

# LiDAR Based Negative Obstacle Detection for Field Autonomous Land Vehicles



## Erke Shang

College of Mechatronic Engineering and Automation, National University of Defense Technology, Changsha, P.R. China, 410073  
e-mail: erke1984@qq.com

## Xiangjing An, Tao Wu, and Tingbo Hu

National University of Defense Technology, Changsha, P.R. China, 410073  
e-mail: AnXiangjingNUDT@163.com, wt.cs@163.com, hutingbo06@163.com

## Qiping Yuan

Beijing Special Engineering Design and Research Institute, Beijing, P.R. China, 100028  
e-mail: Yuanqiping@139.com

## Hangen He

National University of Defense Technology, Changsha, P.R. China, 410073  
e-mail: Hehangen@gmail.com

Received 22 November 2014; accepted 12 May 2015

Negative obstacles for field autonomous land vehicles (ALVs) refer to ditches, pits, or terrain with a negative slope, which will bring risks to vehicles in travel. This paper presents a feature fusion based algorithm (FFA) for negative obstacle detection with LiDAR sensors. The main contributions of this paper are fourfold: (1) A novel three-dimensional (3-D) LiDAR setup is presented. With this setup, the blind area around the vehicle is greatly reduced, and the density of LiDAR data is greatly improved, which are critical for ALVs. (2) On the basis of the proposed setup, a mathematical model of the point distribution of a single scan line is deduced, which is used to generate ideal scan lines. (3) With the mathematical model, an adaptive matching filter based algorithm (AMFA) is presented to implement negative obstacle detection. Features of simulated obstacles in each scan line are employed to detect the real negative obstacles. They are supposed to match with features of the potential real obstacles. (4) Grounded on AMFA algorithm, a feature fusion based algorithm is proposed. FFA algorithm fuses all the features generated by different LiDARs or captured at different frames. Bayesian rule is adopted to estimate the weight of each feature. Experimental results show that the performance of the proposed algorithm is robust and stable. Compared with the state-of-the-art techniques, the detection range is improved by 20%, and the computing time is reduced by an order of two magnitudes. The proposed algorithm had been successfully applied on two ALVs, which won the champion and the runner-up in the “**Overcome Danger 2014**” ground unmanned vehicle challenge of China. © 2015 Wiley Periodicals, Inc.

## 1. INTRODUCTION

A field autonomous land vehicle (ALV) is a kind of robot that can perform desired tasks without human guidance in a field environment. To drive safely, both positive and negative obstacles should be detected by the vehicle as far as possible (Zhou et al., 2012). Much attention has been paid to obstacle detection and terrain analysis (Shang, Li, Ye, & He, 2013; Chen Tongtong, Dai Bin, & Daxue, 2013), but still few ALVs could drive well in a field environment. The issue of positive obstacle detection is widely investigated, and many distinguished detection algorithms have come forth (Pedro Santana & Barata, 2010). Negative obstacles for vehicles re-

fer to ditches, pits, or terrain with a negative slope, which are dangerous factors in travel. Due to its negative properties, negative obstacle detection from a distance is still a big challenge: because the negative obstacle is below the ground level and therefore hard to be viewed by vehicle’s sensors.

On the basis of sensors employed in tasks, negative obstacle detection is usually classified into three primary kinds in literatures. In the first kind, a thermal infrared camera is employed. In the second kind, two cameras are introduced to create a three-dimensional (3-D) stereo vision (Wu & He, 2011). In the third kind, LiDARs are adopted in both positive and negative obstacle detections (Larson & Trivedi, 2011; Heckman, Lalonde, Vandapel, & Hebert, 2007).

In traditional obstacle detection methods (Thrun et al., 2006; Kammel et al., 2008), LiDAR data are first mapped

Direct correspondence to: Erke Shang, email: erke1984@qq.com



**Figure 1.** The proposed setup of 3-D LiDARs on our ALV platform: two compact HDL-32 LiDARs are mounted on the two sides of the vehicle top with a same fixed angle (labeled by a red circle); another HDL-64 LiDAR is upright equipped on the middle of the vehicle top for comparison (labeled by a blue circle). This vehicle won the champion in the “Overcome Danger 2014” ground unmanned vehicle challenge of China, the background is the challenge spot.

onto a grid map (in order to reduce a 3-D point cloud into a 2.5-D map), in which the maximal value, the minimal value, and the median value are computed and marked in each grid. The potential obstacles can be detected in each grid cell by comparing the marked values or analyzing the relationship between adjacent grids. However, in a field environment, terrain may not be even, and only a few parts of the negative obstacle could be viewed by onboard sensors. The depth of the negative obstacle in the grid map would not be salient. Therefore, the traditional grid map based method would not be sufficient for detection in a field environment. In this paper, the relationship between adjacent scan points in the same scan line is employed. There are two advantages: first, the relationship between adjacent scan points won't be affected by the ground's bumpiness; second, instead of detecting obstacles from the whole grid map, this algorithm reduces the computation complexity.

This paper presents a LiDAR feature fusion based algorithm (FFA) to fulfill the negative obstacle detection for field ALVs. First, a novel LiDAR setup method is presented: two compact 3-D LiDARs are mounted on the two sides of the vehicle roof, as shown in Figure 1. The proposed new

setup has two merits: (1) the blind region around the vehicle is greatly reduced, which is very important for the field ALV driving on narrow roads or making a turn. (2) Compared to the upright setup in the traditional way, the density of LiDAR data is greatly improved, which is very beneficial to detect both positive and negative obstacles. Based on the novel setup, a mathematical model of the point distribution of a single scan line is deduced. Then, two negative obstacle detection algorithms, an adaptive matching filter based algorithm (AMFA) and a feature fusion based algorithm are presented on the mathematical model. Experimental results show that the proposed algorithm is valid and reliable. This proposed algorithm has been successfully applied in two ALVs, which won the champion and the runner-up in the “Overcome Danger 2014” ground unmanned vehicle challenge of China (Figure 1 shows our ALV that won the champion, the background is the challenge spot).

The remainder of this paper is organized as follows. Section 2 reviews some related works on negative obstacle detection. Section 3 discusses the drawbacks of the traditional setup and describes the details of the proposed setup method. In Section 4, a mathematical model of the point

distribution for a single scan line is deduced, which is used to generate an ideal scan line to simulate the real scan line. In Section 5, an AMFA is proposed. In Section 6, based on AMFA algorithm, an FFA is introduced to improve the detection performance. In Section 7, experimental results show that the proposed algorithm is robust and stable. Section 8 concludes the paper.

## 2. RELATED WORKS

Negative obstacle detection is a key module in field ALVs, which has received massive attention. As stated in the introduction, three ways are always carried out to implement negative obstacle detection. The first one was to use a thermal infrared camera (Matthies & Rankin, 2003). The principle was that the negative obstacle tended to be warmer than the surrounding terrain at night. By detecting the warmer region in the thermal image, the potential negative obstacles could be detected. The limitation was that it only worked at night, and the temperature of the terrain tended to be affected by weather conditions (A. Rankin & Matthies, 2007). In addition, living organisms, such as animals or leaves near the obstacles could also lead to temperature rise.

The second method was to analyze the geometrical character of the negative obstacle from images or stereo visions. An image sequence based negative obstacle detection algorithm was proposed in paper (Wu & He, 2011), in which both color appearance models and geometrical cues were extracted for detection. In that algorithm, when negative obstacles were far from the vehicle, color appearance models were used as the principal cues for detection; when negative obstacles went nearer, geometrical cues were extracted from the stereo vision for detection. Paper (Fazli, Dehnavi, & Moallem, 2011) also developed a computer vision algorithm for negative obstacle detection based on the stereo vision technique. The limitation of the vision based algorithm was that the camera could be easily disturbed by illuminations. In addition, the complex texture of a field environment would bring great trouble for feature detection from images.

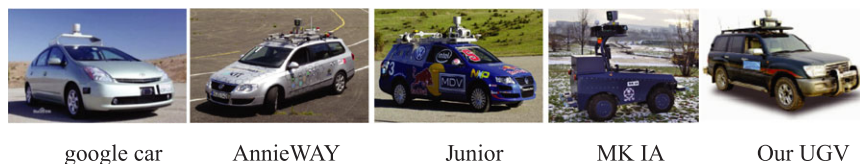
Recently LiDAR is widely applied in obstacle detection since they can accurately get the range information (Larson & Trivedi, 2011; Han et al., 2012). The TerraMax vehicle, one of the five vehicles that has successfully finished the DARPA Grand Challenge desert race in 2005 (Braid, Broggi, & Schmiedel, 2006), employed two single line LiDARs to detect negative obstacles. In their task, negative obstacles were mainly presented as negative road edges or cliff edges, because small ditches or pits in the road surface would not bring risks to TerraMax vehicle. Their algorithm to detect negative road edges was designed as follows: translate the LiDAR data into a local coordinate and compare the relative heights of neighboring scan points. During the process, a history of scan data were maintained in a database (similar to a global map). Detection operation was implemented

in that database, and these negative height discontinuities in the database would be considered as potential negative obstacles. To improve the detection performance, multi-cue detection approaches by different kinds of sensors were also widely employed (A. Rankin & Matthies, 2007; Manduchi, Castano, Talukder, & Matthies, 2005; Dima, Vandapel, & Hebert, 2004). In paper (A. Rankin & Matthies, 2007), both the geometry based cues from the stereo range data and the thermal signature based cue from thermal infrared imagery were applied for negative obstacle detection. Paper (Manduchi et al., 2005) tried to combine two sensors, a color stereo camera and a single line LiDAR, to implement detection. In that algorithm, it analyzed the slope of each surface patch before the vehicle, which was generated from the color stereo camera; then those identified patches represent a hurdle for the vehicle. Paper (Dima et al., 2004) considered obstacle detection as an inference problem, since no sensor could directly measure "obstacleness": the sensors needed to infer information about obstacles from measurements of colors, temperatures or shapes. Thus, they combined color and infrared (IR) imagery with range information from a single line LiDAR for obstacle detection.

Nowadays, multi-line LiDAR becomes more and more popular in both positive and negative obstacle detections. A Negative Obstacle DetectoR (NODR) based algorithm was introduced in paper (Larson & Trivedi, 2011), in which a HDL-64 LiDAR was selected as the on-board sensor. That algorithm "erred on the side of detecting negative obstacles generously" and then labeled them as potential negative obstacles. Then, NODR classified these potential negative obstacles by detecting gaps, an absence of data, where there could exist a real ditch, pit, cliff, or a negative slope.

Paper (Heckman et al., 2007) also used a HDL-64 LiDAR to carry out obstacle detection. In that paper, a missing data interpretation based approach was adopted to label the potential negative obstacles. Three steps were contained in that approach: "the 3-D data accumulation and low level classification, the 3-D occluder propagation, and the context-based occlusion labeling". However, when the negative obstacle was far away from the vehicle, the height of the trailing edge was very small in LiDAR data, which made it very hard to distinguish the negative obstacle from ground in a field environment.

A ground segmentation method with LiDAR is another choice in obstacle detection. In paper (Larson, Trivedi, & Bruch, 2011), the off-road terrain was first analyzed according to a point cloud produced by a 3-D LiDAR, and then the potential hazards were determined above or under the ground. Paper (Morton & Olson, 2011) also described a height-length-density (HLD) terrain classifier that generalized some prior methods and provided a unified mechanism for both positive and negative obstacle detections. Similar works had also been carried out in our team (Chen et al., 2013): an algorithm for real-time segmenting 3-D scans of various terrains was presented. However, the ground



**Figure 2.** The mainstream ALVs: 3-D LiDARs are upright fixed on these platforms.

segmentation based method was not sufficient in a field environment. For example, when the LiDAR was fixed at a height of  $H = 2\text{ m}$ , and the distance between the pit and the vehicle  $D = 8\text{ m}$ . Suppose the width of this pits was  $0.6\text{ m}$ . In an ideal case, as far as  $0.15\text{ m}$  of the depth of the pit can be viewed by the LiDAR. But there usually existed a bumpiness of  $0.15\text{ m}$  on the ground of surface in a field environment. Therefore, that ground segmentation based algorithm can't operate in a field environment.

To improve the traditional grid map based algorithm, paper (Montemerlo & Thrun, 2006) presented a multi-grid representation approach to detect negative obstacles, by combining maps with different resolutions. The map chosen for each detection depended on the overall range: the further a detection point, the coarser the corresponding grid. The key of this approach was to have such an information-resolution map that was generated by a simultaneous localization and mapping (SLAM) algorithm. Unfortunately, SLAM itself in a field environment is a difficult problem.

Sinha et al. successfully applied their gap detection approach on an urban search and rescue robotic recently (Sinha & Papadakis, 2013). The key contribution of that paper was processing 3-D range data by using 2-D image algorithm: (Page 1) "the proposed approach is based on the application of efficient image morphological operations for noise reduction and border following the detection and grouping of gaps." The capability of the proposed approach mainly depended on the LiDAR data density, which was used to generate a 3-D point cloud. However, this paper didn't explain how to generate the LiDAR cloud and its density.

Three-dimensional LiDARs are also employed for negative obstacle detection in this presented paper. Considering that the ground could be bumpy in a field environment, instead of the traditional grid map, features generated by adjacent points are adopted in the proposed algorithm. A novel 3-D LiDAR setup is first introduced to improve the LiDAR data density. Under this setup, the scan line of the LiDAR becomes much denser, which is very beneficial in detection. Based on this setup, a mathematical model of the point distribution for a single scan line is deduced. With this mathematical model, an AMFA algorithm and an FFA algorithm are presented for negative obstacle detection. Experimental results show that the proposed algorithms are robust and stable.

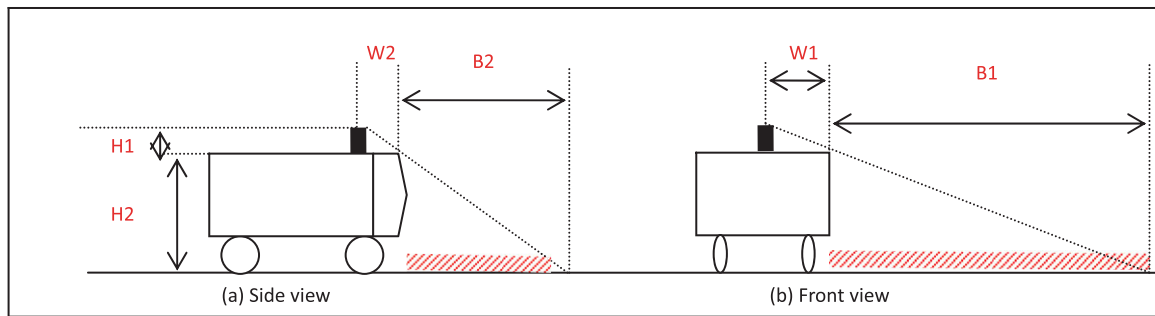
### 3. A NOVEL SETUP METHOD OF 3-D LIDARS

Three-dimensional LiDARs are widely applied on ALVs for obstacle detection. A type of velodyne HDL-64E LiDAR is widely adopted in mainstream ALVs, such as in Google self-driving car, KIT's AnnieWAY (Kammel et al., 2008), Stanford's Junior (Montemerlo et al., 2008), the vehicle in paper (Häselich et al., 2012), and the vehicle in our own team (Chen et al., 2013) (shown in Figure 2).

