electronics

MDPI

Article

Autonomous Path Finding and Obstacle Avoidance Method for Unmanned Construction Machinery

Jiangdong Wu [1,2], Haoling Ren [1,2], Tianliang Lin [1,2,*], Yu Yao [3], Zhen Fang [1,2] and Chang Liu [1,2]

[1] College of Mechanical Engineering and Automation, Huaqiao University, Xiamen 361021, China; 1711121029@stu.hqu.edu.cn (J.W.); rhl@hqu.edu.cn (H.R.)

[2] Fujian Key Laboratory of Green Intelligent Drive and Transmission for Mobile Machinery, Xiamen 361021, China

[3] School of Mechanical Engineering and Automation, Beihang University, Beijing 102206, China; 19013080044@stu.hqu.edu.cn

* Correspondence: ltl@hqu.edu.cn

Abstract: The working environment of construction machinery is harsh, and some operations are highly repetitive. The realization of intelligent construction machinery helps to improve economic efficiency and promote industrial development. Construction machinery is different from ordinary passenger vehicles. Aiming at the fact that the existing environmental perception data set cannot be directly applied to construction machinery, this paper establishes the corresponding data set in combination with the specific working conditions of construction machinery and carries out training based on the PointPillars network to realize the environmental perception function applicable to the working conditions of construction machinery. Most construction machinery runs on unstructured roads, and the existing passenger vehicle path planning algorithm is not applicable to construction machinery. Based on this, this paper uses a hybrid A* algorithm to achieve path planning that meets the kinematics of construction machinery and realizes real-time obstacle detection and avoidance. At the same time, this paper combines environmental perception with a path planning algorithm to provide a method of autonomous path finding and obstacle avoidance for construction machinery. Based on the improved pure pursuit algorithm, the high-precision motion control and established trajectory tracking of construction machinery are realized, which lays a certain foundation for the follow-up research and development of related intelligent technologies of construction machinery.

Keywords: unmanned driving; construction machinery; PointPillars; hybrid A*; improve pure pursuit; autonomous routing

check for updates

Citation: Wu, J.; Ren, H.; Lin, T.; Yao, Y.; Fang, Z.; Liu, C. Autonomous Path Finding and Obstacle Avoidance Method for Unmanned Construction Machinery. *Electronics* **2023**, *12*, 1998. https://doi.org/10.3390/electronics12091998

Academic Editor: Ahmed F. Zobaa

Received: 21 March 2023
Revised: 19 April 2023
Accepted: 20 April 2023
Published: 25 April 2023

1. Introduction

Construction machinery is widely used in the field of engineering and construction, with the characteristics of a poor working environment and highly repeatable operation. With the wave of global intelligence, construction machinery has also begun to develop in the direction of intelligence. Intelligent construction machinery includes intelligent products and construction, intelligent manufacturing, intelligent service and intelligent management. The unmanned driving of construction machinery belongs to the category of intelligent products and construction [1], which refers to the autonomous completion of walking and operation of construction machinery without human intervention.

The driverless technology of passenger vehicles generally includes four modules: environment perception, vehicle localization, planning decision-making and motion control. On this basis, construction machinery also has an automatic operation module. Due to the particularity of the walking and operation of construction machinery, the existing driverless technology of passenger vehicles cannot be directly transferred to construction machinery. For the environmental perception module, most of the existing data sets are collected for the driving environment of passenger vehicles, and cannot meet the requirements of the environmental perception module for the walking and operation of construction machinery. Therefore, it is necessary to make corresponding data sets according to the working conditions of construction machinery. At present, the environment perception in the field of unmanned driving is mainly divided into two categories, one is image environment perception based on camera, the other is point cloud environment perception based on lidar. Among them, image environment perception has the advantages of rich texture information and easy-to-extract feature information, which is more suitable for object classification, but not conducive to the extraction of object depth in large scenes. In contrast, point cloud environmental perception is more suitable for the extraction of object depth in large scenes, and can effectively provide the position and attitude information of work objects and obstacles for construction machinery, and provide data support for planning decision-making, motion control and automatic operation of construction machinery.

At present, point cloud target detection can be roughly divided into four categories: target detection based on fusion image and point cloud, target detection based on 3D space voxelization, target detection based on direct point cloud processing and target detection based on 3D space voxelization and direct point cloud processing. The network framework used in this paper is PointPillars [2], which belongs to the target detection category based on 3D space voxelization. Its advantage is that it can fully use the convolution layer to extract point cloud features, and can meet the real-time requirements of point cloud target detection.

The driving environment of construction machinery is mostly on unstructured roads, so it is difficult to extract road features, and it is impossible to use vector maps commonly used in driverless vehicles to constrain their driving path and rules. Aiming at the above problems, this paper proposes a path planning algorithm for construction machinery based on hybrid A* [3], which generates a cost map in real time through the observation point cloud mapping, so as to complete the path planning and obstacle avoidance of the whole construction machinery.

The point cloud environment perception algorithm in this paper can effectively detect the position and attitude of the work object relative to the whole vehicle, and transfer the position and attitude of the work object as the target point into the path planning module. Through the hybrid A* algorithm, a work path that conforms to the kinematics of the construction machinery is generated, and the construction machinery is controlled to move to the vicinity of the work object through the corresponding motion control algorithm. Because the kinematics model of construction machinery is different from that of ordinary vehicles, the existing motion control algorithms of driverless vehicles cannot be directly applied to the motion control of construction machinery. Based on this, this paper proposes an improved pure pursuit algorithm for motion control of construction machinery based on the traditional pure pursuit algorithm [4] and the kinematics characteristics of construction machinery.

Based on the above algorithms, the whole vehicle simulation platform and the real vehicle test platform of tracked construction machinery are built to verify the above algorithms. The simulation and real vehicle test results show that the environment perception algorithm can effectively detect the working objects, and the path planning algorithm based on hybrid A* and the improved pure pursuit motion control algorithm proposed in this paper can effectively complete the autonomous path finding, autonomous obstacle avoidance and corresponding motion control of construction machinery, which lays a certain foundation for the subsequent research of unmanned construction machinery.

The main contributions of this paper are as follows: (1) A point cloud data set that conforms to the working conditions of construction machinery is produced, and it is trained and verified by using the PointPillars network. (2) Based on the hybrid A* algorithm and the kinematic characteristics of construction machinery, a method suitable for construction machinery path planning is proposed, and real-time obstacle avoidance of construction machinery is realized through a real-time grid map. (3) Improve the existing pure tracking algorithm to make it suitable for crawler construction machinery motion control. (4) Build corresponding construction machinery simulation and real vehicle test platforms to verify and analyze the above algorithms.

This paper is organized as follows: Section 2 introduces the production of construction machinery working condition data set and the construction machinery environment perception algorithm based on PointPillar. Section 3 introduces the path planning algorithm for construction machinery based on hybrid A*. Section 4 introduces the motion control of construction machinery based on the improved pure pursuit algorithm. Sections 5 and 6 introduce the construction of the unmanned construction machinery simulation platform and the real vehicle test platform, and the analysis of related simulation and real vehicle test results. Finally, Section 7 provides an outlook for the conclusions, shortcomings and future work of this paper.

2. An Algorithm for Environmental Perception of Construction Machinery Based on PointPillars

The basic principle of target detection based on 3D space voxelization is to voxelize the 3D laser point cloud, and then extract the features of the voxel lattice through a convolution neural network. At present, the classic algorithms include VoxelNet [5], SECOND [6], Pointpillars and MODet [7]. The main principle of the VoxelNet algorithm is to voxelize the 3D point cloud, then extract the features of the point cloud in each voxel, and finally, extract the voxel features into local features using the voxel feature coding network and process the local features using the 3D convolution neural network, so as to obtain the deep semantic features and their corresponding geometric space representation. The disadvantage of this algorithm is that the point cloud is sparse, and the voxels of the extracted features are empty, which will greatly reduce the computational efficiency and affect the real-time performance of the entire environment perception system. The SECOND algorithm adds a sparse convolution layer on the basis of VoxelNet, which solves the problem of point cloud sparsity to a certain extent, and improves the reasoning speed of the algorithm. However, because the VoxelNet voxel-by-voxel feature method is still used, its deployment in the actual system still cannot meet the real-time requirements. As a common point cloud sensing algorithm in the industry, PointPillars uses the reduced-dimension convolution method to improve the reasoning speed. The feature is that only two-dimensional convolution can achieve end-to-end 3D point cloud learning and target detection.

2.1. Production of Point Cloud Data Set for Working Conditions of Construction Machinery

In this paper, an open field on campus is selected as the data collection site, and the objects such as mounds, roadblocks, carts, wheel excavators, loaders, trucks, pedestrians and so on that are common in engineering construction are taken as the collection objects, as shown in Figure 1. After the data set is collected, it is divided into 3500 laser point cloud frames, and the objects in each point cloud frame are labeled, respectively.

This paper uses the point cloud annotation software SUSTechPoints [8] developed by the South University of Science and Technology. The software interface is shown in Figure 2. Its advantages are that it can edit the annotation box with nine degrees of freedom, and it has good interactivity and has a relatively friendly annotation detail check mechanism. At present, SUSTechPoints has been widely used in point cloud target detection and target tracking data set production.

In this paper, SUSTechPoints is used to label the objects in the 3500 laser point cloud frames collected and divided above. This data set was created according to the standards of the Kitti data set. A total of 7153 mounds, 7314 water horses, 5234 carts, 2245 wheel excavators, 2371 loaders, 1945 trucks and 6845 pedestrians are labeled. Among them, the mound anchor has a width, length, and height of (1.88, 2.50, 1.00) m with a z center of -1.15 m. The water horse anchor has a width, length, and height of (0.34, 1.42, 0.75) m with a z center of -1.275 m. The cart anchor has a width, length, and height of (1.41, 0.77, 0.86) m with a z center of -1.22 m. The construction machinery anchor has a width, length, and height of (6.13, 2.47, 2.99) m with a z center of -0.15 m. Finally, the pedestrian anchor has a width, length, and height of (0.72, 0.71, 1.83) m with a z center of −0.735 m. Each object box labeled in this paper strictly conforms to the relevant point cloud labeling standards, and is attached with the object ID number to facilitate the deployment and application of the subsequent target tracking algorithm. The labeling effect is shown in Figure 3.

