2018, pp. 7938–7943, doi: 10.23919/ChiCC.2018.8484040.

[15] H. Lee and B. Coifman, "Side-fire lidar-based vehicle classification," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2308, no. 1, pp. 173–183, Jan. 2012, doi: 10.3141/2308-19.

[16] B. Lv, H. Xu, J. Wu, Y. Tian, and C. Yuan, "Raster-based background filtering for roadside LiDAR data," *IEEE Access*, vol. 7, pp. 76779–76788, 2019, doi: 10.1109/ACCESS.2019.2919624.

[17] Y. Cui, H. Xu, J. Wu, Y. Sun, and J. Zhao, "Automatic vehicle tracking with roadside LiDAR data for the connected-vehicles system," *IEEE Intell. Syst.*, vol. 34, no. 3, pp. 44–51, May 2019, doi: 10.1109/MIS.2019.2918115.

[18] J. Wu, H. Xu, Y. Sun, J. Zheng, and R. Yue, "Automatic background filtering method for roadside LiDAR data," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2672, no. 45, pp. 106–114, Dec. 2018, doi: 10.1177/0361198118775841.

[19] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 77–85.

[20] Z. Zhang, J. Zheng, H. Xu, X. Wang, X. Fan, and R. Chen, "Automatic background construction and object detection based on roadside LiDAR," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 10, pp. 4086–4097, Oct. 2020, doi: 10.1109/TITS.2019.2936498.

[21] Q. Wang, J. Zheng, H. Xu, B. Xu, and R. Chen, "Roadside magnetic sensor system for vehicle detection in urban environments," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1365–1374, May 2018, doi: 10.1109/TITS.2017.2723908.

[22] Y. Zhang, H. Xu, and J. Wu, "An automatic background filtering method for detection of road users in heavy traffics using roadside 3-D LiDAR sensors with noises," *IEEE Sensors J.*, vol. 20, no. 12, pp. 6596–6604, Jun. 2020, doi: 10.1109/JSEN.2020.2976663.

[23] G. Menardi, "Nonparametric clustering for image segmentation," *Stat. Anal. Data Mining, ASA Data Sci. J.*, vol. 13, no. 1, pp. 83–97, Feb. 2020, doi: 10.1002/sam.11444.

[24] J. Sander, M. Ester, H.-P. Kriegel, and X. Xu, "Density-based clustering in spatial databases: The algorithm GDBSCAN and its applications," *Data Mining Knowl. Discovery*, vol. 2, no. 2, pp. 169–194, 1998, doi: 10.1023/A:1009745219419.

[25] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *J. Royal Stat. Soc., Ser. C, Appl. Statist.*, vol. 28, no. 1, pp. 100–108, 1979, doi: 10.2307/2346830.

[26] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002, doi: 10.1109/34.1000236.

[27] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 1, pp. 72–83, Jan. 1995, doi: 10.1109/89.365379.

[28] J. Chen, H. Xu, J. Wu, R. Yue, C. Yuan, and L. Wang, "Deer crossing road detection with roadside LiDAR sensor," *IEEE Access*, vol. 7, pp. 65944–65954, 2019, doi: 10.1109/ACCESS.2019.2916718.

[29] T. N. Tran, K. Drab, and M. Daszykowski, "Revised DBSCAN algorithm to cluster data with dense adjacent clusters," *Chemometric Intell. Lab. Syst.*, vol. 120, pp. 92–96, Jan. 2013, doi: 10.1016/j.chemolab.2012.11.006.

[30] J. Zhao, H. Xu, H. Liu, J. Wu, Y. Zheng, and D. Wu, "Detection and tracking of pedestrians and vehicles using roadside LiDAR sensors," *Transp. Res. C, Emerg. Technol.*, vol. 100, pp. 68–87, Mar. 2019, doi: 10.1016/j.trc.2019.01.007.

[31] C. Lin, H. Liu, D. Wu, and B. Gong, "Background point filtering of low-channel infrastructure-based LiDAR data using a slice-based projection filtering algorithm," *Sensors*, vol. 20, no. 11, p. 3054, May 2020, doi: 10.3390/s20113054.

YANJIE SONG received the B.S. degree in water resources and hydropower engineering from the School of Water Resources and Hydropower, Sichuan University, in 2016. He is currently pursuing the M.S. degree with the Department of Highway and Railway Engineering, School of Qilu Transportation, Shandong University. His research interests include LiDAR data analysis, traffic safety, traffic signal control, and traffic operations.

HAN ZHANG is currently a Senior Researcher with Shandong Hi-speed Construction Management Group Company Ltd. His research interests include traffic operations and intelligent transportation systems.

YUANQIANG LIU is currently a Senior Researcher with the Yantai Water Building Conservancy Investigation and Design Academy. His research interests include road structure design and traffic safety.

JINZHANG LIU is currently the Legal Representative of Yantai Water Industry Group Company Ltd. His research interests include traffic operations and intelligent transportation.

HONGBO ZHANG received the B.S. degree in traffic civil engineering and the M.S. degree in hydraulic structure engineering from the School of Civil Engineering, Shandong University, in 2000 and 2002, respectively, and the Ph.D. degree in geotechnical engineering from the School of Civil Engineering, Tongji University, in 2006. He was a Visiting Scholar with the University of Kansas, in 2015. He is currently working as an Associate Professor with the School of Qilu Transportation, Shandong University. His research interests include intelligent monitoring of supporting structures, traffic safety, and pavement structure.

XIUGUANG SONG received the B.S. degree in hydropower engineering from the Department of Hydraulic Engineering, Shandong University of Technology, in 1989, and the Ph.D. degree in geotechnical engineering from the Institute of Geotechnical Engineering, Hohai University, in 2000. He worked as an Associate Professor with the School of Civil Engineering, Shandong University of Technology, in 2000. He was a Visiting Scholar with the University of Kansas, in 2016. He is currently a Professor with the Department of Traffic Engineering, School of Qilu Transportation, Shandong University. His research interests include intelligent transportation systems, roadbed disease and treatment, and intelligent monitoring of supporting structures.

• • •