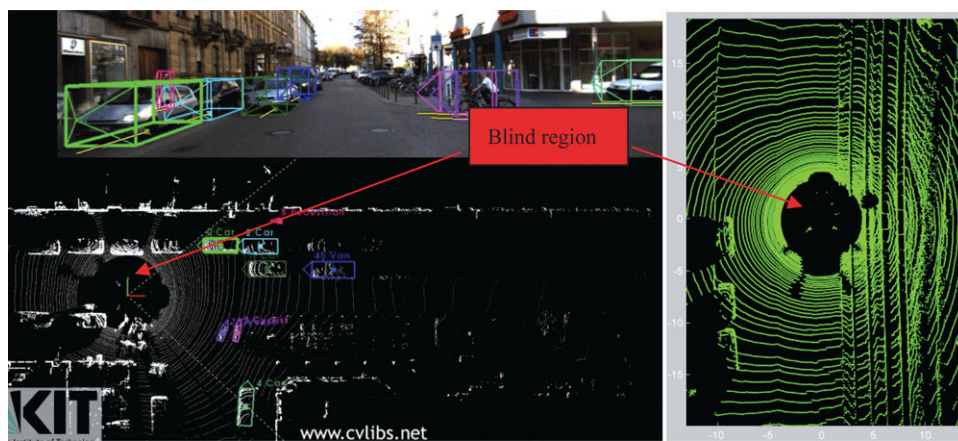
#### 3.1. Drawbacks of the Traditional Upright Setup

Three-dimensional LiDARs were usually equipped upright on the vehicle roof in literatures. There are two drawbacks of the upright setup: first, a large blind region would be produced around the vehicle. The blind area might not be fatal to the ALV when driving in structured flat environments, for roads in these environments are wide enough and the borders on two sides could help keep the vehicle safe (Chris Urmson & etc, 2008). However, in a field environment, the blind area would bring mortal risks to the vehicle, for roads in the field are usually narrow and unstructured with ditches and pits. Under the traditional upright setup, the blind region around the vehicle can be analyzed as follows (Figure 3): the distance from the blind region to the side of the vehicle is  $B1 = W1 \times (H1 + H2)/H1 - W1$ , where  $W1$  is the half width of the vehicle,  $H1$  is the height of the mounted LiDAR, and  $H2$  is the height of the vehicle. The size of the blind region in front of the vehicle is  $B2 = W2 \times (H1 + H2)/H1 - W2$ , where  $W2$  is the distance from the LiDAR to the head of the vehicle. The blind areas are shown as in Figure 3.

The blind region produced in experiments under the traditional setup is shown in Figure 4. Figure 4 shows the visible region of an ALV by a LiDAR sensor. The result in Figure 4 is offered by (Geiger, Lenz, & Urtasun, 2012), in which the width of AnnieWAY is about  $2.2\text{ m}$ , while the diameter of the blind region is longer than  $8\text{ m}$ . It is very dangerous if obstacles appear in this region or if the ALV makes a turn. In addition, as the sensor head spins, each laser horizontally generates points of a high density, but the spacing between adjacent scan lines becomes sparser as the distance from the sensor increases (W. Shane Grant & Itti, 2013). As shown in Figure 4, the distance between two scan lines is as sparse as  $0.5\text{ m}$  when the scanned lines is  $10\text{ m}$



**Figure 3.** Analysis of the blind area under the traditional setup of a 3-D LiDAR.



**Figure 4.** Analysis of the blind area under the traditional setup of a 3-D LiDAR in real experiments. This figure is offered by (Geiger et al., 2012).

away from the vehicle. It is almost impossible to identify obstacles when the width of the obstacle is smaller than  $0.5\text{ m}$ .

### 3.2. Details of the Novel Setup of 3-D LiDAR

Concerning the drawbacks of the traditional upright setup, this paper presents a novel setup of 3-D LiDAR to overcome those drawbacks. The proposed 3-D LiDAR setup is illustrated in Figure 1. Two LiDARs are mounted on the two sides of the vehicle top: one is toward the left and the other is toward the right. Both are equally angled forward.

There are three rules to obey in this setup: (1) make sure that no lasers from two LiDARs illuminate each other directly; otherwise, the sensors will be damaged; (2) the visual range around the vehicle depends on  $\theta$  (Fig. 5(a)); (3) the visual range in front of the vehicle and the region of overlapped scans depends on  $\phi$  (Figure 5(b)). The visual range with the proposed setup is shown in Figure 5, in which angles and their corresponding visual ranges are shown on two viewpoints: a front view and a top view. The overlapped

area, labeled in Figure 5(b), greatly improves the detection capability.

From Figure 5(b), we can see the blind area around the vehicle is reduced markedly with the novel setup. The region on two sides is now visible to the relevant LiDAR. The front region is scanned by both LiDARs, which greatly enhances the resolution and improves the scene understanding capability. Under this setup, the geometrical characters of the negative obstacle in the LiDAR data become much more distinct than that under the traditional way. With this new setup, negative obstacle detection becomes more reliable.

### 3.3. Comparison

Experiments are designed to evaluate the benefit from the proposed setup. A pair of HDL-32E compact LiDARs is selected under the proposed setup on our ALV platform (Figure 1). Another HDL-64 LiDAR is fixed upright in the center of the vehicle top for comparison (Figure 1). Fifteen objects, with the same height of  $0.5\text{ m}$ , are scattered around

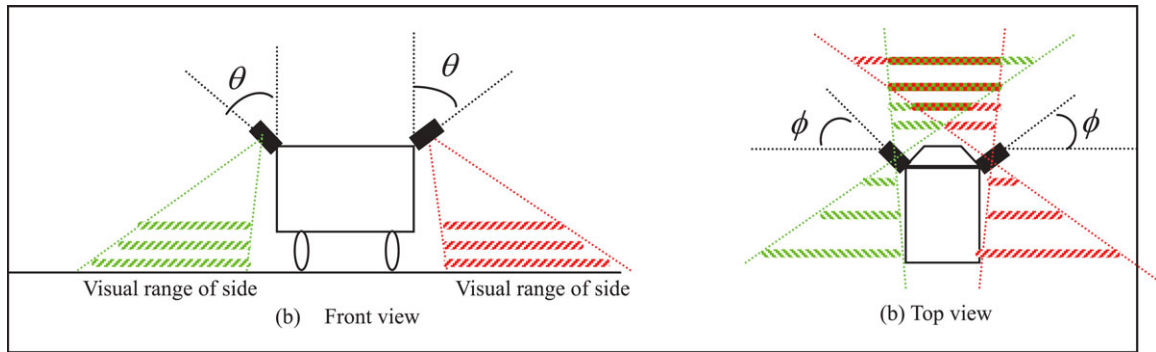


Figure 5. Details of the proposed setup. The visual range depends on  $\theta$  and  $\phi$ .

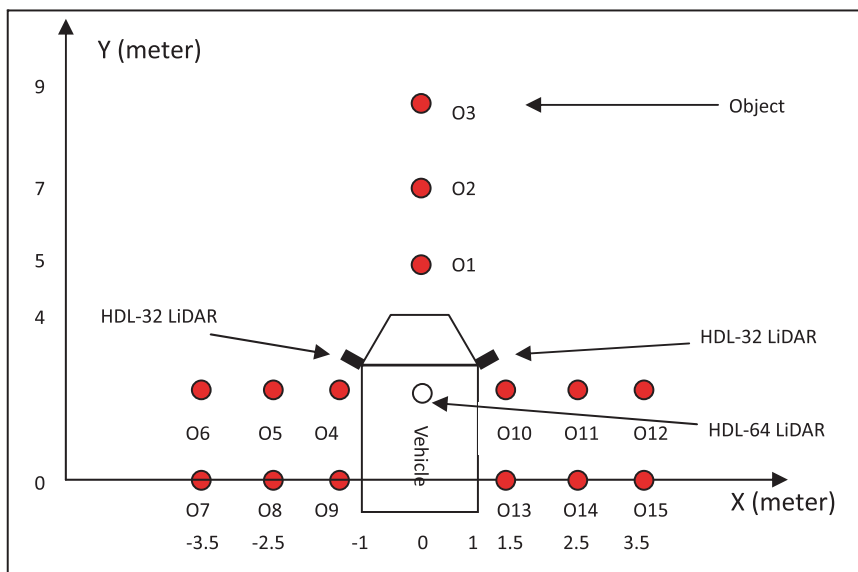


Figure 6. The visual range comparison experiments: between the traditional upright setup and the proposed setup.

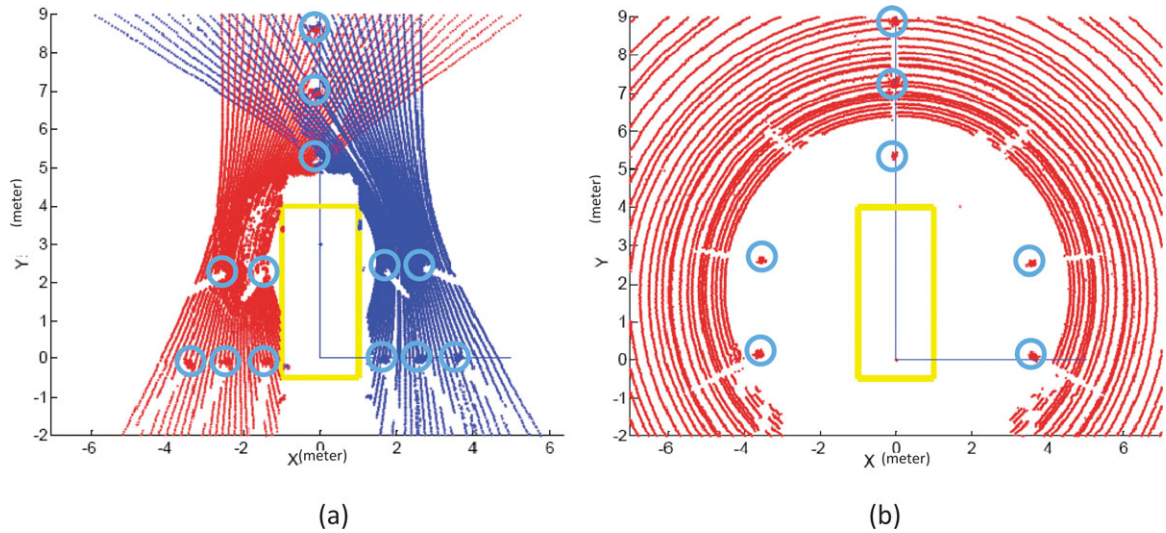
the vehicle to check the visual range of all LiDARs (shown in Figure 6).

Figure 7(a) shows the visual range generated under the proposed setup by two HDL-32 LiDARs and Figure 7(b) shows the visual range generated under the traditional upright setup by an HDL-64 LiDAR. The distribution of LiDARs and objects are labeled in Figure 6. It is found that most of the objects can be viewed by LiDARs under the proposed setup, while many objects get lost in the compared view. These objects would bring risks to the ALV, especially in a field environment. It can be inferred that the proposed setup can practically benefit the ALV.

Another advantage of the proposed setup is that the scan line is much denser in the range of interest (ROI). With this new setup, the scan range in front of the vehi-

cle overlaps in the experiment, which means more points are distributed in this region. Thus, the proposed setup improves the resolution of the LiDAR for obstacle detection. Another experiment is specially designed to demonstrate this improvement (Figure 8). Two obstacles (one of them is a person) are standing in front of the vehicle: one is 6 meters away from the vehicle, labeled as  $O_1$ ; the other is 9 meters away, labeled as  $O_2$ . Both obstacles can be viewed by three LiDARs.

To demonstrate the benefit of the proposed setup, the number of LiDAR points distributed on obstacles are listed in Table I. From Table I, we can read that obstacles get more distributed LiDAR points under the proposed setup, which means the two obstacles are more easily to be detected.



**Figure 7.** The comparison experiments shows the visual ranges of two setups: (a) shows the visual range generated under the proposed setup; (b) shows the visual range generated under the traditional way.

#### 4. THE MATHEMATICAL MODEL OF THE PROPOSED SETUP

With the proposed setup, the 3-D data density is improved remarkably, and the geometrical character of the negative obstacle is much more salient than that with the traditional setup. With this novel setup, a mathematical model of the point distribution of a single scan line is deduced. The deduced mathematical model is applied in the following AMFA algorithm and FFA algorithm.

##### 4.1. Difference Between the Scan Point Distributions Under Two Setups

The geometrical character of a typical negative obstacle is shown in Figure 9(a).  $L$  specifies the width of the obstacle, and  $H$  specifies the height of the mounted LiDAR. The scan lines are distributed over the obstacle, and  $D$  specifies the distance between LiDAR and the obstacle.

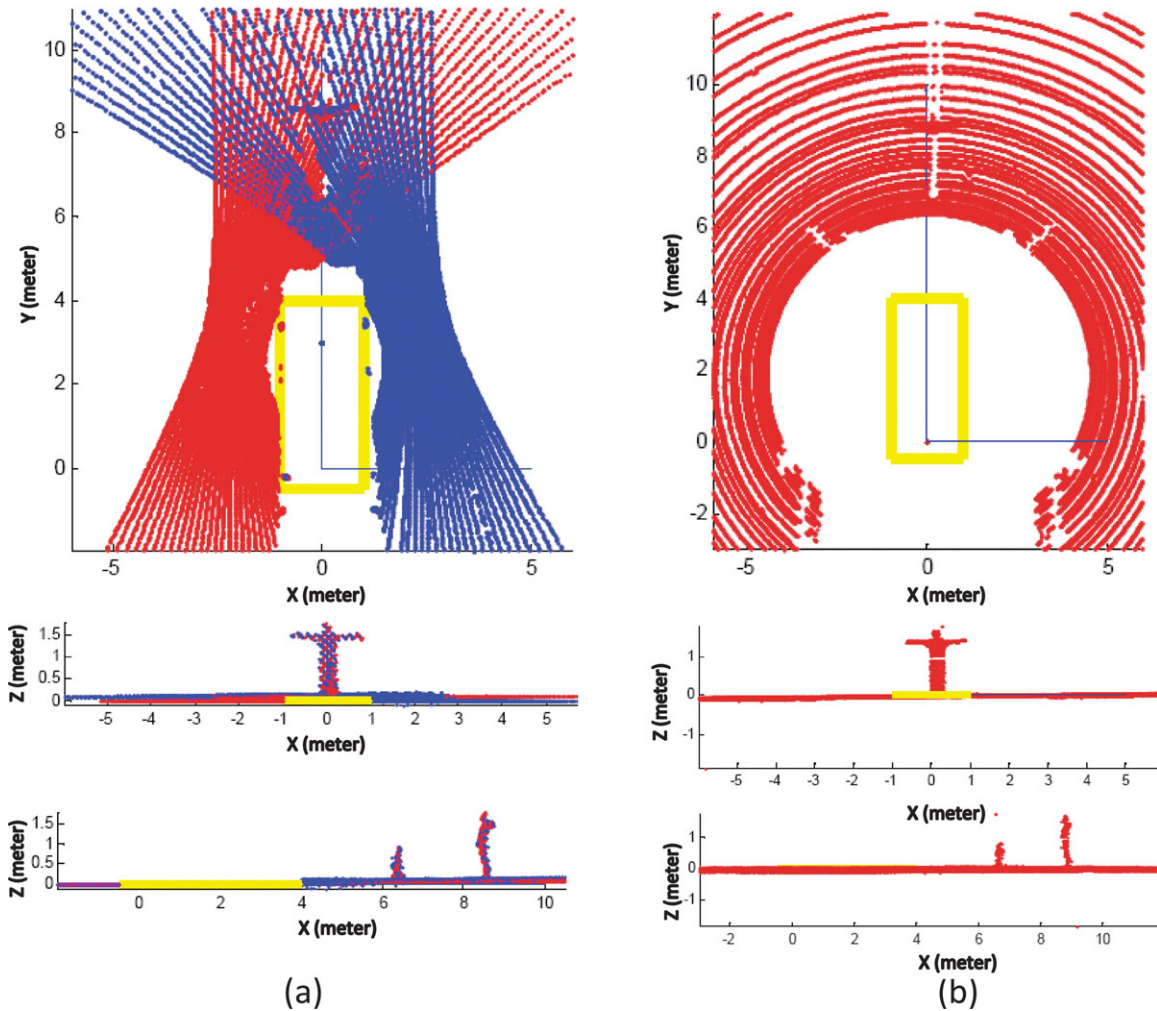
Two cues are widely adopted in negative obstacle detection. One is the distance between two adjacent scan points. The distance between the scan points on the obstacle will become much longer if a negative obstacle appears. As shown in Figure 9(a),  $d_{P_1 P_2}$  is much bigger than  $d_{P_1 P'_2}$  when a pit is scanned by the LiDAR scan line. In addition, the distance between the proceeding scan points will become shorter than usual. For example,  $d_{P_2 P_3}$  is much smaller than  $d_{P'_2 P'_3}$ . This is the principal geometrical character of negative obstacles with a LiDAR sensor. The other cue is the slope of the up-side (the back) of the ditch. There are always several scan points that are lower than others in  $Z$  axis (for

instance,  $P_2, P_3$  are lower than  $P_1, P_4$ ). These two cues are both employed as features in our algorithm.