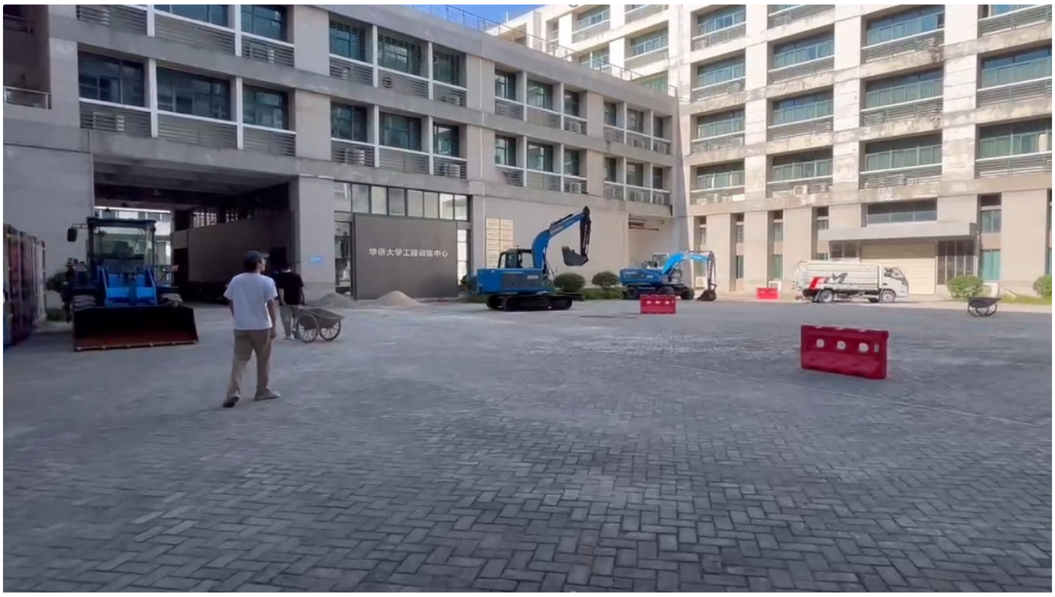


Figure 1. Data set collection scenario diagram.

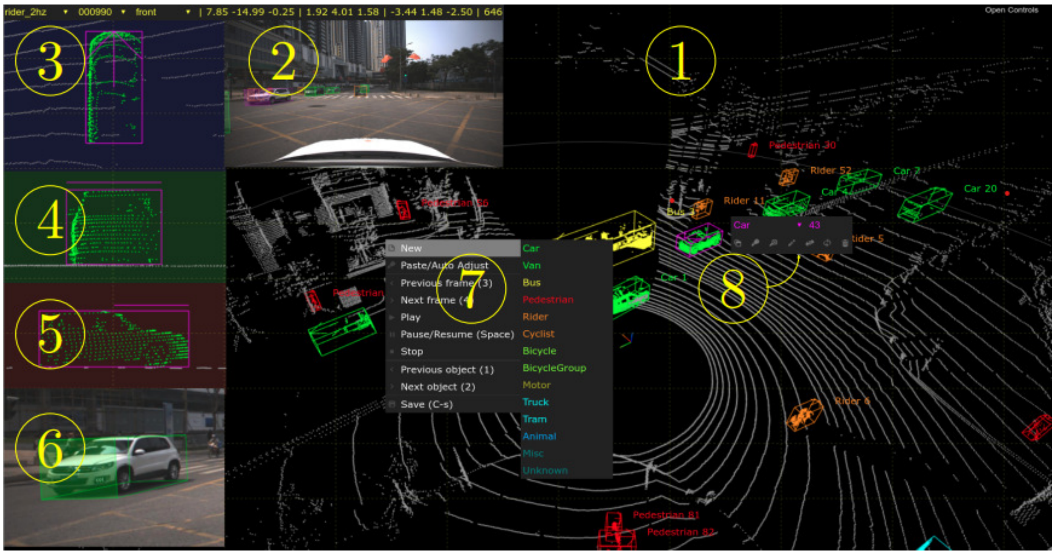


Figure 2. The main UI of SUSTechPoints. ① the perspective view of the 3D point cloud; ② the photo context, which is resizable and can be automatically switched among multiple camera images; ③ the top view of the selected object; ④ the front view of the selected object; ⑤ the side view of the selected object; ⑥ the focused photo context, which can be automatically chosen for the selected object; ⑦ the context menu, which provides tools of context operations; ⑧ the floating fast toolbox, which provides most used tools.

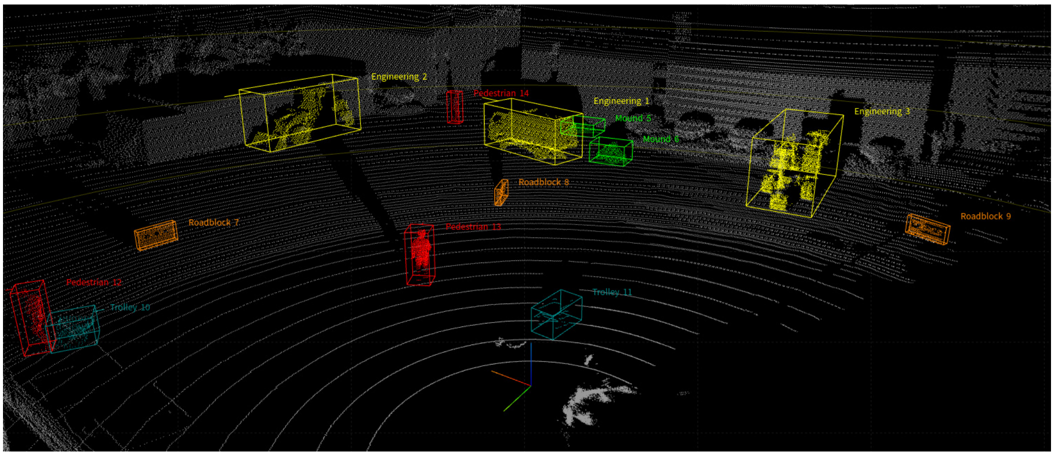


Figure 3. Rendering of point cloud annotation.

2.2. *Principle of PointPillars Algorithm*

In this paper, PointPillars is used to train and verify the data set of construction machinery working conditions collected and labeled. The network schematic diagram is shown in Figure 4, which is mainly divided into three main stages. The first stage uses pillar coding to encode the original point cloud, and the main point is to rasterize on the XY plane to determine that the laser points falling into the same grid belong to the same pillar. Suppose there are P non-empty pillars in a frame point cloud, each pillar contains N laser points, and each laser point is represented by a nine-dimensional vector D, as shown in Equation (1), so far the frame point cloud can be represented by a tensor (D, P, N).

$$D = (x, y, z, r, x_c, y_c, z_c, x_p, y_p) \tag{1}$$

$$x_p = x - x_c \tag{2}$$

$$y_p = y - y_c \tag{3}$$

where $(x,\ y,\ z)$ is the coordinate of the laser point data in the lidar coordinate system, $(x_c,\ y_c,\ z_c)$ is the geometric center coordinates of all laser points in the pillar where the laser point is located, and $(x_p,\ y_p)$ is the distance between the laser point and the geometric center.

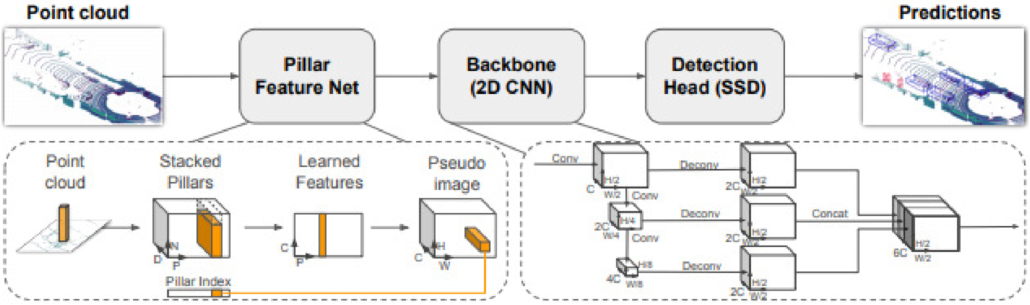


Figure 4. PointPillars network diagram.

Perform feature extraction operations on tensors (D, P, N) to obtain the feature tensors (C, P, N) after reducing the dimension. Finally, the maximum pooling of the feature tensor is carried out to obtain a planar feature map of the (C, P) dimension. P is divided into two dimensions, H and W, that is, the height and width of the pseudo-image, so as to complete the feature extraction operation of the original point cloud and successfully convert the three-dimensional original point cloud into a two-dimensional pseudo-image.

The second stage is to use the backbone network to extract the features of the pseudo-image. The backbone network consists of two networks. One network is used to continuously reduce the resolution of the feature map and increase its dimension, and the other network is used to upsample the feature map generated by the previous network, generate the feature map with the same resolution, and fully connect it to generate a fully connected layer. The third stage is mainly to use the Single Shot Detector (SSD) [9] to detect 3D objects in the fully connected layer, and the inspection head is modified accordingly to make it suitable for the detection of labeled objects in the data set of this paper.

2.3. Pointpillars Loss Function

The loss function is the most important part of the convolution neural network. It is the basis of the parameter iteration of the whole neural network. By solving the gradient of the loss function relative to the parameters of the neural network, the network model can be iterated and updated in the correct direction, thus effectively improving the object detection accuracy of the network model. The loss function adopted by PointPillars is similar to the SECOND algorithm, which represents the Box of each detected object with a seven-dimensional vector E, as shown in Equation (4).

$$E = (x, y, z, w, h, l, \theta) \tag{4}$$

where $(x,\ y,\ z)$ represents the center coordinate of each box, $(w,\ h,\ l)$ represents the length, width and height of each box, and θ represents the azimuth of each box.