When the LiDAR is upright mounted, the scan lines will be distributed as shown in Figure 9(b). The scan points ( $P_1, P_2, P_3, P_4$ ) scattered on the obstacle locate in different scan lines. The angles between every two adjacent scan lines depend on the intrinsic parameter of the LiDAR. In comparison, by setting the LiDAR under the proposed setup, the scan points ( $P_1, P_2, P_3, P_4$ ) are generated by the same scan line, as shown in Figure 9(c). The benefit of this new setup is that these scan points are much denser, and the angle between every two points is fixed.

From the intrinsic parameter of the HDL-32E LiDAR (Velodyne LiDAR, 2012), we know that the 32 scan lines cover a  $41.3^\circ$  vertical field of view, which means the average angle of  $\theta_i$  between every two adjacent scan lines is about  $1.29^\circ$ . Besides, 2,000 scan points covers a total  $360^\circ$  horizontal field of view, which means the angle  $\theta_i$  between every two adjacent scan points of a same scan line is  $0.18^\circ$ . Suppose the LiDAR is equipped at a height of  $H = 2\text{ m}$ , and a negative obstacle is to be located away from the vehicle at a distance of  $D = 10\text{ m}$ , then according to Eq.(2),  $d_{P_1 P_2} = 1.32\text{ m}$  when the LiDAR is traditionally upright fixed, while  $d_{P_1 P'_2} = 0.17\text{ m}$  when the LiDAR is fixed under the proposed setup. Because the distance between points on the ground is 1.32m under the upright setup, a pit with or smaller than this size could not be detected. The proposed setup reduces this distance to 0.17 m; therefore, the same pit can be detected under the proposed setup.

$$d_{P_1 P'_2} = \tan \left( \arctan \left( \frac{D}{H} \right) + \theta_i \right) \times H \quad (1)$$



**Figure 8.** The comparison experiments for positive obstacle detection of two setups: (a) shows the result generated by the proposed setup; (b) shows the result generated by the traditional setup.

**Table 1.** The point distribution on positive obstacles under the proposed setup and the traditional setup.

	$O_1$	$O_2$
Traditional upright setup	243	597
The proposed setup	322	740

#### 4.2. The Mathematical Model of the Point Distribution of a Single Scan Line

When the LiDAR is mounted under the proposed setup and calibrated onto the vehicle coordinate frame (the calibration approach can be found in our previous work (Shang et al., 2014)), the scan point distribution is shown as follows:

- 1) The angle between every two adjacent points is a fixed angle  $\theta_i$  (Figure 10), which can be figured out by equation (2).

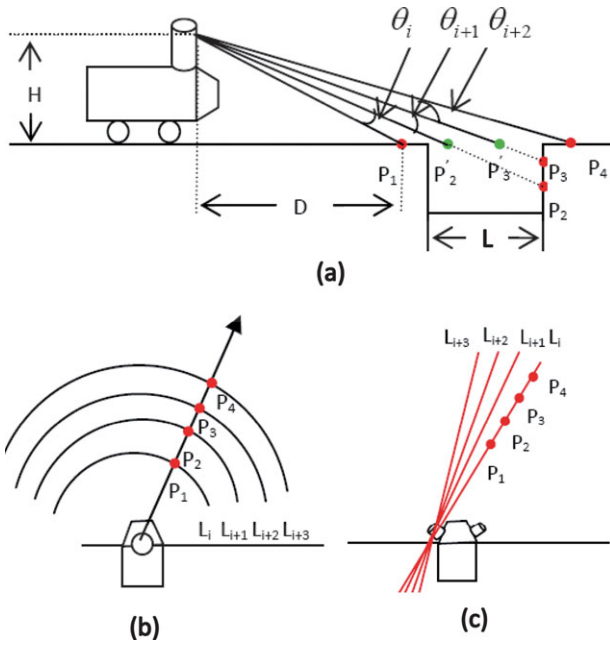
In Figure 10,  $\theta'_i$  depends on the number  $M$  when the scan line works in a circle. Thus,  $\theta'_i = \frac{360^\circ}{M}$ , where  $M$  is a fixed parameter.  $\beta_i$  is also a fixed intrinsic parameter of the LiDAR.

$$\sin(\theta_i/2) = \sin(\theta'_i/2) * \cos(\beta_i) \tag{2}$$

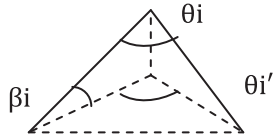
- 2) The relationship between  $D$  and  $t$  can be expressed as in Eq. (3), where  $t$  denotes the numbers of scan points from the vehicle to the obstacle,  $D$  denotes the horizontal distance from the vehicle to the obstacle, and  $H$  denotes the height of the mounted LiDAR.

$$t = \text{atan}(D/H)/\theta_i \tag{3}$$





**Figure 9.** Analysis of the geometrical character of the negative obstacle detection under the traditional upright setup and the proposed setup: (a) shows the geometrical character of a typical negative obstacle; (b) shows the scan points generated by the upright fixed LiDAR; (c) shows the scan points generated by the proposed setup LiDAR.



**Figure 10.** The scan angle between adjacent points in a same scan line.

- 3) The distance between every two adjacent scan points can be expressed as  $W = P(t + 1) - P(t)$ , where  $P(t) = D$ , and  $P(t + 1)$  can be estimated by equation (4).

$$P(t + 1) = \tan(\text{atan}(D/H) + \theta_i) * H \quad (4)$$

- 4) The number of scan points on the negative obstacle  $Num$  can be expressed as in Eq. (5):

$$\begin{cases} t_1 = \text{atan}(D/H)/\theta_i \\ t_2 = \text{atan}((D + L)/H)/\theta_i \\ Num = t_2 - t_1; \end{cases} \quad (5)$$

- 5) The mathematical model of a negative obstacle in a single scan line can be expressed as follows:

$$\begin{cases} t = \text{atan}(D/H)/\theta_i; \\ Num = \text{atan}((D + L)/H)/\theta_i - \text{atan}(D/H)/\theta_i; \\ x(t) = D; \\ x(t + 1 : t + Num) = D + L; \\ y(t) = 0; \\ y(t + i) \approx -H * L * (Num - i)/D/Num; i \in [1, Num] \end{cases} \quad (6)$$

Where  $x$  denotes the distance between the vehicle and the obstacle, and  $y$  denotes the height of each scan point.

According to the mathematical model of the negative obstacle (Eq. (6)), a scan line can be generated when the parameter of the negative obstacle is set. Scan lines both generated by the mathematical model and by the real LiDAR are shown in Figure 11. Figure 11(a) is an image scene where an obstacle is located before the vehicle. Figure 11(b) shows the scan line generated by the mathematical model. Figure 11(c) shows the features of width and height of the obstacle in a single scan line. Figure 11(d) shows the real scan line generated by an onboard LiDAR. Figure 11(e) shows the width feature and height feature of the real obstacle. It can be found that the scan line generated by the mathematical model resembles the real scan line.

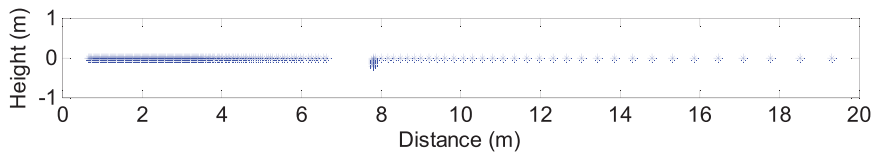
### 5. ADAPTIVE MATCHING FILTER BASED ALGORITHM (AMFA) FOR NEGATIVE OBSTACLE DETECTION

From Figure 11, we can see that the width and the height are distinguished when an obstacle emerges in the scan line. Thus, the obstacle can be detected by analyzing the two features in each scan line. Each scan line can also be considered as a signal wave. Therefore, the issue of obstacle detection can be transformed into a signal detection problem in each wave. The mathematical model of the obstacle can be described as  $F(H, D, L, i)$ , where  $H$  denotes the height of the mounted LiDAR,  $D$  denotes the distance between the obstacle and the vehicle,  $L$  denotes the width of the obstacle, and  $i$  denotes that this mathematical model is generated by the  $i^{th}$  scan line.

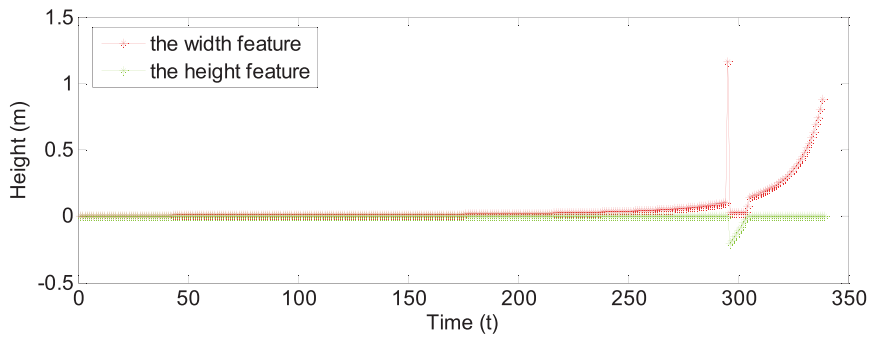
The idea of AMFA algorithm is that at first simulated obstacles with different sizes at different positions are generated by the mathematical model; then, features of these simulated obstacles are employed as adaptive filters to match with the real scan lines. The potential obstacle will be identified if the features of simulated obstacles match with the real scan lines, and a peak will emerge at the relevant position in the scan line.



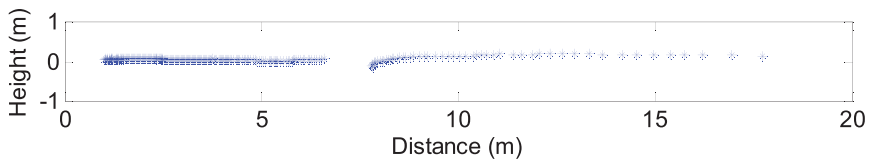
(a) Experiment scene (a ditch is placed in front of the vehicle)



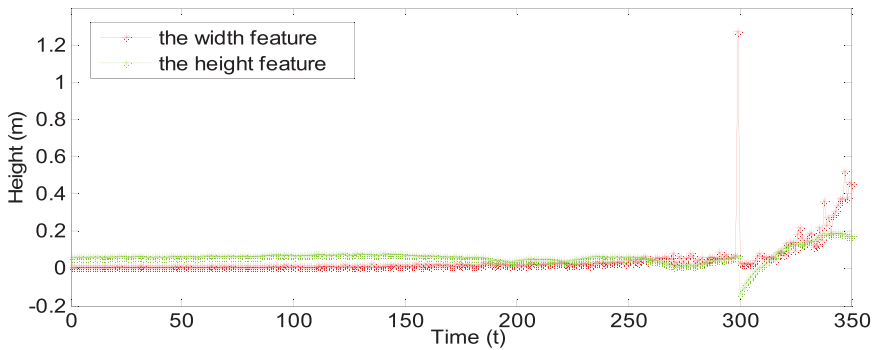
(b) The  $i^{\text{th}}$  scan line wave generated by the mathematical model



(c) The width feature and the height feature of a scan line generated by the mathematical model



(d) The  $i^{\text{th}}$  scan line wave generated by real LiDAR data



(e) The width feature and the height feature of a scan line generated by real LiDAR data

**Figure 11.** The scan line and its features generated by the mathematical model and generated by the real scan line.

This approach can be summarized as in Algorithm 1.

---

Algorithm 1. Adaptive matching filter based algorithm for negative obstacle detection.

---

**Require:**

- the mathematical model  $F(H, D, L, i)$  of a scan line;
  - 1: Traverse the distance parameter  $D$ , where  $D \in [RangeD1, RangeD2]$ . RangeD1 and RangeD2 denote the distance ranges for obstacle detection in this algorithm;
  - 2: Traverse the width parameter  $L$ , where  $L \in [RangeL1, RangeL2]$ . RangeL1 and RangeL2 denote the width ranges for this algorithm;
  - 3: By changing  $D$  and  $L$ , a series of ideal obstacle signals are generated by the mathematical model  $F(H, D, L, i)$ , which are labeled as  $F_{DL}$ ;
  - 4: Translate the real scan line  $W_i$  into the feature wave  $P_i$ , such as the width feature or the height feature;
  - 5: Translate the ideal obstacle signal  $F_{DL}$  into the corresponding feature wave  $P_{DL}$ ;
  - 6: Matching all  $P_{DL}$  with the real wave  $P_i$ , and accumulating the results in the D-L 2-D parameter space;
  - 7: Finding the peak in the parameter space. If there is a potential obstacle, the relevant position will produce a peak in the parameter space, according to the parameter  $D_R$  and  $L_R$ ; the parameter of the potential obstacle ( $D_R, L_R$ ).
  - 8: **return** the parameter of the potential obstacle ( $DR, LR$ ).
- 

Some typical ideal obstacle feature waves  $P_{DL}$ , generated by Step 5 in Algorithm 1 are listed in Figure 12. As in Algorithm 1, both the width parameter  $L$  and the distance parameter  $D$  are traversed with a fixed step within its range. Thus, a series of ideal obstacle signals can be generated as described in Step 3. Then, these ideal obstacle signals are transformed into different feature waves  $P_{DL}$ , as shown in Figure 12. All the waves  $P_{DL}$  are employed to match with the real wave (such as Figure 11(e)). Thus, a peak would emerge at the position of a potential obstacle in the D-L 2-D parameter space.

In our application, the distance parameter  $D \in [2m, 25m]$ , and the width parameter  $L \in [0.5m, 5m]$ , which means our algorithm can only detect negative obstacles in front of the vehicle 2 m to 25 m away, with a width from 0.5 m and 5 m. To reduce the filter numbers, the step sizes of  $D$  and  $L$  are both set 0.5 m. Thus, the size of 2-D parameter space  $[D_R, L_R]$  is  $46 \times 9$ , and a simple threshold is set to find the maximal position. During experiments, it is found that the step size of  $D$  is more sensitive to the detection results than the step size of  $L$ , for the width feature only captures one pixel that may not be matched at a relevant position. Thus, the sparse step of  $D$  would seriously affect the detection capability. A solution in our experiments is that the width feature waves are filtered by Gaussian filter beforehand.

To validate the proposed AMFA algorithm, a series of outdoor experiments are carried out on some ALV platforms. LiDARs are equipped at different heights ( $H = 1.4m$  and  $H = 2m$ ). Various negative obstacles are placed in the unstructured environment to verify the capability of the proposed algorithm. Some typical experimental results are shown in Figure 13: Figure 13(a) shows the scene where the obstacles are placed; Figure 13(b) lists the results of both LiDARs, in which the green line denotes the position of the detected obstacle. Experimental results on different kinds

of negative obstacles at different distances show that the proposed AMFA algorithm is effective.

The third row of Figure 13 demonstrates the case in which a LiDAR works in an error mode: some scan lines are absent. In this case, the proposed AMFA algorithm can't operate normally (the left LiDAR in the third row), because the proposed algorithm is based on every single scan line.

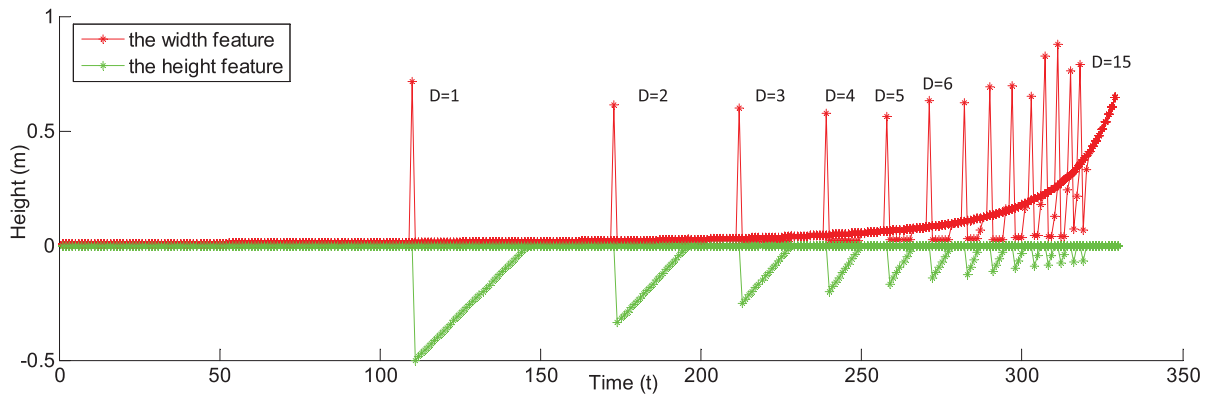
## 6. FEATURE FUSION BASED ALGORITHM (FFA) FOR NEGATIVE OBSTACLE DETECTION

Although the proposed AMFA algorithm is effective most of the time, there are still some drawbacks:

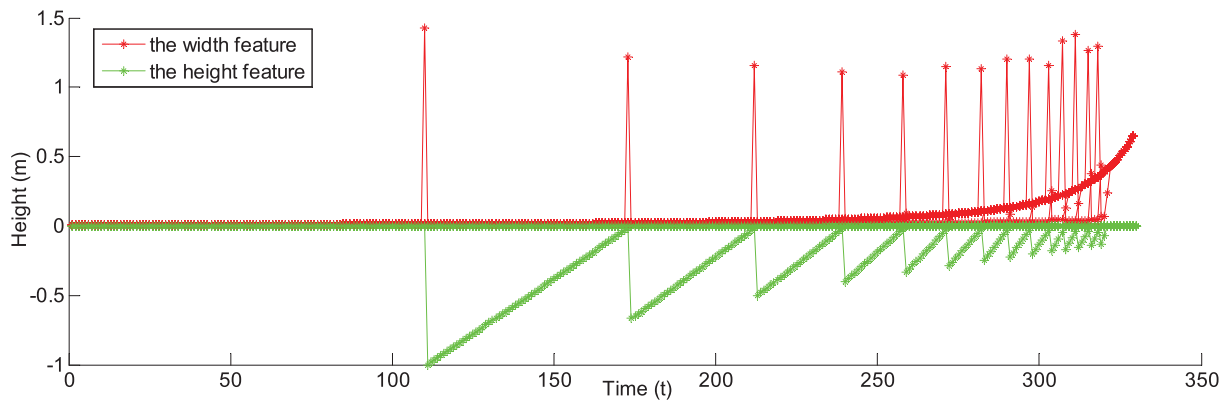
- 1) AMFA algorithm detects an obstacle in each scan line, which would give false alarms in complex environments.
- 2) AMFA algorithm would be seriously affected due to LiDAR sensor failures.
- 3) The detection results in each scan line are independent but only fused at the final step. The fusion efficiency is low.

On the basis of the AMFA algorithm, another obstacle detection algorithm is introduced. This revised algorithm fuses all features, generated by AMFA algorithm from two LiDARs into a global map according to GPS information. Thus, a history of detected features are accumulated in this global map. During the accumulating process, each detected feature would be assigned different weight to show its confidence, which is estimated by Bayesian rule. To detect the obstacle, the global map is translated into a current map. Potential negative obstacles are detected from this current map finally.

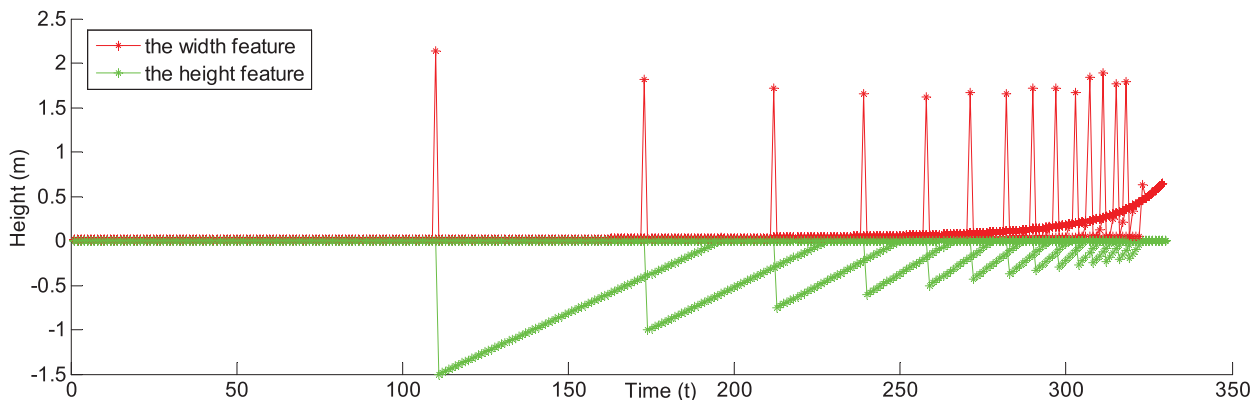
The framework of FFA algorithm can be summarized in Figure 14. As shown in Figure 14, all features, which



(a)  $L=0.5, D=[2:1:15]$



(b)  $L=1, D=[2:1:15]$

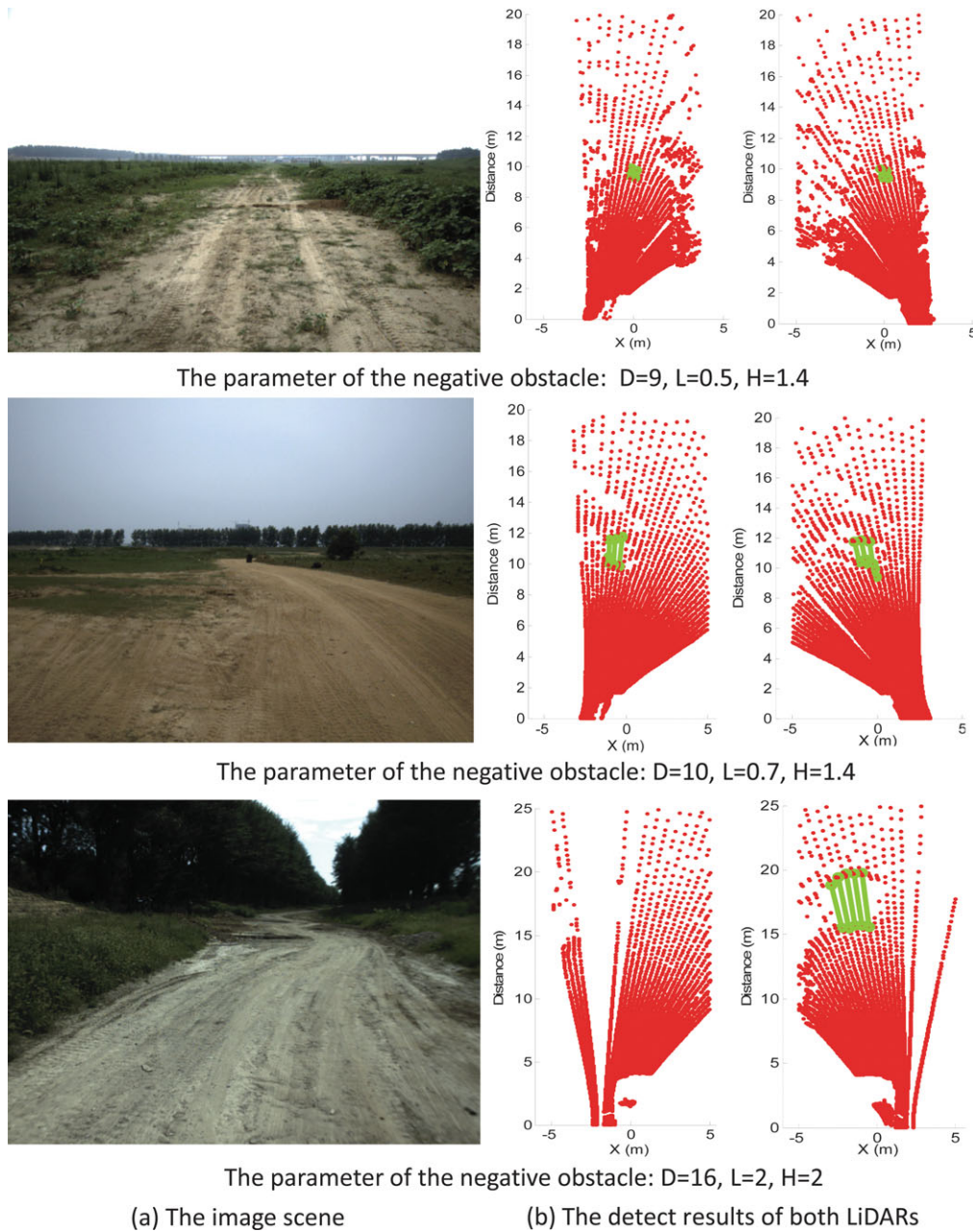


(c)  $L=1.5, D=[2:1:15]$

**Figure 12.** Some typical ideal obstacle feature waves  $P_{DL}$ .

are detected by the AMFA algorithm from different LiDARs or captured at different frames, are accumulated into a global map. This global map also includes all history features. During the accumulation process, the Bayesian rule is employed to estimate the weight of each feature.

GPS information is employed to estimate the relationship between the current map and the global map. Then, the global map is changed into the current map again, which is in the vehicle coordinate frame. Last, the filter is implemented and peaks are detected from this current map,

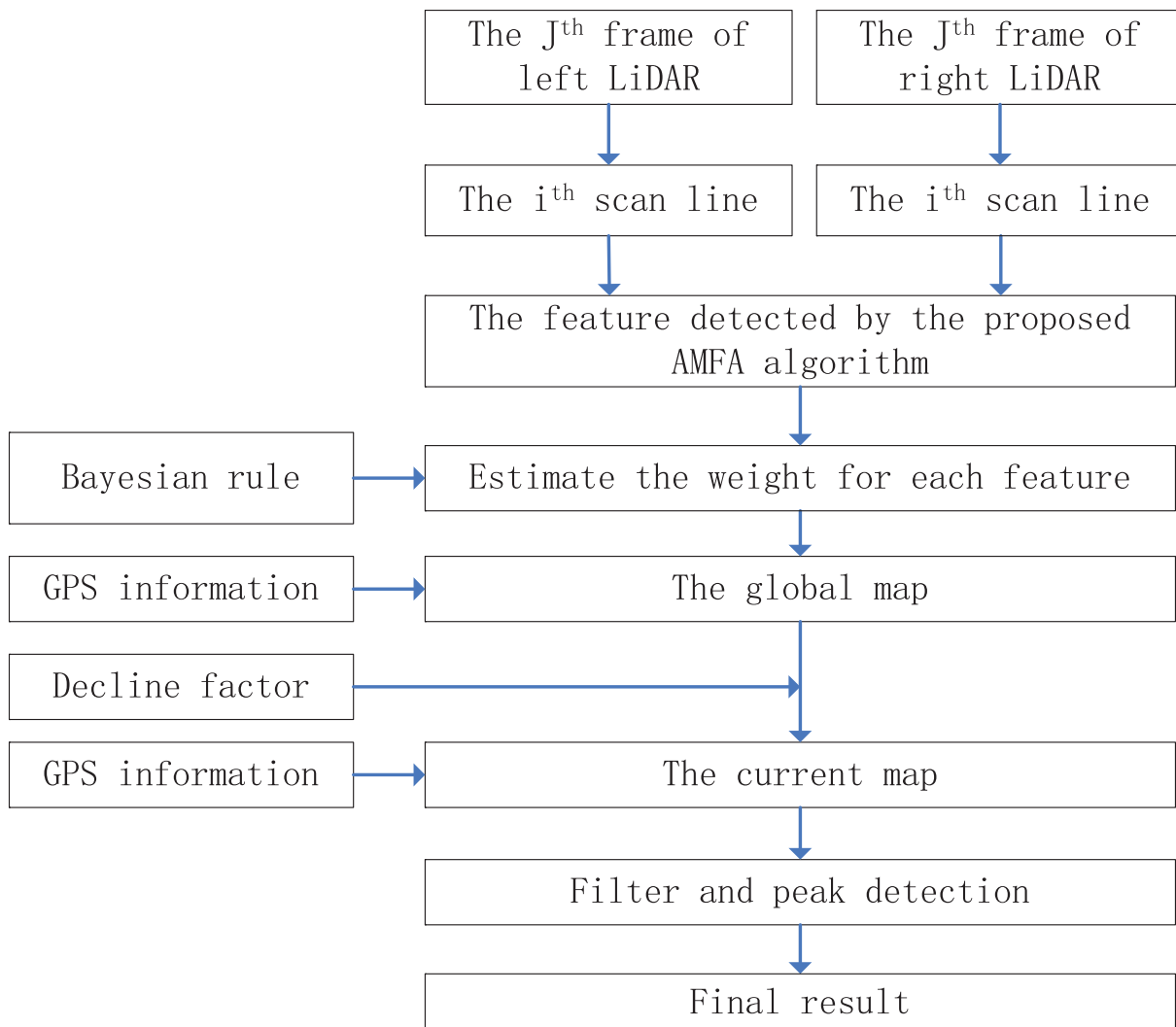


**Figure 13.** Experimental results by the proposed AMFA algorithm.

which denotes the corresponding position of the detected obstacles.

To describe the detail of FFA algorithm more clearly, Figure 15 illustrates the whole process of implementing the proposed algorithm. Figure 15(a) is the image scene where there exists two negative obstacles. Figure 15(b) and Figure 15(c) are the scan points generated by two LiDARs,

in which the potential obstacles detected by the AMFA algorithm are labeled by green lines. Both real and false obstacles are detected by the AMFA algorithm. These features are translated into a global map according to GPS information. The global map is shown as in Figure 15(d). During its accumulation process, different features from the same frame time would assign different weights, which are



**Figure 14.** The framework of the feature fusion based algorithm.

estimated by Bayesian rule. Figure 15(d) lists all of the detected features in a fixed region. To detect obstacles more conveniently, we change the global map into the current map in the vehicle coordinate frame again, which is shown as Figure 15(e). Figure 15(f) is the corresponding parameter space that accumulates all the weights of these features. Peaks are detected in Figure 15(f) and the final result is shown in Figure 15(g).