Calculate the offset between the corresponding value of the seven-dimensional vector E of each detected object's Box and the real value, as shown in Equations (5)–(12):

$$\Delta x = \frac{x^{gt} - x^a}{d^a} \tag{5}$$

$$\Delta y = \frac{y^{gt} - y^a}{d^a} \tag{6}$$

$$\Delta z = \frac{z^{gt} - z^a}{h^a} \tag{7}$$

$$\Delta w = \log \frac{w^{gt}}{w^a} \tag{8}$$

$$\Delta l = \log \frac{l^{gt}}{l^a} \tag{9}$$

$$\Delta h = \log \frac{h^{gt}}{h^a} \tag{10}$$

$$\Delta \theta = \sin(\theta^{gt} - \theta^a) \tag{11}$$

$$d^a = \sqrt{(w^a)^2 + (l^a)^2} \tag{12}$$

where $(x^{gt}, y^{gt}, z^{gt}, w^{gt}, h^{gt}, l^{gt}, \theta^{gt})$ is the true value of each object Box, $(x^a, y^a, z^a, w^a, h^a, l^a, \theta^a)$ is the predicted value of each object Box, and $(\Delta x, \Delta y, \Delta z, \Delta w, \Delta l, \Delta h, \Delta \theta)$ is the offset between the true value and the predicted value of each object Box.

Based on the offset of each object Box and the Smooth L1 loss function, the location loss function of all detected object boxes in the input point cloud is constructed, as shown in Equation (13):

$$\mathcal{L}_{loc} = \sum_{b \in (x,y,z,w,l,h,\theta)} SmoothL1(\Delta b) \tag{13}$$

Similar to the SECOND algorithm, in order to avoid the wrong direction of object detection, the Softmax loss function is introduced to learn the direction of the object, which is recorded as $\mathcal{L}_{dir}$.

The Focal Loss function [10] is used as the function for object category detection, as shown in Equation (14):

$$\mathcal{L}_{cls} = -\alpha_a (1 - p^a)^{\gamma} \log p^a \tag{14}$$

To sum up, the total loss function of PointPillars is:

$$\mathcal{L} = \frac{1}{N_{pos}} (\beta_{loc}\mathcal{L}_{loc} + \beta_{cls}\mathcal{L}_{cls} + \beta_{dir}\mathcal{L}_{dir}) \tag{15}$$

3. Path Planning of Construction Machinery Based on Hybrid A*

Most construction machinery drives on unstructured roads without clear road characteristics, which makes the existing path planning algorithm of driverless vehicles not well compatible with construction machinery. Therefore, it is necessary to select an appropriate algorithm for path planning based on the driving characteristics and kinematics constraints of construction machinery. The existing path planning algorithms are mainly divided into three categories, namely, traditional path planning algorithms, sample-based path planning algorithms and intelligent bionic path planning algorithms. Among them, the representative works of traditional path planning algorithms include Dijkstra algorithm [11], A* algorithm [12], D* algorithm [13], and artificial potential field method [14], which have the advantage of simple algorithm and easy deployment, but the disadvantage is that it is only suitable for simple planning problems in small scenarios, and it is easy to fall into local optimization in large and complex scene planning, resulting in an endless loop.

The classic sample-based path planning algorithms include the PRM algorithm [15], RRT algorithm [16], and RRT variant algorithm [17–19]. Their advantages are high search efficiency, suitable for path planning in high-dimensional space and complex constraints. The disadvantages are that the planned path is not the optimal path, and the planned path curvature is discontinuous, which cannot be directly used. Moreover, the sample-based path planning algorithm does not support real-time path planning in dynamic scenes.

In recent years, with the continuous popularization of intelligent concepts and the continuous development of deep learning, reinforcement learning and other related disciplines [20,21], intelligent bionic path planning algorithms are also more and more widely used, which are mainly represented by the particle swarm optimization algorithm [22], ant colony algorithm [23], genetic algorithm [24] and path planning algorithm based on deep learning [25] and reinforcement learning [26]. Its advantages are that it conforms to the ecological characteristics of nature, it has strong trial and error ability and can be continuously optimized and updated through learning. Its disadvantages are that the algorithm needs continuous learning and optimization through data set training, and its generalization is poor at present.

In addition, there are many existing Unmanned Aerial Vehicle (UAV) obstacle avoidance algorithms. Cruz et al. [27] proposed a potential field obstacle avoidance algorithm based on fluid mechanics panel methods, which can complete the establishment of the environmental map by relying only on the on-board sensor of UAV, and has good obstacle avoidance performance. Aguilar et al. [28] proposed an obstacle avoidance method based on visual detection, which integrates visual detection and proportional control algorithms to achieve complete obstacle avoidance functions by using only monocular cameras. Hermand et al. [29] propose a constraint control scheme based on the ERG framework, which can complete obstacle avoidance in bounded space with a small amount of computation.

Considering that the driving environment of construction machinery is complex and changeable and the deployment of intelligent bionic path planning algorithm still needs the accumulation of relevant data sets of construction machinery, this paper uses the traditional path planning algorithm to plan and generate the driving path of construction machinery.

By comparing the classical algorithms of traditional path planning, it can be concluded that the Dijkstra algorithm and A* algorithm are algorithms that simplify the whole vehicle to a particle for path planning, without considering the geometric and kinematic constraints of the whole vehicle, so they cannot be directly applied to the path planning of construction machinery. The D* algorithm has high search efficiency and can be used in dynamic environments, but it is too large to be suitable for real-time planning of changeable scenes. The artificial potential field method uses the idea of a virtual force field to design the surrounding environment of the robot as an abstract artificial gravity field. The target point generates "gravity" on the mobile robot, and the obstacles generate "repulsive force" on the mobile robot. The advantage of the artificial potential field method is that the planned path is smooth and safe, and has certain robustness, but it easily falls into local optimization, and the gravity and repulsion are relative, which is not suitable for path planning in the case of too far away from the target point or too close to the obstacle. UAVs are different from construction machinery, and their obstacle avoidance methods cannot be directly applied to construction machinery obstacle avoidance.

To sum up, the hybrid A* algorithm is used to plan the driving path of construction machinery. Its advantages are that the kinematics constraints of construction machinery are considered, and the planned trajectory curvature is continuous and smooth, which is suitable for tracking and control of construction machinery. In addition, the hybrid A* algorithm combined with the real-time grid map can update the travel route of construction machinery according to the obstacle information in the grid map at all times, so as to realize the automatic obstacle avoidance function of construction machinery.

3.1. Hybrid A* Heuristic Function

Hybrid A* uses two heuristic functions, one is the non-holonomic constraint heuristic cost without obstacles, and the other is the holonomic heuristic cost with obstacles. Among them, the characteristic of non-holonomic constraint heuristic cost without obstacles is that the obstacle is not included in the calculation scope, only the vehicle kinematics constraint is considered, and the minimum turning radius of the vehicle is taken as the input quantity. The process is to determine the starting point (x_i, y_i, θ_i) and target point (x_g, y_g, θ_g). The Reeds–Shepp curve is used to plan an optimal curve between the starting point and the target point, and the distance of the optimal curve is calculated as the heuristic cost $h_1(n)$. The heuristic is mainly to trim the branches of the A* search tree properly to prevent the path direction to the target point from being inconsistent with the set direction.

In contrast, the holonomic heuristic cost with obstacles only considers the obstacle information and ignores the impact of vehicle kinematics constraints. At each node, the Dijkstra algorithm is used to obtain the nearest distance to the target point as the heuristic cost $h_2(n)$. The purpose of this action is to use the Dijkstra algorithm to obtain the characteristics of the shortest path from the expansion node to the target point in the maze environment, to prevent the wrong direction of Hybrid A* expansion during the expansion process, resulting in unnecessary computational waste.

To sum up, two heuristic costs $h_1(n)$ and $h_2(n)$ can be obtained by hybrid A*. In the actual path search process, the larger generation value is taken as the heuristic cost of hybrid A*, that is $h(n) = \max(h_1(n), h_2(n))$.

3.2. Voronoi Map

In the path planning algorithm of the whole vehicle, the map is an indispensable part. It provides the necessary prior information for path planning so that it can plan a global path that conforms to the vehicle kinematics characteristics and maintains a certain safe distance from the obstacles. Common maps mainly include grid maps, distance maps and Voronoi maps. The grid map is generally composed of grids of the same size, and each grid is assigned a value. According to each grid value, determine whether it is occupied, that is, whether there are obstacles, as the basis for global path planning. The distance map is to provide the nearest distance between the expansion node and the obstacle in the search process of the planning algorithm. The nearest distance is calculated by the Euclidean distance. The Voronoi map is an extension of the distance map, which is composed of a distance map and a generalized Tyson polygon.

The Voronoi map is also called the Tyson polygon or Dirichlet map [30]. As shown in Figure 5, it is a continuous polygon composed of a set of vertical bisectors connected by adjacent points. In the Voronoi map, each polygon has a generator, and the distance from any point in each polygon to the generator of the polygon is shorter than the distance from other generators. The distance from the point on the polygon boundary to the generator that generates the boundary is equal.

In the hybrid A* algorithm, the potential energy value $\rho_v(x,y)$ between the expansion node and the obstacle is calculated using the Voronoi map, as shown in Equation (16), and it is used as part of the generation value of the heuristic function so that a certain safe distance is always maintained between the expansion node and the obstacle.

$$\rho_v(x,y) = \begin{cases} (\frac{\alpha}{\alpha + d_o(x,y)})(\frac{d_v(x,y)}{d_o(x,y) + d_v(x,y)})(\frac{d_o - d_{\max}}{d_{\max}})^2 \; d_o \leq d_o^{\max} \\ 0 \; other \end{cases} \tag{16}$$

where $\rho_v(x,y)$ is the potential energy at node (x,y), the value is from 0 to 1, when $d_o > d_o^{\max}$, the potential energy of (x,y) is 0, and the maximum potential energy value is reached when (x,y) is on or inside the obstacle. The minimum potential energy value is reached when (x,y) is on the edge of the generalized Tyson polygon. d_o is the distance at which the node reached the nearest obstacle. d_v is the closest distance from the node to the edge of the Voronoi map. α represents the rate of potential energy decrease, which is a constant, $\alpha \in [0, +\infty)$. As the value of α increases, the potential energy at (x,y) decreases more slowly. $d_o^{\max}$ is the control range of potential energy and is constant.