The proposed FFA algorithm incorporates the following two advantages: first, fusing the filter responses (features detected by the AMFA algorithm) in a global map before peak detection. These features are generated by different LiDARs or captured at different frames. Thus, these LiDARs are not required to be synchronous. Second, the Bayesian rule is adopted to estimate each weight of these

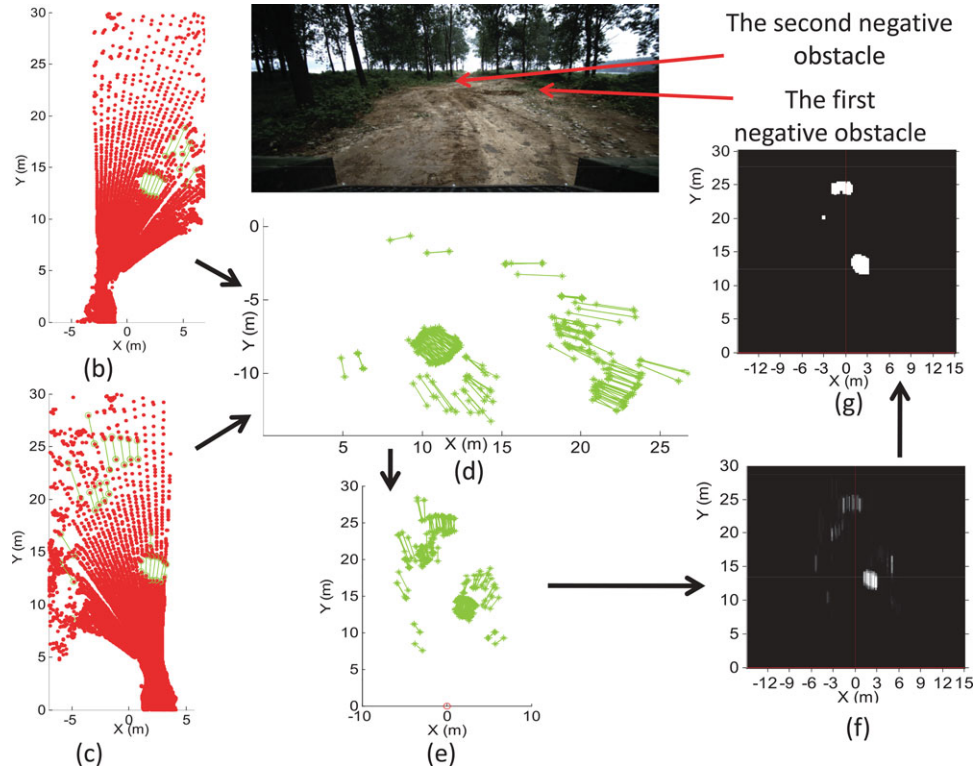
features, which is very important for accumulating peaks in the parameter space.

Bayesian rule can be described as follows:

$$P(A|X_i) = \frac{P(X_i|A)P(A)}{P(X_i)} \quad (7)$$

When there are multi-cues, Eq. (7) can be transformed into Eq. (8),

$$\begin{aligned} P(A|X_i, X_{i+1}) &= \frac{P(X_{i+1}|A, X_i)P(X_i|A)P(A)}{P(X_{i+1}|X_i)P(X_i)} = \frac{P(X_{i+1}|A, X_i)}{P(X_{i+1}|X_i)} P(A|X_i) \end{aligned} \quad (8)$$



**Figure 15.** The process of the proposed FFA algorithm. (a) the scene contains two negative obstacles; (b) and (c) the scan points generated by two LiDARs; (d) the global map, in which all feature generated by AMFA algorithm; (e) the current map under the vehicle coordinate frame, which is translated from the global map according to GPS information; (f) the parameter space that accumulates all features according to Bayesian rule. Peaks are detected in (f) and the final result is shown in (g).

According to the probability rule, Eq. (8) can be rewritten into Eq. (9),

$$\begin{aligned}
 &P(A|X_i, X_{i+1}) \\
 &= \frac{P(X_{i+1}|A, X_i)}{P(X_{i+1}|A, X_i)P(A|X_i) + P(X_{i+1}|\sim A, X_i)P(\sim A|X_i)} P(A|X_i)
 \end{aligned} \quad (9)$$

The detection process can be many scan lines used to detect potential targets independently. Thus,  $P(X_{i+1}|A, X_i) = P(X_{i+1}|A)$ . Thereby, equation (9) can be transformed into Eq. (10),

$$\begin{aligned}
 &P(A|X_i, X_{i+1}) \\
 &= \frac{P(X_{i+1}|A)}{P(X_{i+1}|A)P(A|X_i) + P(X_{i+1}|\sim A)P(\sim A|X_i)} P(A|X_i)
 \end{aligned} \quad (10)$$

The key step of applying Bayesian rule is iterating the probability in the global map. Parameters in Eq. (10) can be explained as follows:  $X_i$  specifies the feature detected by AMFA algorithm.  $A$  specifies the feature that there exists an

obstacle.  $P(A|X_i)$  specifies the probability to be an obstacle stored in the global map.  $P(X_{i+1}|A)$  denotes the probability to be an obstacle generated by the current detection.  $P(X_{i+1}|\sim A)$  is a likelihood probability, which denotes the probability not to be an obstacle according to the scan line model.

Bayesian rule in the proposed FFA algorithm is applied as follows:

- Step 1: In the traditional AMFA algorithm, each detected feature was assigned "0," which means no obstacle; or assigned "1," which means a potential obstacle by a previous threshold. Now it outputs a probability  $P_i$  instead of "0" or "1," where  $P_i \in [0, 1]$ , according to the matching level in AMFA algorithm.
- Step 2: Generalization is carried out among adjacent positions before mapping the probability  $P_i$  into the global map.
- Step 3: Read the probability  $P_i$  stored in the global map, and iterate it according to equation (10) to generate new probability  $P_{i+1}$ .
- Step 4: Write the value  $P_{i+1}$  into the global map back.

Suppose the probability from each detected feature is  $P(X_{i+1}|A) = 0.7$  to be an obstacle, the prior probability  $P(X_{i+1}|\sim A) = 0.1$ , the initial value of the whole global map (the prior probability, which is not sensitive to the detection process) is set  $P(A) = P(A|X_0) = 0.01$ . According to Eq. (10), the iterating process is illustrated as in Eq. (11):

$$\left\{ \begin{array}{l} P(A|X_1) = \frac{P(X_1|A)}{P(X_1|A)P(A|X_0) + P(X_1|\sim A)P(\sim A|X_0)} P(A|X_0) = \frac{0.7}{0.7 \times 0.01 + 0.1 \times (1 - 0.01)} \times 0.01 = 0.066 \\ P(A|X_2) = \frac{P(X_2|A)}{P(X_2|A)P(A|X_1) + P(X_2|\sim A)P(\sim A|X_1)} P(A|X_1) = \frac{0.7}{0.7 \times 0.066 + 0.1 \times (1 - 0.066)} \times 0.066 = 0.3311 \\ P(A|X_3) = \frac{P(X_3|A)}{P(X_3|A)P(A|X_2) + P(X_3|\sim A)P(\sim A|X_2)} P(A|X_2) = \frac{0.7}{0.7 \times 0.3311 + 0.1 \times (1 - 0.3311)} \times 0.3311 = 0.776 \\ P(A|X_4) = \frac{P(X_4|A)}{P(X_4|A)P(A|X_3) + P(X_4|\sim A)P(\sim A|X_3)} P(A|X_3) = \frac{0.7}{0.7 \times 0.776 + 0.1 \times (1 - 0.776)} \times 0.776 = 0.96 \\ P(A|X_5) = \frac{P(X_5|A)}{P(X_5|A)P(A|X_4) + P(X_5|\sim A)P(\sim A|X_4)} P(A|X_4) = \frac{0.7}{0.7 \times 0.96 + 0.1 \times (1 - 0.96)} \times 0.96 = 0.994 \end{array} \right. \quad (11)$$

According to Eq. (11), when five scan lines detect a potential obstacle at the same position independently, the probability of being a real obstacle reaches 99.4% according to this algorithm. Therefore, the proposed FFA algorithm can detect the obstacle by examining the probabilities in the global map.

Figure 16 illustrates the process to apply Bayesian rule in FFA algorithm in experiment. Figure 16(a) is a scene that a negative obstacle is placed in front of the vehicle. Figure 16(b1-f1) are the results detected by AMFA algorithm. Figure 16(b2-f2) show the corresponding map where weights are accumulated according to Bayesian rule. Figure 16(b1) and (b2) are the results generated from the left LiDAR at a single frame, whereas (c1) and (c2) are the results from the right LiDAR at a single frame. Figure 16(d1) and (d2) are the results generated from both two LiDARs at a single frame. Figure 16(e1) and (e2) are the results from two LiDARs by accumulating two series frames, and Figure 16(f1) and (f2) are the results generated from two LiDARs by accumulating five series frames. It can be found that the probability where there is a real obstacle has achieved almost 100% in Figure 16(f2). Thus, it is very efficient to detect the obstacle by setting a threshold for FFA algorithm.

## 7. EXPERIMENTAL RESULTS AND ANALYSIS

To evaluate the proposed algorithms, experiments are carried out both in structured environments and field environments. Our experiments are composed of four parts: first, details of the testing environments and the kinds of negative obstacles are introduced in Part I. The detection accuracy and the maximum detection distance of the proposed algorithm are analyzed in Part II. In Part III, the detection performance in different kinds of scenes are described, and some comparison with state-of-the-art algorithms are also discussed. In Part IV, some false alarm cases are discussed.

### 7.1. Part I: Description of the Testing Environments and Negative Obstacles

Two structured roads and six unstructured roads filled with negative obstacles are adopted to validate the proposed algorithms. Different types of negative obstacles are listed in Figure 17 (we prepared about 25 different negative

obstacles, labeled as  $O1$  to  $O25$ , most of which are dug by manual workers). The sizes of obstacles vary from  $0.5\text{ m}$  to  $3\text{ m}$ , and the depths are deeper than  $0.5\text{ m}$ . The two structured roads are labeled as  $S1$  and  $S2$ . Each of the two roads is dug with a pit. Six unstructured roads are categorized into five types: a grass lane, labeled as  $U1$ ; two dirt roads, labeled as  $U2$  and  $U3$ ; a highland road, in which the surface is seriously bumpy, labeled as  $U4$ ; a path in a forest, labeled as  $U5$ ; a race track in the “**Overcome Danger 2014**” **ground unmanned vehicle challenge of China**, labeled as  $U6$ . Details of obstacles distributions are listed in Table II. Most of the testing roads are built not only for verifying the proposed algorithm, but also for examining the whole process of autonomous driving, such as navigation, positive obstacle detection, path plan, and so forth.

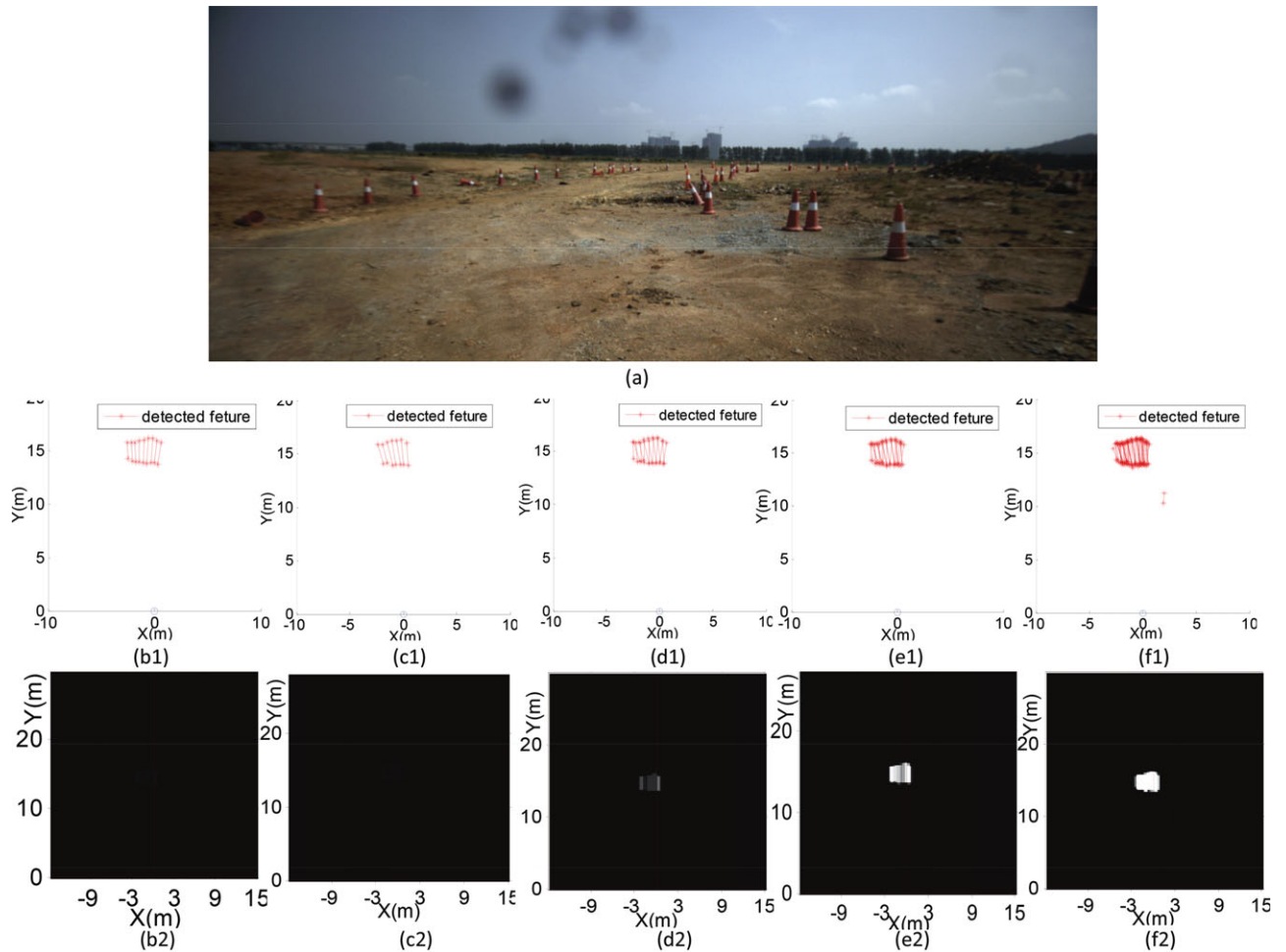
The right-handed coordinate is defined as follows:  $X$  goes to the right of the vehicle,  $Y$  measures what is in front, and  $Z$  comes out of the ground plane. The size of the obstacle is defined as  $X1 \times Y1$ , where  $X1$  means the length in  $X$  axis,  $Y1$  means the length in  $Y$  axis. Whether a ditch can be crossed safely mainly depends on the diameter of the wheel and the length of  $Y1$ . Thus,  $Y1$  is chosen to describe the size of a negative obstacle in this paper.

Most of the testing roads in the field environments are bumpy. Figure 18 shows two typical rough roads,  $U4$  and  $U5$ , which are drawn according to GPS information. Obstacles' position are also marked in the trajectory. In Figure 18,  $Z$  axis measures the bumpiness of the testing road. Most of the testing obstacles are placed at the most bumpy sites.

### 7.2. Part II: Analysis of the Detection Accuracy and the Maximal Detection Distance of the Proposed Algorithm

The interface of the experimental result is first introduced in this part, as shown in Figure 19. Figure 19(a) is the LiDAR





**Figure 16.** The process of applying Bayesian rule in the proposed FFA algorithm: (a) the scene where the experiment is carried out; (b1-f1) the features detected by the proposed AMFA algorithm; (b2-f2) corresponding map where weights are accumulated according to Bayesian rule; (b1) and (b2) the results generated by the left LiDAR at a single frame; (c1) and (c2) the results generated by the right LiDAR at a single frame; (d1) and (d2) the results generated by both LiDARs at a single frame; (e1) and (e2) the results generated by both two LiDARs at two accumulating series frames; (f1) and (f2) are the results generated by both LiDARs at five accumulating series frames.

view, in which points from the left LiDAR are labeled blue and points from the right LiDAR are labeled red. The detected negative obstacle is labeled by green points in figure (a). Figure 19(b) is the grid map under the vehicle coordinate frame, and each grid is  $0.2\text{ m} \times 0.2\text{ m}$ . The origin is placed at the cross of the two blue lines. Two green columns are  $2\text{ m}$  apart from the origin. The distance between every row of the green lines is  $5\text{ m}$  in the grid map. The detected obstacle is marked red in the map, as shown in Figure 19(b). This map is the final result of the negative obstacle detection module and will be sent to other modules of ALV in real time. Figure 19(c) shows the control platform of the proposed algorithm, and the frame number is shown on this platform.

Figure 19(d) shows the scene where the experiment is carried out and the location of the negative obstacle. According to the relationship between the image coordinate and the vehicle coordinate, the detection result is labeled by red lines in this scene image. A camera is used to present a visualized view of the negative obstacle for observers. The camera is not strictly synchronous with LiDARs. That is why the detection result labeled in the image does not fit the real position identically. Two LiDARs are also not strictly synchronous, which is not required in our proposed algorithms.

The details of the whole detection process can be illustrated in Figure 20 and in Figure 21. A negative obstacle (O7) is adopted to illustrate this process. The experiment is

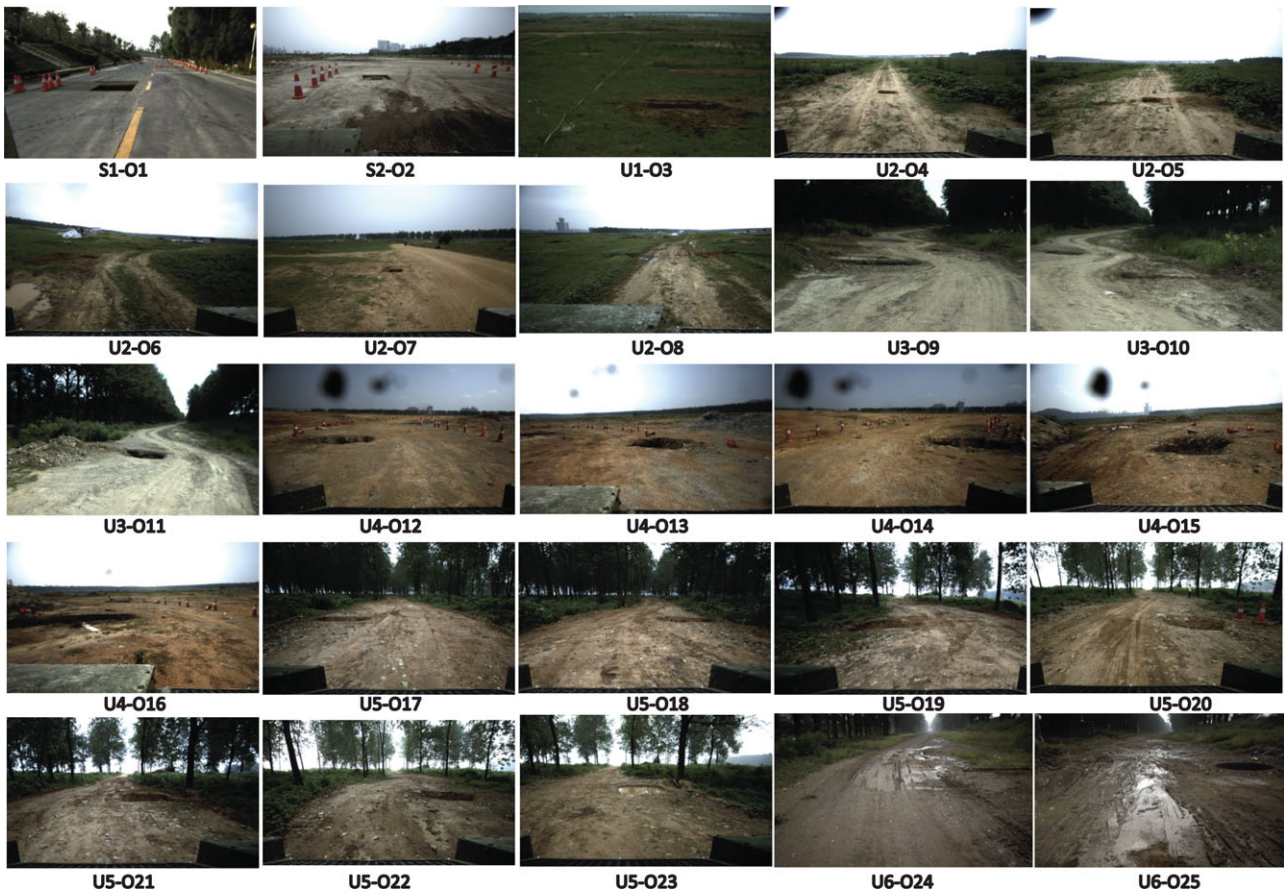


Figure 17. Various negative obstacles in both structured and unstructured environments.

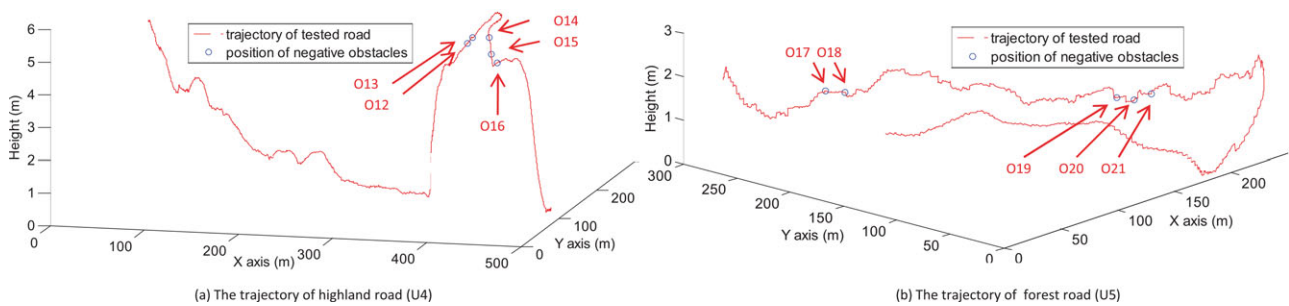


Figure 18. The track of two testing roads and the positions of the corresponding negative obstacles.

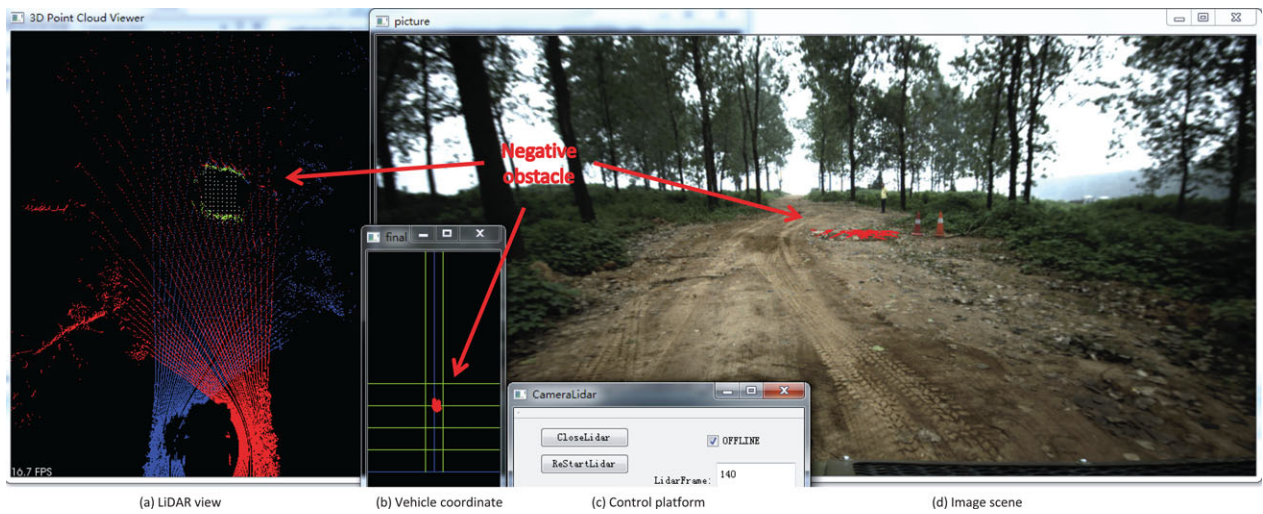
designed as follows: a vehicle drives toward the obstacle. A sequence of frames of detection results are listed in Figure 21, which describes the process of the vehicle approaching the target. We can see the target is detected more and more precise when the vehicle is approaching the target.

In the process of detection, the detection result will approximate the real obstacle when the vehicle approaches nearer, as illustrated in Figure 20. When the vehicle drives toward the obstacle, the detection result in the grid map

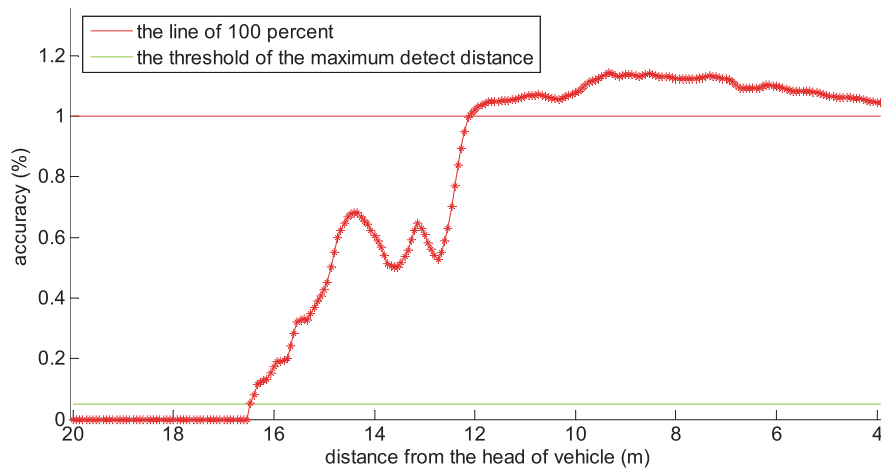
(Figure 19(b)) is recorded. Thus, the size of the detected obstacle is known every time during the whole process illustrated as in Figure 20. X axis in Figure 20 describes the distance from the vehicle to the target, and Y axis denotes the detection accuracy of the proposed algorithm. The accuracy is defined as follows:  $acc = \frac{A_{detected}}{A_{standard}} \times 100\%$ , where  $A_{detected}$  is the area taken by the detected target by the algorithm, and  $A_{standard}$  is the real size of the target obstacle. In Figure 20, the obstacle was first detected (a part

**Table II.** Detailed descriptions of the testing roads and negative obstacles.

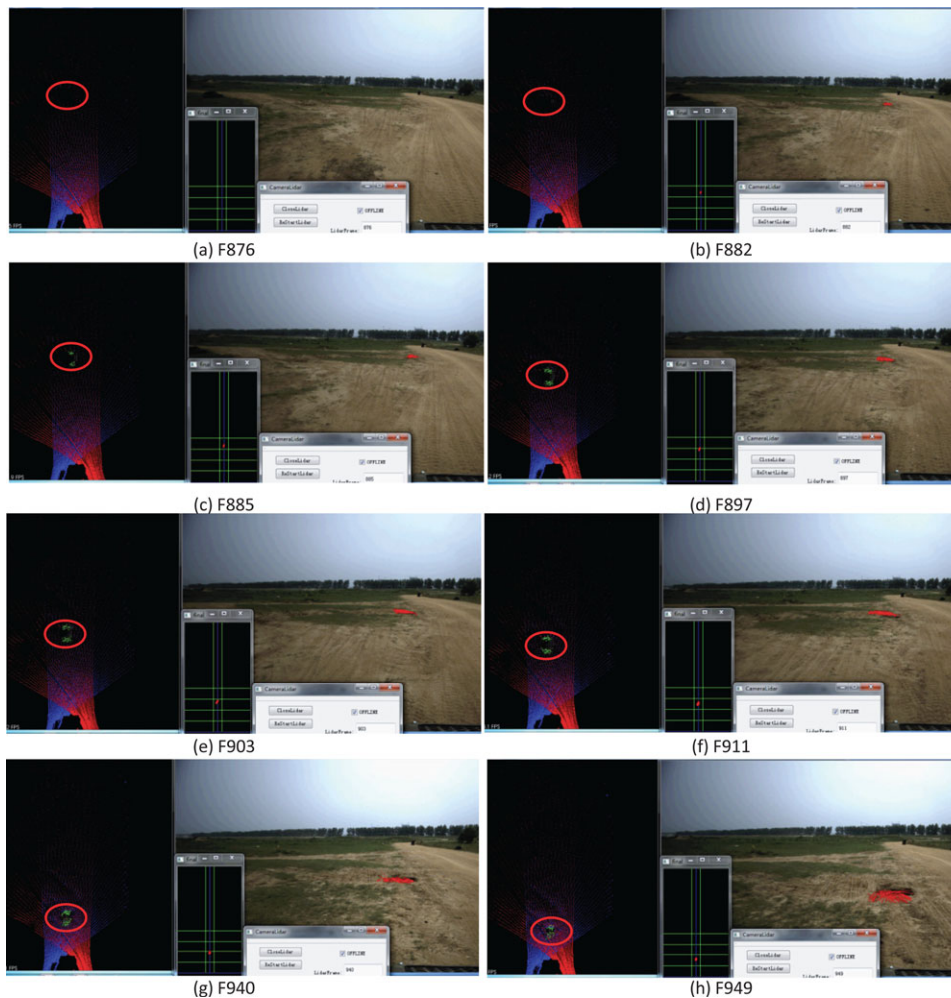
Label	Road type	Number	Size: $X1 \times Y1$ (m)	Length of road (km)
S1	Structured road	1	O1: $1.5 \times 1$	3
S2	Structured road	1	O2: $1 \times 0.7$	1
U1	Grass road	1	O3: $1 \times 1$	0.1
U2	Dirt road	5	O4,O5: $0.5 \times 0.5$ , O6,O7: $1 \times 0.7$ O8: $3 \times 0.7$ (not human dug)	5
U3	Dirt road	3	O9: $2 \times 2$ , O10: $3 \times 1$ O11: $2 \times 2$	3
U4	Highland road	5	O12: $2 \times 1.5$ , O13: $1.5 \times 1.5$ O14,15: $2 \times 2$ , O16: $3 \times 1.5$	5
U5	Forest road	7	O17: $1.5 \times 1$ , O18: $1.5 \times 1.5$ O19: $1.5 \times 2$ , O20: $R=2.5$ O21, O22: $2 \times 1.5$ , O23: $2 \times 1.5$ puddle	5
U6	Race track	2	O24: $3 \times 1$ , O25: $R=2.5$	3



**Figure 19.** The interface window of the experimental result: (a) shows the LiDAR data; (b) grid map in the vehicle coordinate frame; (c) the control platform; (d) the corresponding scene image.