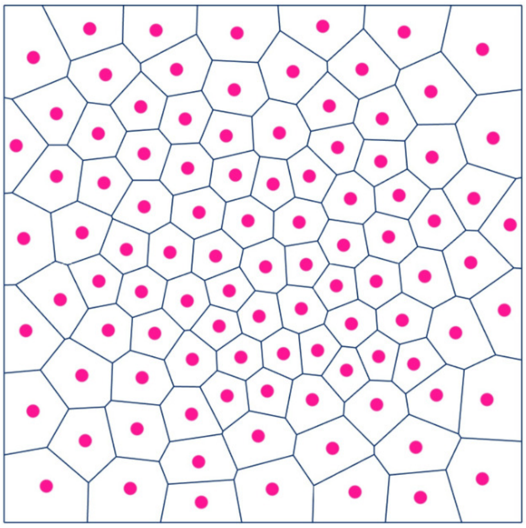
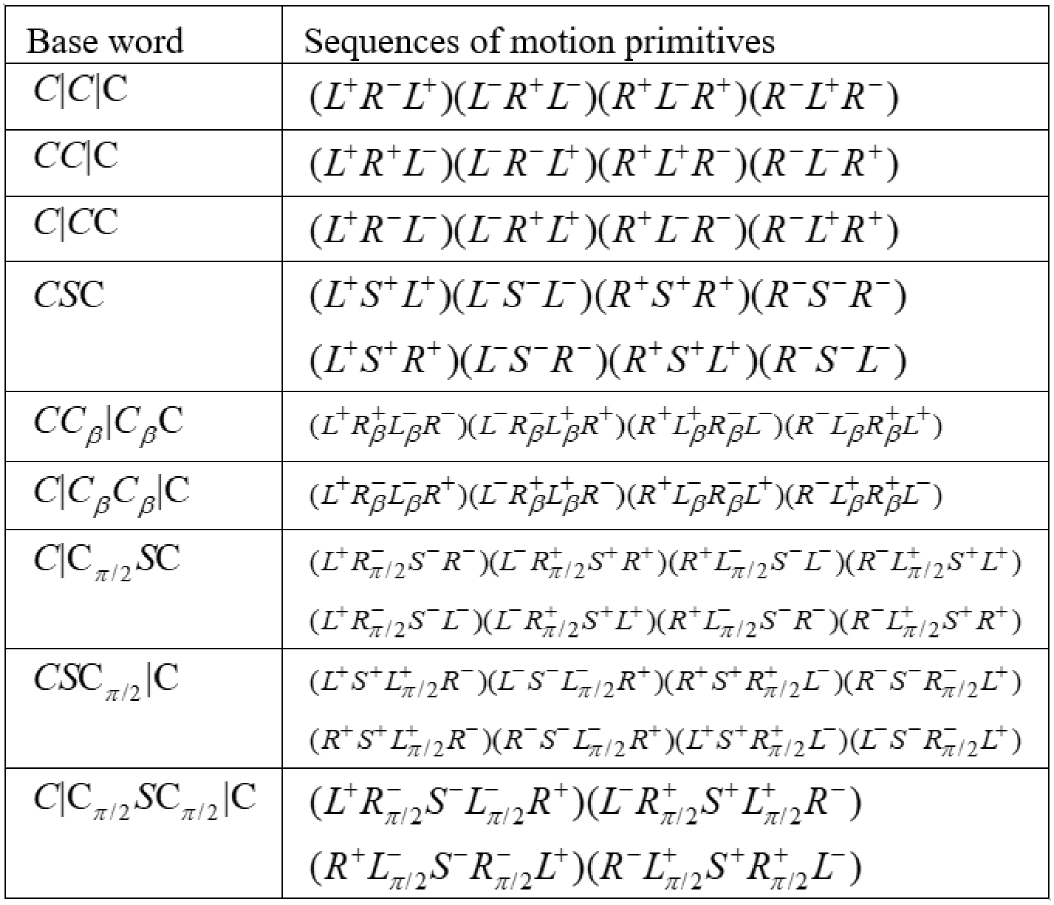


Figure 5. Voronoi diagram.

3.3. Reeds–Shepp Curve

The Reeds–Shepp curve was proposed by Reeds and Shepp in 1990 [31], which solved the problem that the Dubins curve could not generate a reverse path. The Reed–Shepp curve is represented by 9 base words or 48 motion primitives, as shown in Figure 6. The Reed–Shepp curve is mainly composed of straight lines, forward curves and backward curves. Wherein, S represents the straight line part, C represents the curve of turning left or right, C_u represents the curve part of two curves with equal length, $C_{\pi/2}$ represents the curve part with an angle of 90°, | represents the gear conversion (from front driving to back driving or from back driving to forward driving).



Base word	Sequences of motion primitives
$C\|C\|C$	$(L^+R^-L^+)(L^-R^+L^-)(R^+L^-R^+)(R^-L^+R^-)$
$CC\|C$	$(L^+R^+L^-)(L^-R^-L^+)(R^+L^+R^-)(R^-L^-R^+)$
$C\|CC$	$(L^+R^-L^-)(L^-R^+L^+)(R^+L^-R^-)(R^-L^+R^+)$
CSC	$(L^+S^+L^+)(L^-S^-L^-)(R^+S^+R^+)(R^-S^-R^-)$ $(L^+S^+R^+)(L^-S^-R^-)(R^+S^+L^+)(R^-S^-L^-)$
$CC_{\beta}\|C_{\beta}C$	$(L^+R^+_{\beta}L^-_{\beta}R^-)(L^-R^-_{\beta}L^+_{\beta}R^+)(R^+L^+_{\beta}R^-_{\beta}L^-)(R^-L^-_{\beta}R^+_{\beta}L^+)$
$C\|C_{\beta}C_{\beta}\|C$	$(L^+R^-_{\beta}L^-_{\beta}R^+)(L^-R^+_{\beta}L^+_{\beta}R^-)(R^+L^-_{\beta}R^-_{\beta}L^+)(R^-L^+_{\beta}R^+_{\beta}L^-)$
$C\|C_{\pi/2}SC$	$(L^+R^-_{\pi/2}S^-R^-)(L^-R^+_{\pi/2}S^+R^+)(R^+L^-_{\pi/2}S^-L^-)(R^-L^+_{\pi/2}S^+L^+)$ $(L^+R^-_{\pi/2}S^-L^-)(L^-R^+_{\pi/2}S^+L^+)(R^+L^-_{\pi/2}S^-R^-)(R^-L^+_{\pi/2}S^+R^+)$
$CSC_{\pi/2}\|C$	$(L^+S^+L^+_{\pi/2}R^-)(L^-S^-L^-_{\pi/2}R^+)(R^+S^+R^+_{\pi/2}L^-)(R^-S^-R^-_{\pi/2}L^+)$ $(R^+S^+L^+_{\pi/2}R^-)(R^-S^-L^-_{\pi/2}R^+)(L^+S^+R^+_{\pi/2}L^-)(L^-S^-R^-_{\pi/2}L^+)$
$C\|C_{\pi/2}SC_{\pi/2}\|C$	$(L^+R^-_{\pi/2}S^-L^-_{\pi/2}R^+)(L^-R^+_{\pi/2}S^+L^+_{\pi/2}R^-)$ $(R^+L^-_{\pi/2}S^-R^-_{\pi/2}L^+)(R^-L^+_{\pi/2}S^+R^+_{\pi/2}L^-)$

Figure 6. Word combination of Reeds–Shepp curves.

In the process of hybrid A* algorithm using Reeds–Shepp curve to solve the optimal path, given the initial point attitude (x_i, y_i, θ_i) and target point attitude (x_g, y_g, θ_g), 48 equations can be solved to obtain a shortest distance motion primitive. Finally, the corresponding waypoints of the global path can be calculated through the optimal motion primitive solution. The solution process is shown in Figure 7.

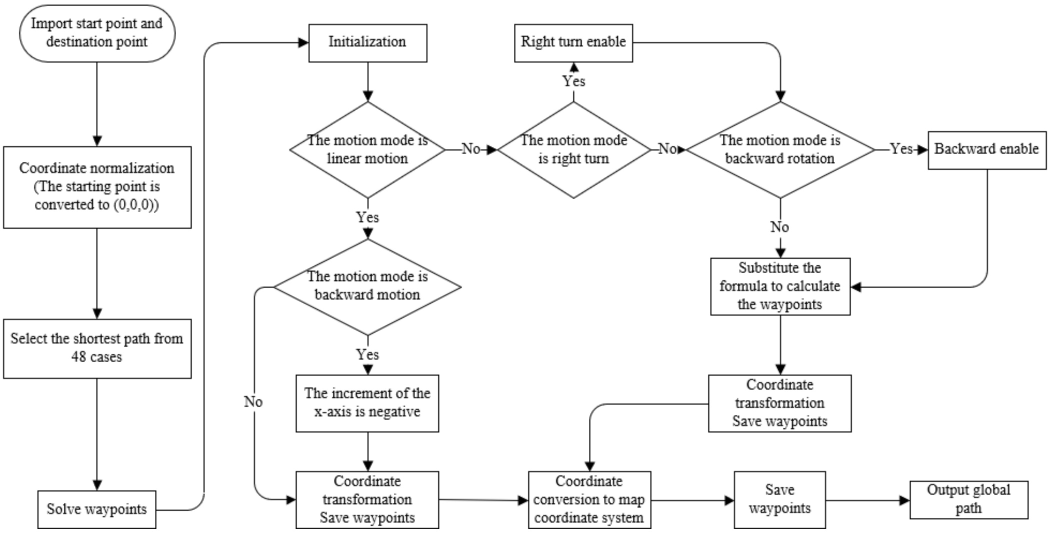

Import start point and destination point
Coordinate normalization (The starting point is converted to (0,0,0))
Select the shortest path from 48 cases
Solve waypoints
Initialization
The motion mode is linear motion
No
Yes
The motion mode is backward motion
Yes
No
The increment of the x-axis is negative
Coordinate transformation Save waypoints
Right turn enable
Yes
The motion mode is right turn
No
The motion mode is backward rotation
Yes
Backward enable
No
Substitute the formula to calculate the waypoints
Coordinate transformation Save waypoints
Coordinate conversion to map coordinate system
Save waypoints
Output global path

Figure 7. Flow chart of global path solution.