**Figure 20.** Analyzing the detection accuracy when the vehicle is approaching a negative obstacle.



**Figure 21.** The detection process: a sequence frames of detection results when the vehicle approaches a negative obstacle.

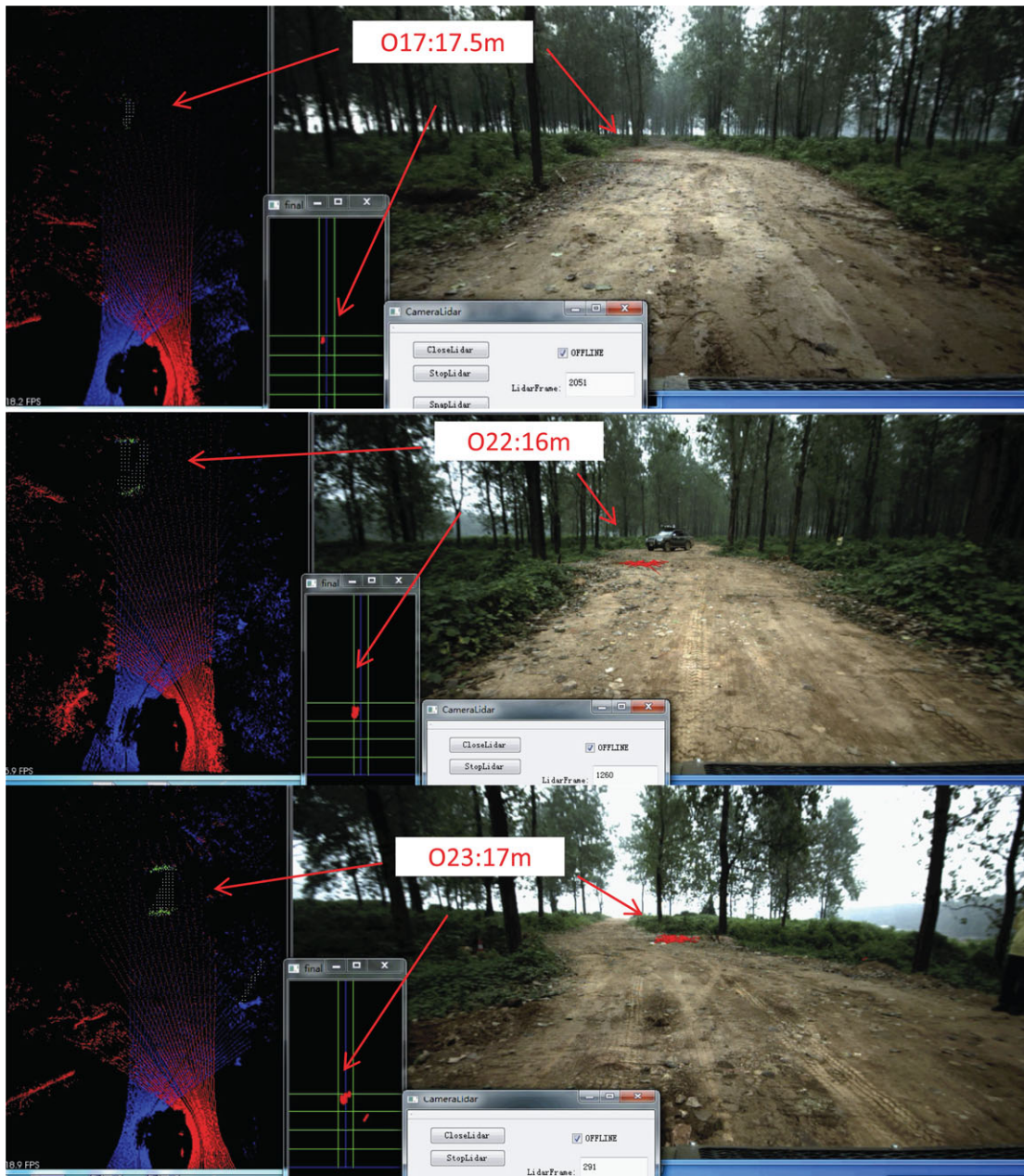
of the obstacle) when the obstacle was located at a distance of  $16.5\text{ m}$  away from the vehicle, which means this target emerged in the grid map. When the vehicle approaches nearer, the detected area became bigger and bigger. When the distance is less than  $12\text{ m}$  away, the detected area will become larger than the standard one ( $acc > 100\%$ ). There are several reasons for this outcome: (1) The GPS signal drifts even between adjacent frames, which is used to translate the position of the detected obstacle onto the global map. (2) There is a calibration error between the LiDAR coordinate and the vehicle coordinate.

In real applications, it is allowed to detect the potential obstacle a little larger than the real one, but it is forbidden to detect the potential obstacle smaller than the real one. If the detected obstacle is smaller than the real obstacle, the vehicle would encounter danger when crossing the obstacle. This aspect has been taken into consideration in the proposed algorithm.

The maximal detection range is used to evaluate the performance of the proposed algorithm, which is defined as follows: When the size of the detected target is bigger than a fixed threshold, the corresponding distance between the vehicle and the target is the maximal detection distance. A filter is operated before mapping the detection result onto the final grid map in real application, thus, the detection results of too small areas would be considered as noise and be filtered. Once an obstacle is emerged in the grid map, the corresponding distance between the vehicle and the obstacle is defined as the maximal detection distance in our experiments.

### 7.3. Part III: Adaptability Analysis and the Comparison

Various negative obstacles are employed in different experiments to verify the adaptability of the proposed algorithm



**Figure 22.** Typical experiment results of the proposed algorithm in different scenes (I).

in this part. Two vehicle platforms with LiDARs set at different heights ( $H = 1.4\text{ m}$  and  $H = 2\text{ m}$ ) are employed in the experiments. For each vehicle, two compact HDL-32 LiDARs are equipped on the two sides of the vehicle top, as shown in Figure 1. Some typical experimental results are listed in Figure 22, Figure 23, and Figure 24. In those experimental results, the detected obstacles are marked in the LiDAR data, the grid map, and the images. The distance between the vehicle and the obstacle at the current moment

is also marked beside the obstacle (they are not the maximal detection distance). Experimental results illustrate the high performance of the proposed algorithm. More details of experimental results for different negative obstacle detections are available at <http://pan.baidu.com/s/1hq2KcnU>, or at <http://yunpan.cn/cy5FLHpVgNPzI><sup>1</sup>.

<sup>1</sup>the acquired code is 9e0a

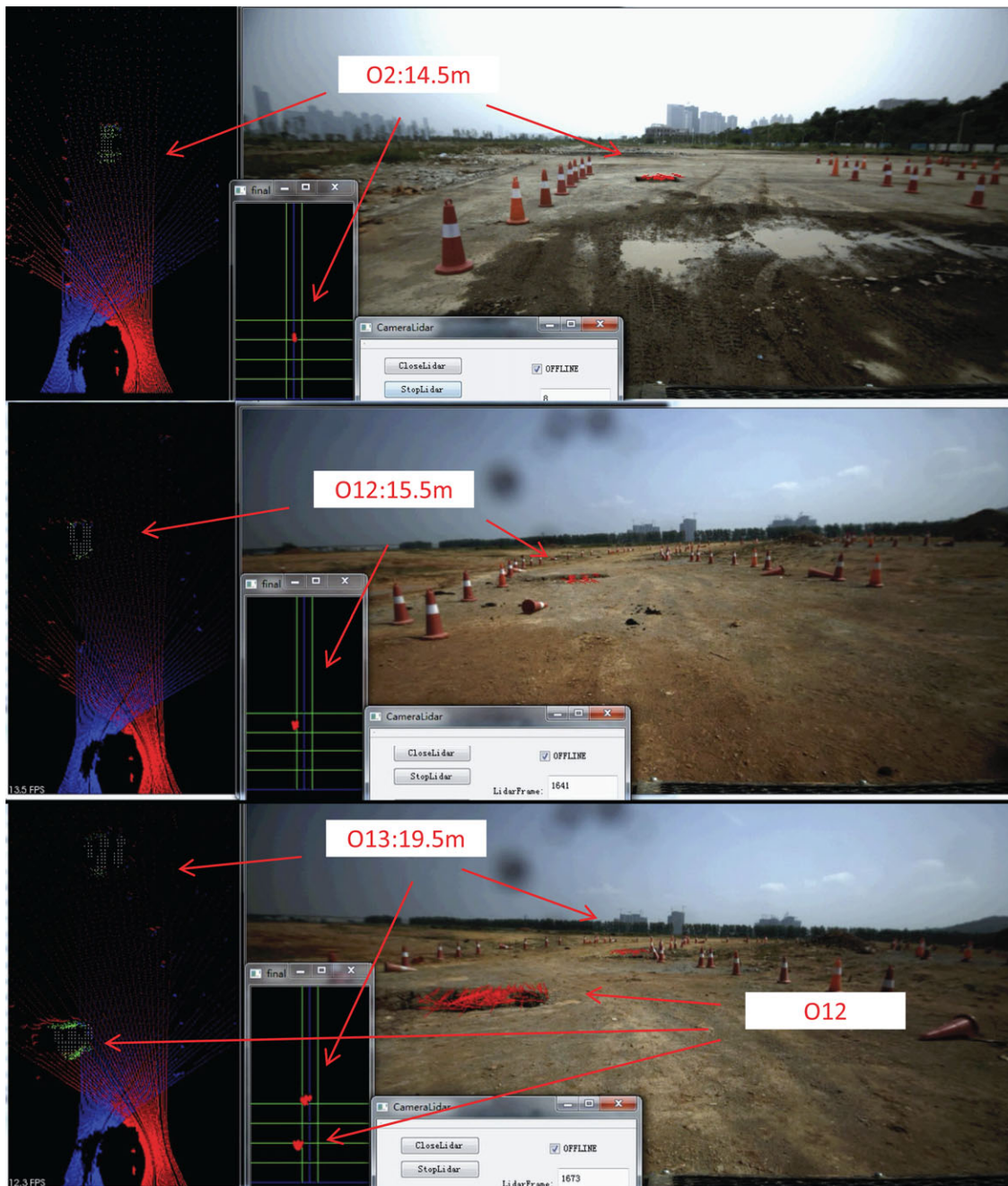
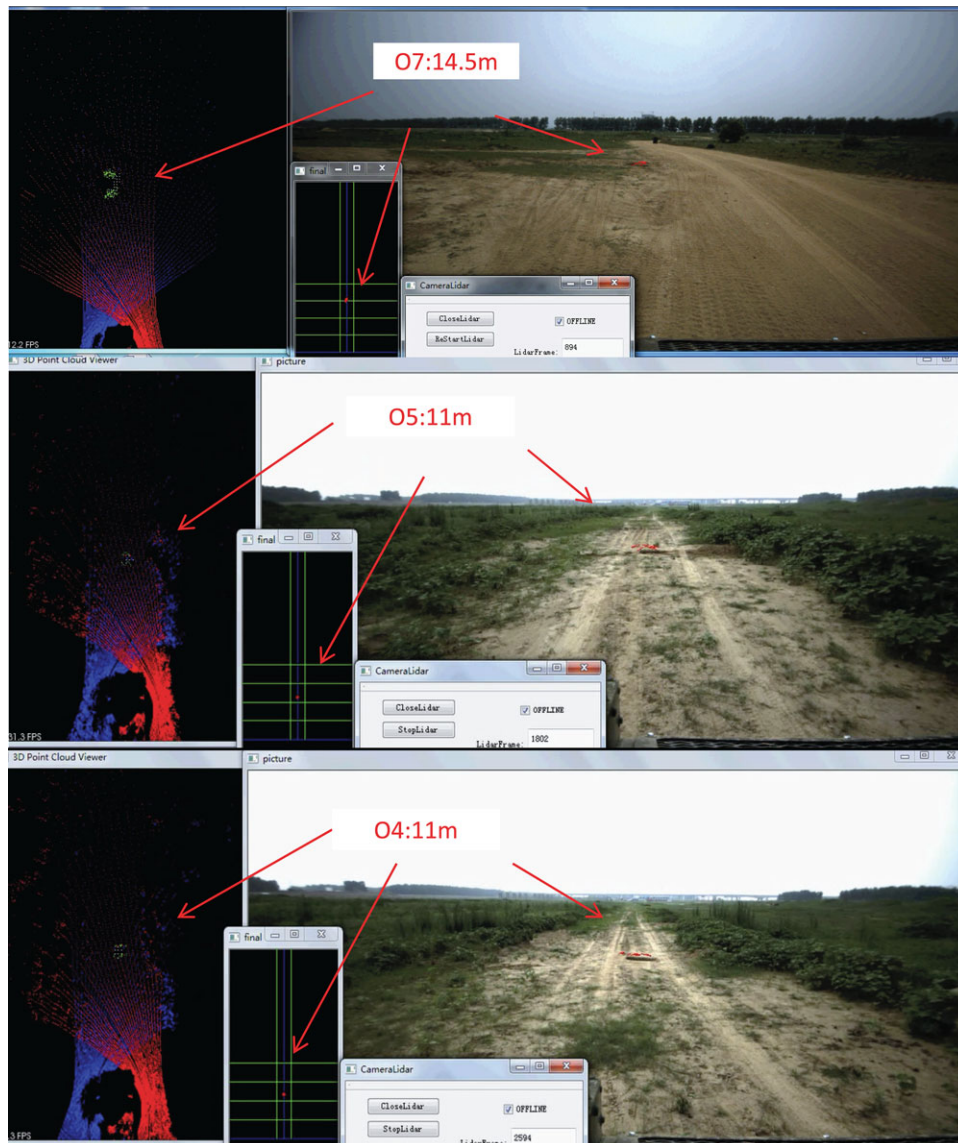


Figure 23. Typical experiment results of the proposed algorithm in different scenes (II).

Both AMFA algorithm and FFA algorithm are verified by detecting different kinds of negative obstacles. The maximal detection distance is employed to evaluate the capability of the proposed algorithms. The position of the potential obstacle is hand-selected. Therefore, the false alarms generated by the roadside or other positive obstacles would not disturb the maximal

detection distance. When the vehicle approaches the target in different directions, or at different speeds, or with a jarring vehicle, the maximal detection distance will vary. Thus, an average of 10 results is employed to measure the final maximal detection distance for a negative obstacle.

In addition, the maximal detection distance for different obstacles with a same size placed on different



**Figure 24.** Typical experiment results of the proposed algorithm in different scenes (III).

roads is also different. For example, the maximal detection distance is larger when the obstacle is placed in a structured environment than when it is placed in an unstructured environment. Thus, for a same-sized obstacle, the average of the maximal detection distances is used to measure the capability of the proposed algorithm. The maximal detection distances for all obstacles are listed in Table III. In Table III, all of the maximal detection distances are the average values by repetitious experiments.

Unfortunately, to the authors' best knowledge, there is no public database especially for negative obstacles, because the shape of a negative obstacle is different, and the

definition of a negative obstacle is hard to fix. In addition, the environment where the obstacle is placed also would affect the detection distance of an approach. Thus, a straightforward comparison between the proposed algorithm with other state-of-the-art approaches seems infeasible.