4. Motion Control of Construction Machinery Based on Improved Pure Pursuit

In the field of driverless vehicles, common motion control algorithms include the pure pursuit algorithm, Stanley algorithm [32], linear quadratic regulator (LQR), model predictive control (MPC), and intelligent control algorithm [33]. Among them, the pure pursuit and Stanley algorithms are algorithms developed based on vehicle kinematics, which are only applicable to the motion control of low-speed driverless vehicles. LQR and MPC take vehicle dynamics into account, and minimize the vehicle motion control error and control amount by solving the state equation to achieve optimal control, which is applicable to the motion control of high-speed driverless vehicles. Intelligent control algorithms use methods such as deep learning and reinforcement learning to control the motion of the entire vehicle, and typical algorithms include end-to-end algorithms [25]. Intelligent control algorithms are currently not particularly mature, as they require training and validation of a large number of driving data sets through convolutional neural networks. Considering that the commonly used construction machinery belongs to low-speed mobile machinery and the construction of accurate dynamic equations of construction machinery is more complex, and the pure pursuit algorithm is simpler than the Stanley algorithm in parameter adjustment, and the tracking performance of the two is similar. Therefore, this paper adopts the pure pursuit algorithm as the motion control algorithm for construction machinery.

4.1. Traditional Pure Pursuit Algorithm Based on Ackerman Steering

The traditional pure pursuit algorithm is designed based on the Ackerman steering kinematics model. Its basic principle is that the vehicle kinematics model is simplified to a bicycle model, and a reference path composed of several waypoints is given, and the preview distance is set. Search for the waypoint closest to the preview distance from the vehicle position in the reference path, and set it as the preview point. Build an arc between the vehicle and the preview point as the real-time tracking path of the vehicle. By changing the front wheel angle, the vehicle always moves along or tends to move along the real-time tracking path. With the continuous movement of the vehicle, the preview point of the vehicle changes constantly, and the vehicle tracking path continues to move forward until it reaches the end of the reference path.

Figure 8 is the schematic diagram of the traditional pure pursuit algorithm, in which, R is the arc radius of the vehicle tracking path, α is the included angle between the vehicle body and the preview point, δ is the front wheel angle of the vehicle, e is the lateral error between the vehicle and the preview point, L is the vehicle wheelbase, l_d is the preview distance, and (x_r, y_r) is the coordinate of the preview point.

It can be seen from Figure 8 that the front wheel angle δ of the vehicle can be obtained by solving the geometric relationship. First, under the condition that the included angle α between the vehicle body and the preview point and the preview distance l_d are known, the arc radius R of the tracking path and the curvature radius k of the tracking path can be solved by the sine theorem. The derivation process is shown in Equations (17) and (18).

$$\frac{l_d}{\sin(2\alpha)} = \frac{R}{\sin(\frac{\pi}{2} - \alpha)} \tag{17}$$

$$k = \frac{1}{R} = \frac{2\sin\alpha}{l_d} \tag{18}$$

After calculating the radius of curvature of the tracking path, the front wheel angle of the vehicle can be further calculated by using the geometric relationship. The specific derivation process is shown in Equations (19)–(21).

$$\tan(\delta) = \frac{L}{R} \tag{19}$$

$$\sin\alpha = \frac{e}{l_d} \tag{20}$$

$$\delta = \arctan(\frac{L}{R}) = \arctan(kL) = \arctan(\frac{2L\sin(\alpha)}{l_d}) = \arctan(\frac{2Le}{l_d^2}) \tag{21}$$

After the front wheel angle δ is solved, the rear wheel center of the whole vehicle can always follow or approach the tracking path by controlling the front wheel angle of the whole vehicle, and then complete the tracking control of the whole vehicle to the given reference path. So far, the theoretical derivation of the traditional pure pursuit algorithm has been completed.

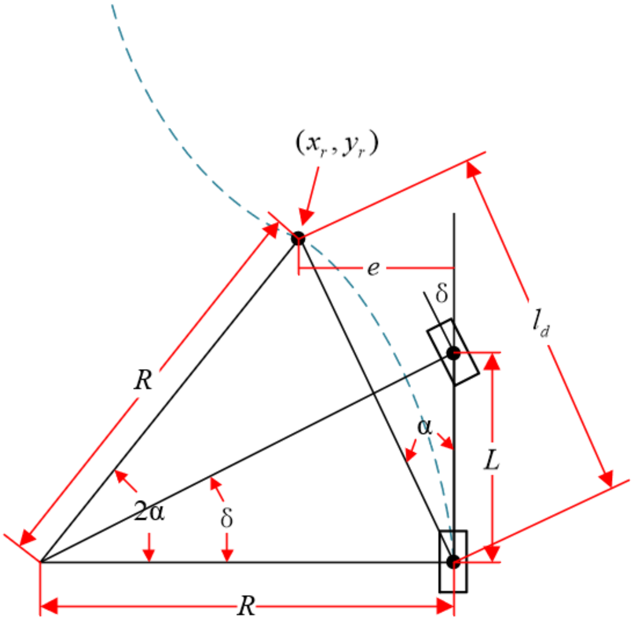


Figure 8. Principle diagram of traditional pure pursuit algorithm.

4.2. Improved Pure Pursuit Algorithm Based on Kinematic Model of Tracked Construction Machinery

The kinematics model of construction machinery is different from that of ordinary vehicles. The unmanned driving simulation and real vehicle test platform of construction machinery built in this paper are based on tracked construction machinery as the motion control carrier. Therefore, the relevant analysis will be carried out around the kinematic model of tracked construction machinery. For the improvement of other types of construction machinery motion control algorithms, we can refer to the research ideas of this paper and make appropriate adjustments based on the kinematic characteristics of various construction machinery, that is, the improved pure pursuit algorithm proposed in this paper can be applied to the motion control of various construction machinery.

Since the tracked construction machinery adopts the differential steering motion model, and the turning and forward are highly coupled, that is, the straight and steering of the whole vehicle are completed by controlling the driving speed of the left and right tracks, the traditional pure pursuit algorithm cannot be directly applied to the motion control of the tracked construction machinery, and it needs to be improved accordingly.

Figure 9 is the schematic diagram of the improved pure pursuit algorithm proposed in this paper for the kinematics model characteristics of tracked construction machinery. Wherein, R is the arc radius of the tracking path of tracked construction machinery, α is the included angle between the vehicle body and the preview point, e is the lateral error between the vehicle and the preview point, l_d is the preview distance, b is the half value of the left and right track spacing of tracked construction machinery, and (x_p, y_p) is the coordinate of the preview point.

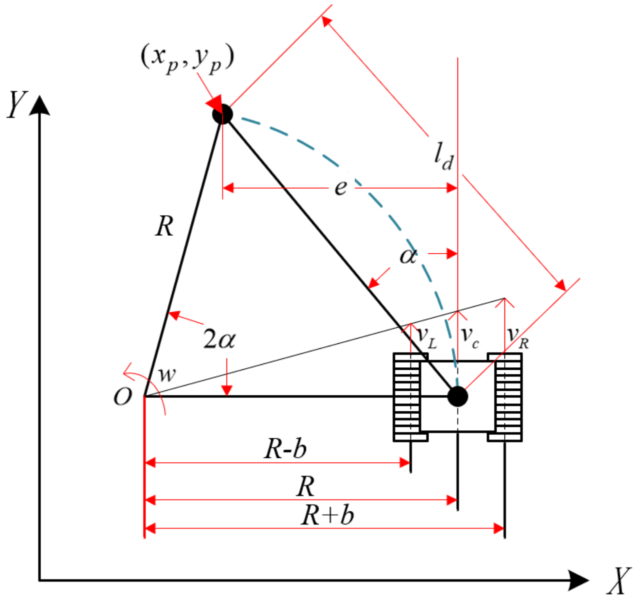


Figure 9. Principle diagram of improved pure pursuit algorithm.

Similar to the traditional pure pursuit algorithm, the improved pure pursuit algorithm proposed in this paper is also given a reference path composed of multiple waypoints, sets the preview distance, finds the waypoint closest to the preview distance from the vehicle position in the reference path and uses it as the preview point. It builds an arc with radius R between the preview point and the whole vehicle as the real-time tracking path of the whole vehicle, and updates the preview point and real-time tracking path as the whole vehicle moves forward until the whole vehicle reaches the destination.

Unlike the traditional pure pursuit algorithm, the traditional pure pursuit algorithm makes the whole vehicle approach or tend to follow the tracking path by controlling the front wheel angle. The improved pure pursuit algorithm in this paper is to control the speed of the left and right tracks to make the tracked construction machinery always follow the given track.

The improved pure pursuit algorithm proposed in this paper is theoretically derived. Since the derivation of the arc radius R and curvature radius k of the tracking path are the same as that of the traditional pure pursuit algorithm, this paper will not repeat the derivation, and the specific derivation is shown in Equations (17) and (18). After calculating the arc radius R and curvature radius k of the tracking path, calculate the angular velocity w of the whole vehicle around the center of the tracking path according to the given tracking speed v_c of the whole vehicle, and then calculate the tracking speed and of the left and right tracks, respectively, through the angular velocity w. The specific derivation process is shown in Equations (22)–(24). When the tracking speed of the left and right tracks is the same, the whole vehicle will go straight; when the left track speed v_L is less than the right track speed v_R, the vehicle turns left; when the left track speed v_L is greater than the right track speed v_R, the vehicle turns right. Through the precise control of the tracking speed of the left and right tracks, the motion control of the tracked construction machinery is realized so that its tracking track and reference path can be well fitted.

$$w = \frac{v_c}{R} = kv_c = \frac{2ev_c}{l_d^2} \tag{22}$$

$$v_L = w(R-b) = \frac{2ev_c(R-b)}{l_d^2} \tag{23}$$

$$v_R = w(R+b) = \frac{2ev_c(R+b)}{l_d^2} \tag{24}$$

5. Construction and Test of Simulation Platform Based on Gazebo

In order to verify the feasibility of applying hybrid A* to automatic routing of construction machinery, a construction machinery simulation platform is built based on Gazebo. Its advantage is that Gazebo integrates a large number of sensors needed for the research and development of driverless-related technologies, which greatly reduces project research and development costs. In addition, Gazebo is equipped with a relatively complete physical engine, which supports the simulation and verification of various driverless algorithms and eliminates the potential risk of the algorithm being directly deployed to the real vehicle for verification. Moreover, Gazebo supports the independent construction of various simulation environments, which can be effectively applied to the verification and simulation of relevant algorithms of unmanned construction machinery under extreme conditions. Finally, Gazebo supports multi-model collaborative simulation, which lays a foundation for the subsequent research of unmanned construction machinery cluster and cooperative operation related algorithms.

As shown in Figure 10, excavators commonly used in engineering operations are modeled and their simulation environment is built. In addition, the Velodyne lidar and IMU are deployed on the excavator model to enable it to collect environmental information and vehicle movement information.

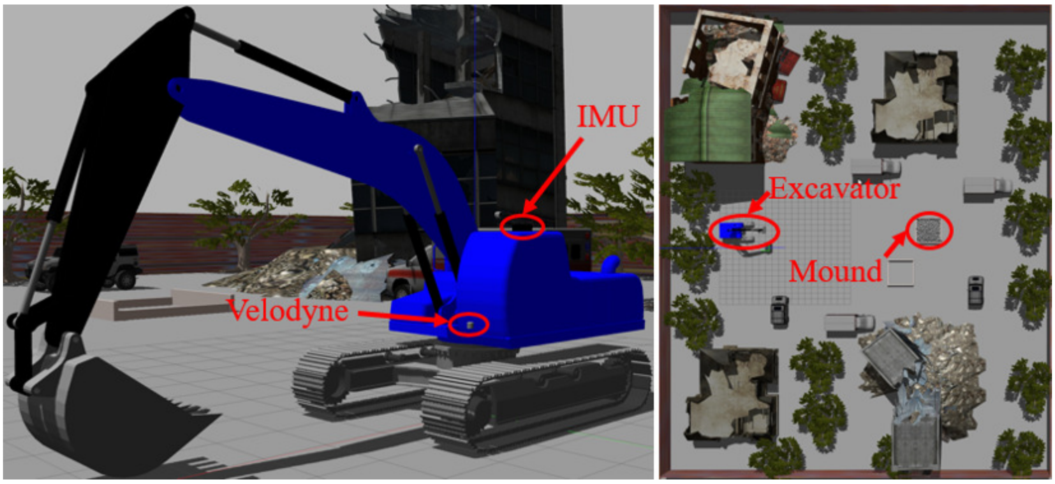


Figure 10. Construction machinery simulation platform based on Gazebo.

Based on the Gazebo construction machinery simulation platform, combined with the PointPillars, Hybrid A* and improved pure pursuit algorithm proposed in this paper, the simulation tests are carried out on the functional modules of automatic path finding and autonomous obstacle avoidance of construction machinery.

5.1. Simulation of Automatic Routing Function of Construction Machinery

With the lidar point cloud data simulated by Gazebo as the input, the weight file obtained by using the PointPillars algorithm to train the working condition data set of construction machinery in this paper is verified, and the effect is shown in Figure 11.

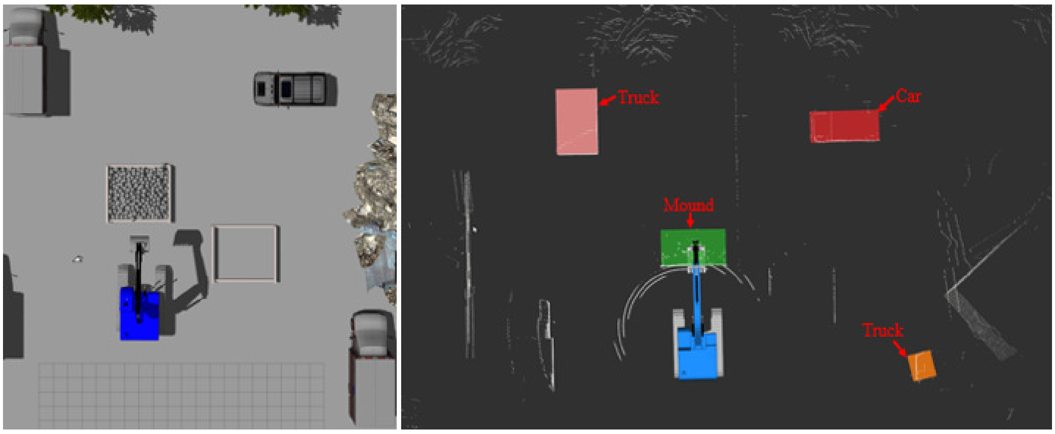


Figure 11. Simulation and verification of PointPillars algorithm.

It can be seen from Figure 11 that the PointPillars algorithm has effectively detected cars, mounds and trucks, without missing or wrong detection. It is proved that it is feasible to detect the position and attitude of the target object relative to the vehicle as a perception algorithm.

Take the position and attitude of the whole vehicle currently located on the point cloud map as the starting point of path planning, and take the position and attitude of the work object detected by the PointsPillars algorithm as the target point of path planning, and use hybrid A* to independently plan a feasible path, with the effect shown in Figure 12. Among them, the point cloud map is used to match the observation point cloud of the lidar carried by the vehicle to obtain the vehicle localization information in real time. The grid map serves as the prior information of hybrid A* for path planning, and provides a basis for hybrid A* to plan a safe and reliable global path.

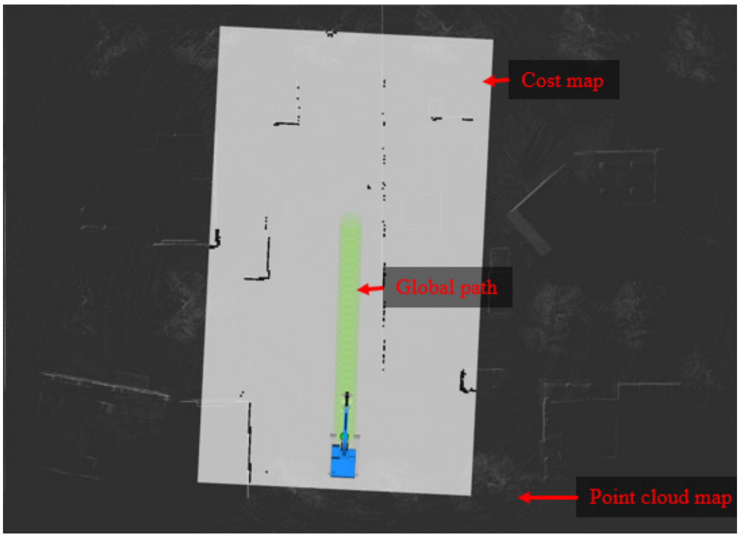


Figure 12. Effect picture of automatic routing.

After the hybrid A* has planned the global path required for the construction machinery to travel near the work object, the improved pure pursuit algorithm proposed in this paper is used to control the motion of the built-tracked construction machinery, so that it can track along the global path. The tracking speed set in this paper is 3 km/h, i.e., 0.833 m/s, which corresponds to the maximum speed that can be reached by the rabbit block of the actual tracked construction machinery. Figure 13 shows the tracking speed of the left and right tracks when the tracked construction machinery is under motion control.

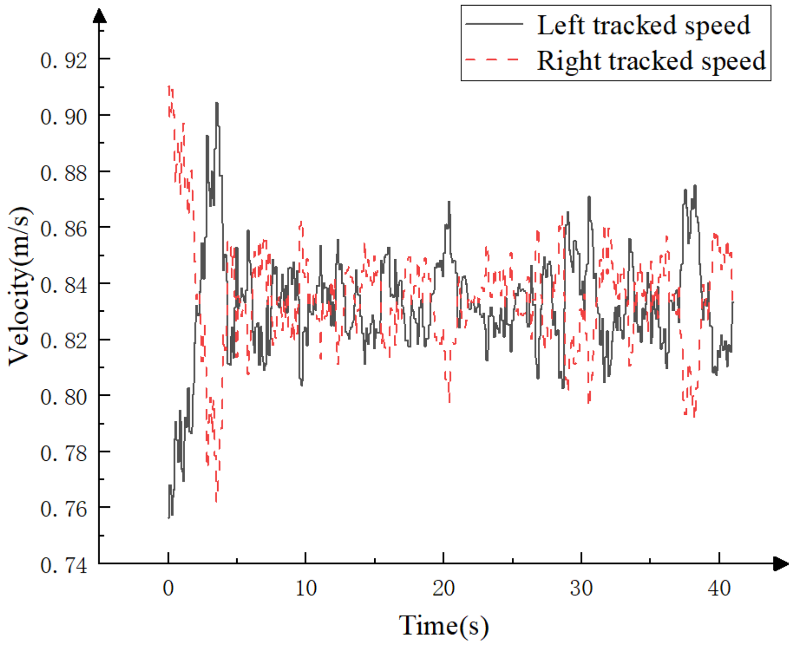


Figure 13. Tracking speed of left and right tracks for automatic routing.

The control accuracy of the improved pure pursuit algorithm proposed in this paper is analyzed, and the results are shown in Figures 14 and 15.