Tingbo Hu is the co-author of this paper, thus their previous works (Tingbo, 2012; Wu & He, 2011) are suitable for comparison. In their previous work, two cameras were used to generate a 3-D scene, and both the color information and the geometrical character were applied as cues to enlarge the detection distance. Some typical negative obstacles mentioned in Figure 17 had also been

**Table III.** The maximal detection distance for some typical negative obstacles by the proposed algorithm.

Unit (m)	Height of LiDAR	Width of obstacle <sup>a</sup>	Maximal detection distance
AMFA	H=1.4	L=0.5	D=9
AMFA	H=2	L=0.5	D=9
FFA	H=1.4	L=0.5	D=9
FFA	H=2	L=0.5	D=9.5
AMFA	H=1.4	L=0.7	D=10
AMFA	H=2	L=0.7	D=10.5
FFA	H=1.4	L=0.7	D=11
FFA	H=2	L=0.7	D=12
AMFA	H=1.4	L=1	D=12
AMFA	H=2	L=1	D=13.5
FFA	H=1.4	L=1	D=14
FFA	H=2	L=1	D=15
AMFA	H=1.4	L=1.5	D=13
AMFA	H=2	L=1.5	D=14
FFA	H=1.4	L=1.5	D=15
FFA	H=2	L=1.5	D=16
AMFA	H=1.4	L=2	D=15
AMFA	H=2	L=2	D=16
FFA	H=1.4	L=2	D=19
FFA	H=2	L=2	D=21
paper (Tingbo, 2012)	H=2	L=1	D=12

<sup>a</sup>L is the length in Y axis of the vehicle coordinates, which is used to describe the size of a negative obstacle.

detected by their approach. The maximal detect distance was also listed in Table III. Results show that if the sensor is equipped at a same height ( $H = 2\text{ m}$ ), and to detect the same obstacle ( $L = 1\text{ m}$ ), the maximal detection distance of the proposed FFA algorithm can be improved by 20%, compared to the approach in (Tingbo, 2012). With the same distance ( $D = 12\text{ m}$ ), the proposed FFA algorithm can detect much smaller ditches ( $L = 0.7\text{ m}$ ).

Larson et al. (Larson & Trivedi, 2011; Larson et al., 2011) used a HDL-64 LiDAR for negative obstacle detection as mentioned in the related work. In their paper, (Page 5) “multiple simulated scenes were created, filled with negative obstacles,” but the size information of negative obstacles are not described. In their work, the results were given as follows: the detection rate was 31% when the target was  $50\text{ m}$  away, and the detection rate was 89% when the target was  $16\text{ m} - 20\text{ m}$  away. In addition, Larson also pointed out that the real detection range of their system was between  $4.76\text{ m}$  and  $5.91\text{ m}$  for detecting a slope.

Another performance criterion of a detection algorithm is the computing time. The proposed FFA algorithm is based on the proposed AMFA algorithm, in which all computing operations are based on adjacent points. With an Inter core i7-2620M CPU and 4GB RAM computer, the computing time of the proposed FFA is less than  $10\text{ ms}$ .

The computing time in paper (Tingbo, 2012) with the same computer is more than  $100\text{ ms}$ . Thus, compared to the algorithm in (Tingbo, 2012), the computing time is reduced by an order of two magnitudes. The computing time is also mentioned in paper (Larson & Trivedi, 2011), in which the approach was at an average rate of  $2\text{ Hz}$ . A LiDAR was also used in paper (Heckman et al., 2007), in which the computing operation was carried out on a 3-D data accumulation map, running at a slow frame rate of  $1\text{ Hz}$ . Paper (Sinha & Papadakis, 2013) also pointed out the computing time of their approach, though the detection range was not public: in a 64-bit system, Intel-I7 CPU and  $7.8\text{ GB}$  memory, the average computing time costs  $15\text{ ms}$ . The computing time are listed in Table IV. To the best of the authors’ knowledge, the proposed algorithm is the fastest among the existing negative obstacle detection algorithms.

#### 7.4. Part IV: False Alarm Analysis

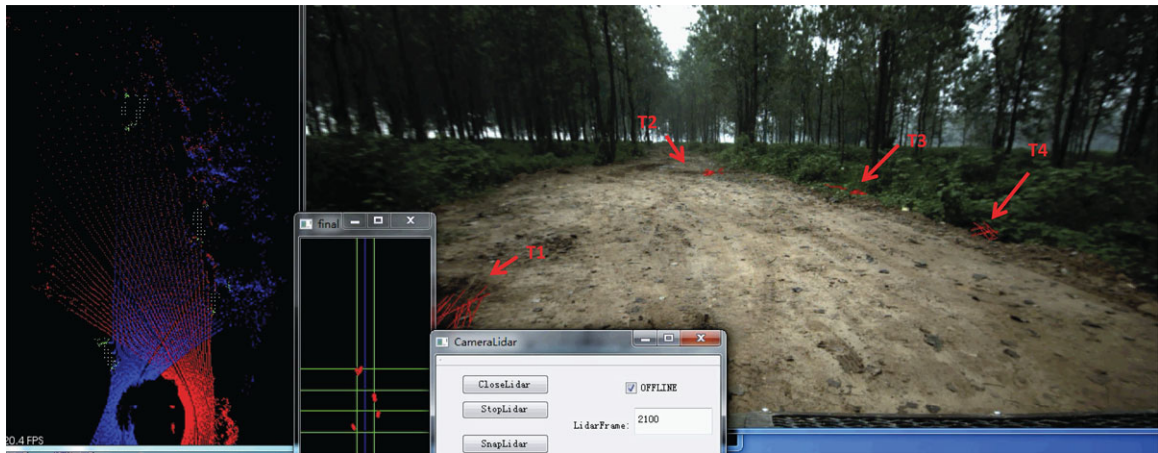
Another performance criterion of the negative obstacle detection algorithm is the false alarm rate. As difficult as it is to define a negative obstacle, it is also difficult to define a false alarm, and how to weigh it. On the one hand, the environment is an open aggregation, and every new scene or new road might bring a potential false alarm to the algorithm. On the other hand, it is even harder to determine whether it is a false alarm, because the negative obstacle itself is hard to define. For example, the sides of the road are usually lower than the road surface in an unstructured environment. In this case, it is hard to distinguish the false alarms from the real negative obstacles. Figure 25 shows a typical scene in which both real negative obstacles and pseudo-negative obstacles are detected by the proposed algorithm. In Figure 25, four targets have been detected, labeled as  $T1, T2, T3, T4$ . Among these four targets,  $T1$  and  $T2$  are real pits on the road surface;  $T3$  and  $T4$  are the roadsides, which are lower than the road surface. However,  $T3$  and  $T4$  would also bring serious risks to the vehicle if it drives on them. Therefore, it is hard to define that  $T3$  and  $T4$  are false alarms or real negative obstacles. Similar “false alarm” occurs when there are two positive obstacles toward the LiDAR. The area between two positive obstacles are lower than the positive obstacles, thus the algorithm would consider it as a ditch. A rough road can cause an occasional false alarm. More experimental results about false alarms generated by the proposed algorithm are offered at <http://pan.baidu.com/s/1hqtKcnU>, or at <http://yunpan.cn/cy5FLHpVgNPzI2>, on which some videos of experimental results are provided. In these videos, the negative obstacle detection is recorded, as well as the generated false alarms. Up to now, the proposed FFA algorithm has been successfully applied on more than four different field ALVs. These field ALVs have driven thousands of miles in both structured environments and field

<sup>2</sup>the acquired code is 9e0a



**Table IV.** Comparison of computing time with state-of-the-art algorithms.

	Compute target	Platform	Computing time (ms)
This paper	adjacent points by LiDAR	Inter core i7-2620M CPU, 4GB RAM	less than 10
Paper (Sinha & Papadakis, 2013)	3-D image data generated by LiDAR	64 bit system, Intel-I7 CPU, 7.8 GB RAM	15
Paper (Tingbo, 2012)	3-D image data by stereo camera	Inter core i7-2620M CPU, 4GB RAM	more than 100
Paper (Larson & Trivedi, 2011)	3-D LiDAR data	not mentioned	500
Paper (Heckman et al., 2007)	3-D LiDAR data	3 GHz CPU, 2 GB RAM	800

**Figure 25.** It is hard to distinguish false alarm from real potential negative obstacle.

environments under autonomous mode in last year. False alarms generated by the proposed algorithm didn't bring trouble to our ALVs when driving under the autonomous mode.

## 8. CONCLUSION

This paper introduces a LiDAR based negative obstacle detection algorithm by designing a novel setup of 3-D LiDAR for field ALVs. Different from the traditional upright setup, this proposed setup arranges two 3-D LiDARs on the two sides of the vehicle top to reduce the blind region around the vehicle and to improve the density of the scan lines. On the basis of this new setup, a mathematical model of the point distribution for the single scan line is deduced. Ideal scan lines are generated by the mathematical model to simulate the real one. With the mathematical model, an AMFA and an FFA are proposed. Lots of experiments are carried out, and experimental results reveal the effectiveness and high performance. Compared with the state-of-the-art techniques, the detection range is expanded by 20% and the computing time is reduced by an order of two magnitudes. The proposed FFA algorithm has been successfully applied

in two ALVs, which won the champion and the runner-up in the "Overcome Danger 2014" ground unmanned vehicle challenge of China, with a score of negative obstacle detection task of 14.4%.

In addition, it is a valuable choice to employ two HDL-32 LiDARs instead of a HDL-64 LiDAR for field ALVs. First, it costs almost the same for two HDL-32 LiDAR compared with a HDL-64 LiDAR. Second, the visual range generated by two HDL-32 LiDARs is more suitable for a field ALV, compared with a upright HDL-64 LiDAR. Third, the density in the ROI is much denser under the new setup, which is very useful for both positive obstacle detection and negative obstacle detection. Fourth, it is better to detect negative obstacles as discussed in this paper. The idea to deal with obstacle detection by adjacent points also can be applied to detect positive obstacles. The biggest advantage of this idea is real-time implementation, because all distributed processing on board the vehicle had to be real-time capable. Fusing features from different sensors or at different frames and iterating the probability by Bayesian rule are also the contributions of this paper. All in all, we hope our work would help to improve the ability of field ALVs.

**ACKNOWLEDGMENTS**

This work was supported in part by the National Natural Science Foundation of China under Grant 61473303, 91220301.

**REFERENCES**

- Braid, D., Broggi, A., & Schmiedel, G. (2006). The terramax autonomous vehicle. *Journal of Field Robotics*, 23(9), 693–708.
- Chen Tongtong, Dai Bin, W. R., & Daxue, L. (2013). Gaussian-process-based real-time ground segmentation for autonomous land vehicles. *Journal of Intelligent & Robotic Systems*, (pp. 1–20).
- Dima, C. S., Vandapel, N., & Hebert, M. (2004). Classifier fusion for outdoor obstacle detection. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 1, (pp. 665–671). IEEE.
- Fazli, S., Dehnavi, H. M., & Moallem, P. (2011). A robust negative obstacle detection method using seed-growing and dynamic programming for visually-impaired/blind persons. *Optical Review*, 18(6), 415–422.
- Geiger, A., Lenz, P., & Urtasun, R. (2012). Are we ready for autonomous driving? The KITTI Vision Benchmark Suite. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, (pp. 3354–3361). Rhode Island, USA. IEEE.
- Grant, W. S., Voorhies, R. C., & Itti, L. (2013). Finding planes in LiDAR point clouds for real-time registration. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, (pp. 4347–4354). IEEE.
- Han, J., Kim, D., Lee, M., & Sunwoo, M. (2012). Enhanced road boundary and obstacle detection using a downward-looking LIDAR sensor. *Vehicular Technology, IEEE Transactions on*, 61(3), 971–985.
- Häselich, M., Arends, M., Wojke, N., Neuhaus, F., & Paulus, D. (2012). Probabilistic terrain classification in unstructured environments. *Robotics and Autonomous Systems*, 61(10), 1051–1059.
- Heckman, N., Lalonde, J.-F., Vandapel, N., & Hebert, M. (2007). Potential negative obstacle detection by occlusion labeling. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, (pp. 2168–2173). San Diego, California, USA. IEEE.
- Kammel, S., Ziegler, J., Pitzer, B., Werling, M., Gindele, T., Jagzent, D., Schröder, J., Thuy, M., Goebel, M., Hundelshausen, F., et al. (2008). Team AnnieWAY's autonomous system for the 2007 DARPA urban challenge. *Journal of Field Robotics*, 25(9), 615–639.
- Larson, J., & Trivedi, M. (2011). Lidar based off-road negative obstacle detection and analysis. In *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, (pp. 192–197), Beijing, China. IEEE.
- Larson, J., Trivedi, M., & Bruch, M. (2011). Off-road terrain traversability analysis and hazard avoidance for ugv's. Technical report, DTIC Document, Department of Electrical Engineering, University of California San Diego, San Diego, CA.
- Manduchi, R., Castano, A., Talukder, A., & Matthies, L. (2005). Obstacle detection and terrain classification for autonomous off-road navigation. *Autonomous Robots*, 18(1), 81–102.
- Matthies, L., & Rankin, A. (2003). Negative obstacle detection by thermal signature. In *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 1, (pp. 906–913), Las Vegas, Nevada, USA. IEEE.
- Montemerlo, M., Becker, J., Bhat, S., Dahlkamp, H., Dolgov, D., Ettinger, S., Haehnel, D., Hilden, T., Hoffmann, G., Huhnke, B., et al. (2008). Junior: The Stanford entry in the urban challenge. *Journal of Field Robotics*, 25(9), 569–597.
- Montemerlo, M., & Thrun, S. (2006). Large-scale robotic 3-D mapping of urban structures. In *Experimental Robotics IX*, (pp. 141–150). Springer, verlag berlin heidelberg.
- Morton, R. D., & Olson, E. (2011). Positive and negative obstacle detection using the hld classifier. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, (pp. 1579–1584), San Francisco, CA, US. IEEE.
- Pedro Santana, Magno Guedes, L. C., & Barata, J. (2010). Stereo-based all-terrain obstacle detection using visual saliency. *Journal of Field Robotics*, 28(2), 241–263.
- Rankin, A. H., & Matthies, L. (2007). Nighttime negative obstacle detection for off-road autonomous navigation. In *Intelligent Robots and Systems (IRS), 2007 Proceedings of the International Conference on*, (pp. 906–913), Orlando, Florida, USA. SPIE DIGITAL LIBRARY. IEEE.
- Shang, E., An, X., Shi, M., Meng, D., Li, J., & Wu, T. (2014). An efficient calibration approach for arbitrary equipped 3-D LiDAR based on an orthogonal normal vector pair. *Journal of Intelligent & Robotic Systems*, Published online, 09 July 2014: (pp. 1–16).
- Shang, E., Li, X. A. J., Ye, L., & He, H. (2013). Robust unstructured road detection: The importance of contextual information. *International Journal of Advanced Robotic Systems*, 10(3), 1–8.
- Sinha, A., & Papadakis, P. (2013). Mind the gap: Detection and traversability analysis of terrain gaps using LIDAR for safe robot navigation. *Robotica*, 31(07), 1085–1101.
- Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., & Aron, A., Halpenny, M. (2006). Stanley: The robot that won the DARPA Grand Challenge. *Journal of Field Robotics*, 23(9), 661–692.
- Tingbo, H. (2012). Research on Stereo Vision Algorithms and Their Application in Obstacle Detection for ALV. PhD thesis, National University of Defense Technology.
- Urmson, C., Anhalt, J., Bagnell, D. & etc (2008). Autonomous driving in urban environments: Boss and the urban challenge. *Journal of Field Robotics*, 25(8), 425–466.
- Velodyne LiDAR, I. (2012). HDL-32E User's Manual. [www.velodynelidar.com](http://www.velodynelidar.com).

Wu, T. H. Y. N. T., & He, H. (2011). Negative obstacle detection from image sequences. In *Digital Image Processing (ICDIP), 2011 Third International Conference on*, (pp. 797–802), Chengdu, China. IEEE.

Zhou, S., Xi, J., McDaniel, M. W., Nishihata, T., Salesses, P., & Iagnemma, K. (2012). Self-supervised learning to visually detect terrain surfaces for autonomous robots operating in forested terrain. *Journal of Field Robotics*, 29(2), 277–297.