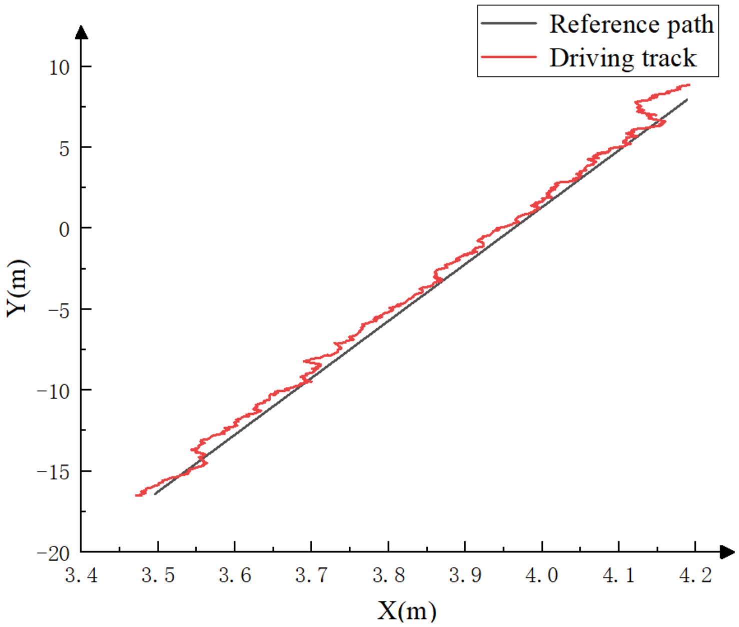


Figure 14. Comparison diagram of automatic routing trajectory.

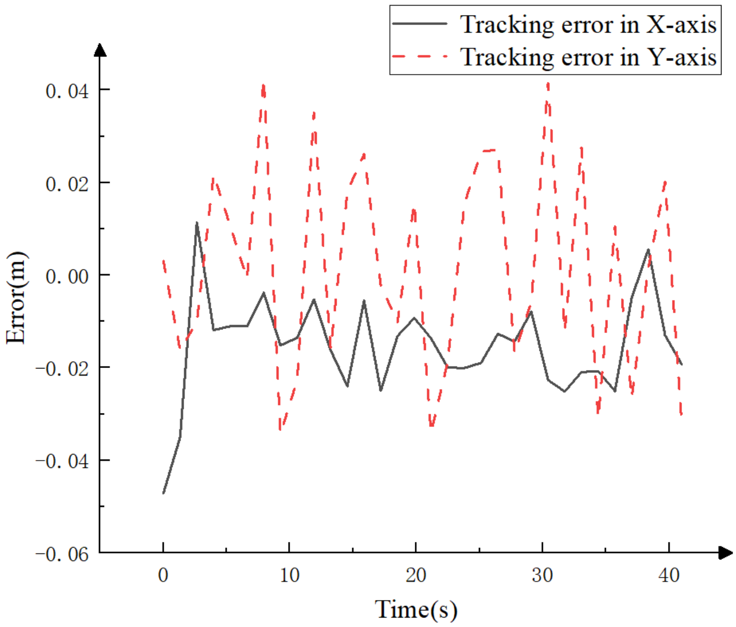


Figure 15. Automatic routing track tracking error.

It can be seen from the analysis in Figures 14 and 15 that the improved pure pursuit algorithm proposed in this paper has a good effect on the motion control of the whole vehicle, and the tracking track is basically consistent with the reference path. The maximum tracking error in the X-axis direction is 0.0472 m, and the average error is 0.0164 m; The maximum tracking error in the Y-axis direction is 0.0426 m, and the average error is 0.0198 m, which meets the accuracy requirements of construction machinery motion control.

5.2. *Simulation of Autonomous Obstacle Avoidance Function of Construction Machinery*

Place two roadblocks between the tracked construction machinery model and the work object to test the autonomous obstacle avoidance function of the construction machinery, as shown in Figure 16.

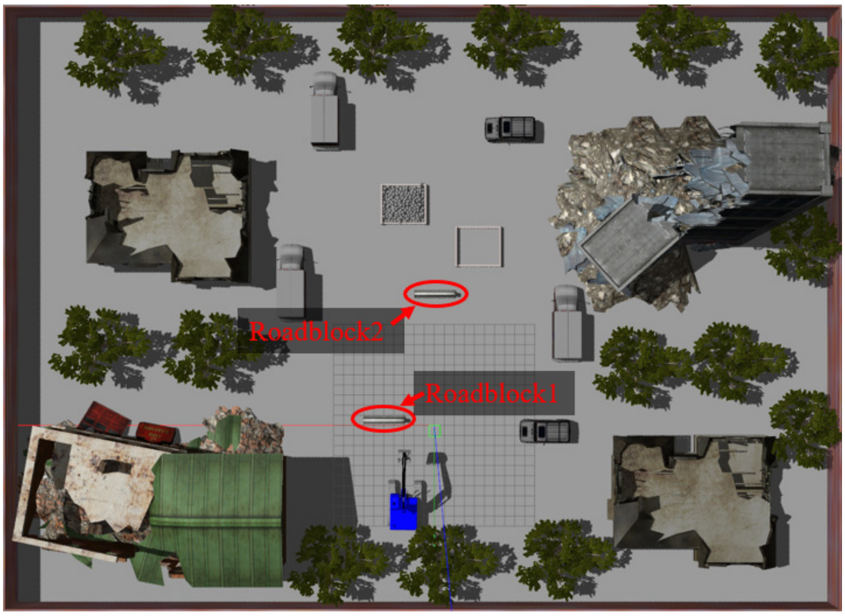


Figure 16. Autonomous obstacle avoidance simulation environment.

Similar to Section 4.1, the PointPillars algorithm is used to detect roadblocks and work objects to obtain the position and attitude of roadblocks and work objects relative to the vehicle. The actual detection effect is shown in Figure 17.

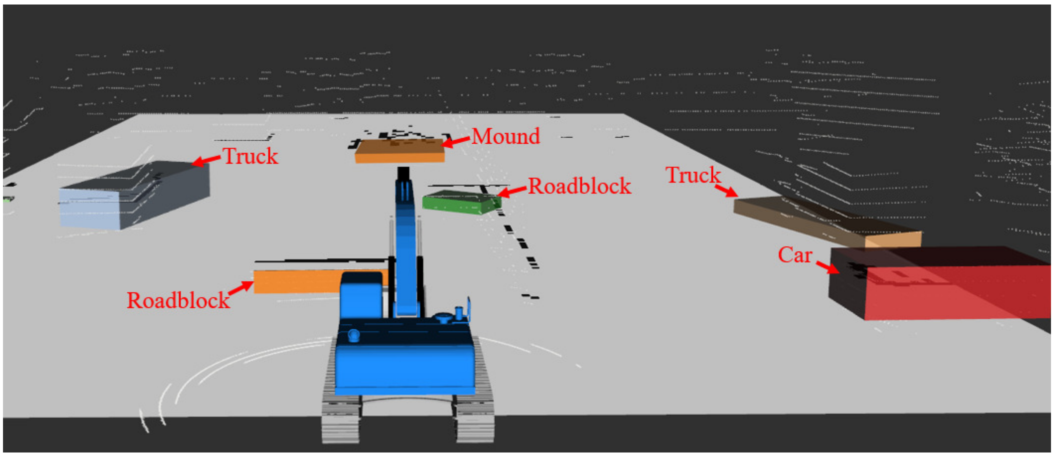


Figure 17. PointPillars algorithm detection rendering.

After obtaining the position and attitude information of roadblocks and work objects through the perception algorithm, combined with the information of obstacles and accessible areas provided by the grid map, the hybrid A* algorithm is used to plan a global path that meets the kinematics of tracked construction machinery and can achieve obstacle avoidance, as shown in Figure 18, which is used for the track tracking control of subsequent vehicles.

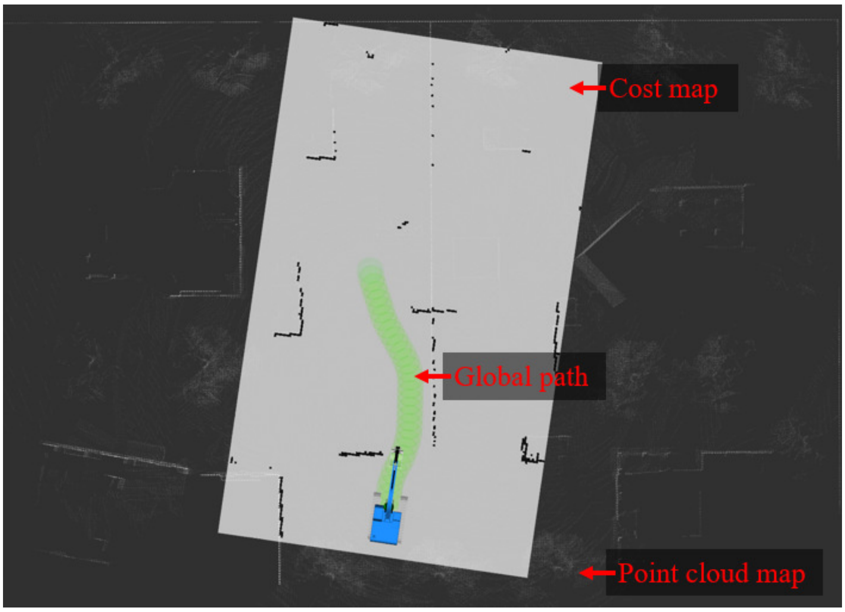


Figure 18. Rendering of autonomous obstacle avoidance path generation.

Similarly, after the hybrid A* algorithm has planned the global obstacle avoidance path, the improved pure pursuit algorithm proposed in this paper is used to control its motion. The tracking speed is still set at 3 km/h, i.e., 0.83 m/s. The walking and obstacle avoidance of the tracked construction machinery model are controlled by controlling the walking speed of the left and right tracks. Figure 19 shows the tracking speed of the left and right tracks that control the movement of the tracked construction machinery during autonomous obstacle avoidance.

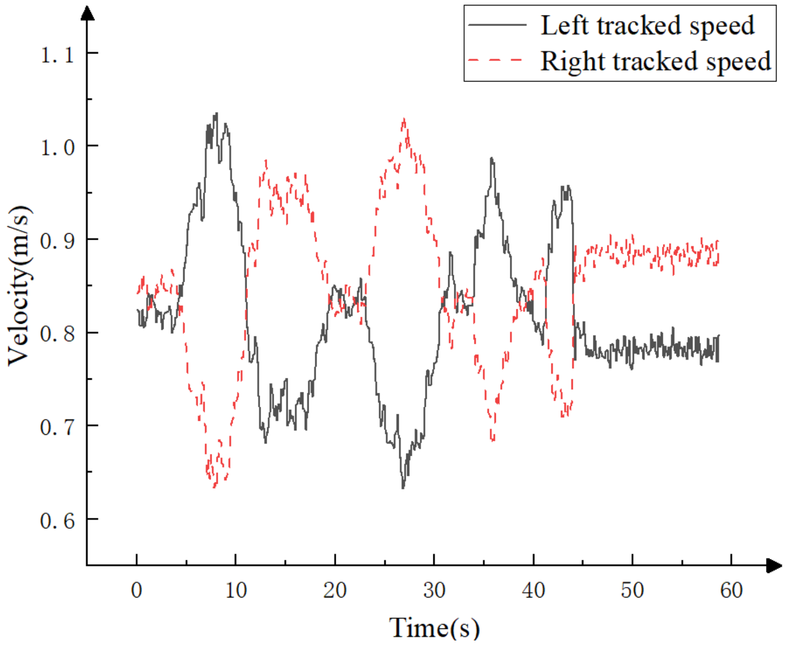


Figure 19. Tracking speed of left and right tracks for autonomous obstacle avoidance.

Similarly, the control accuracy of the improved pure pursuit algorithm proposed in this paper is analyzed, and the results are shown in Figures 20 and 21.

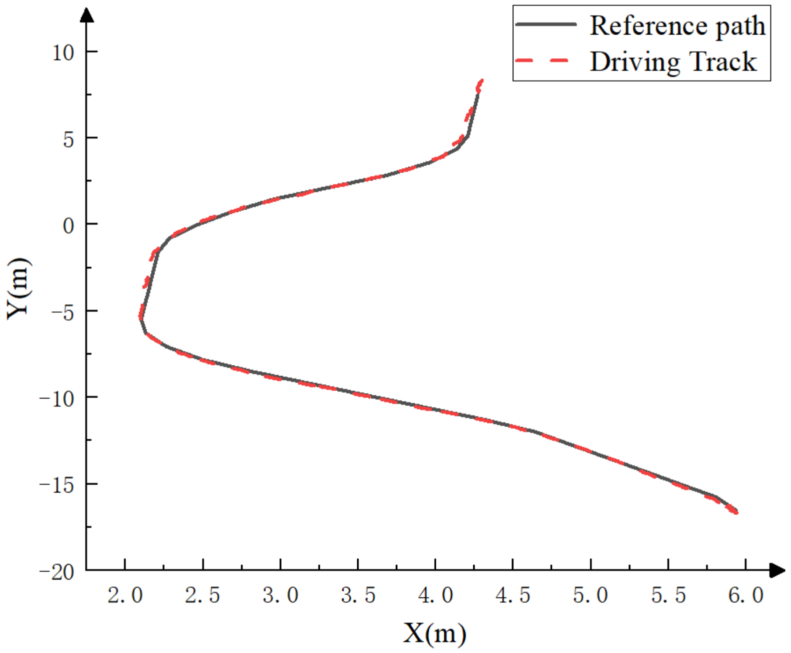


Figure 20. Comparison diagram of autonomous obstacle avoidance trajectory.

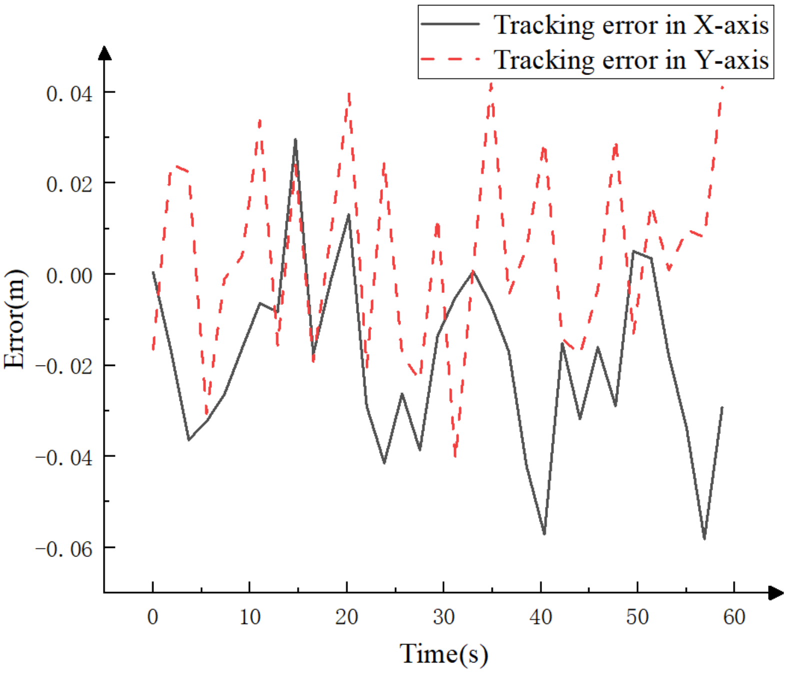


Figure 21. Track tracking error of autonomous obstacle avoidance.

It can be seen from the analysis in Figures 20 and 21 that the improved pure pursuit algorithm proposed in this paper can still ensure good vehicle motion control accuracy when autonomous obstacle avoidance, and the vehicle travel path is basically consistent with the reference path. The maximum tracking error in the X-axis direction is 0.0581 m, and the average tracking error is 0.0219 m; the maximum tracking error of the Y-axis direction is 0.0418 m, and the average tracking error is 0.0187 m, which meets the requirements of construction machinery for motion control accuracy.

6. Construction of Real Vehicle Test Platform and Relevant Tests

In order to further verify the algorithms proposed in this paper, a real vehicle test platform for tracked construction machinery is built, as shown in Figures 22 and 23.

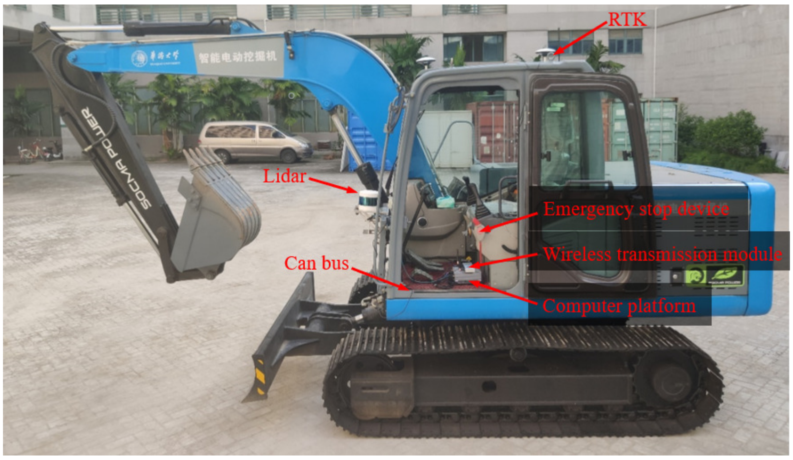


Figure 22. Tracked construction machinery real vehicle test platform.

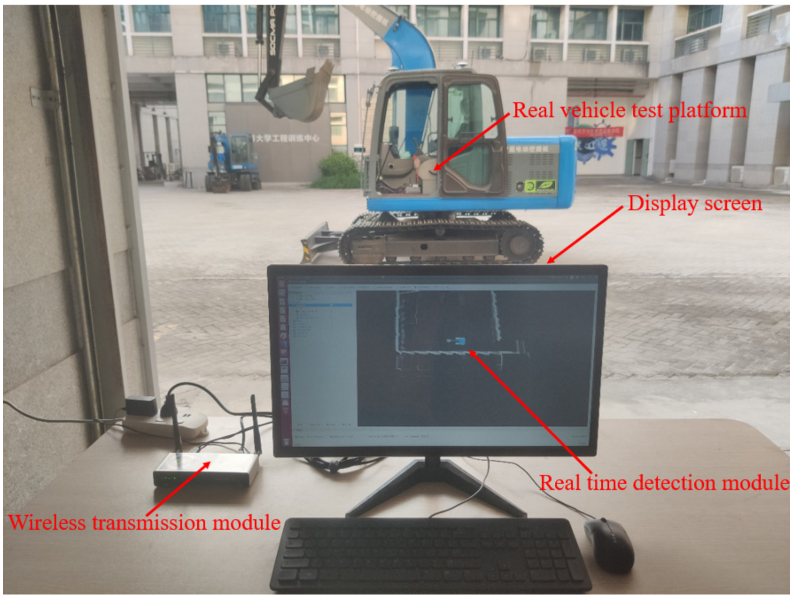

Real vehicle test platform
Display screen
Real time detection module
Wireless transmission module

Figure 23. Real vehicle test debugging and supervision platform.

Tracked construction machinery real vehicle test platform is mainly equipped with modules such as lidar, RTK, CAN bus, emergency stop device, computing platform and wireless image transmission. Among them, the lidar is mainly responsible for collecting the environmental point cloud information required by the perception and localization algorithm, RTK is responsible for recording the vehicle's travel path and correcting the vehicle's position and attitude calculated by the point cloud localization algorithm, and the CAN bus is responsible for sending the control signal output by the computing platform into the vehicle controller, so as to realize the vehicle's motion control. The emergency stop equipment is the safety guarantee device for the whole vehicle driverless test. When there is a problem in the algorithm operation, the safety officer can dial the emergency stop equipment to realize the whole vehicle emergency stop to ensure the safety of the safety officer and relevant test equipment. The computing platform is the brain of the entire unmanned vehicle test platform, responsible for the input and output of management information, and supporting the operation of algorithms such as perception, localization, planning, decision-making and motion control. Wireless image transmission is responsible for transmitting the image information of the computing platform to the real vehicle test debugging and supervision platform, so as to realize the debugging of each driverless algorithm and real-time monitoring of the vehicle status, and fully guarantee the safety of the test personnel.

In the real vehicle test, the position and attitude of the object detected by the PointPillars algorithm relative to the whole vehicle are still used as the target point of path planning, the current position and attitude of the whole vehicle are used as the starting point of path planning, and two roadblocks are set between the starting point and the target point, respectively, to verify the automatic operation path finding and autonomous obstacle avoidance functions of the whole vehicle driverless system. The actual test scenario is shown in Figures 24 and 25.

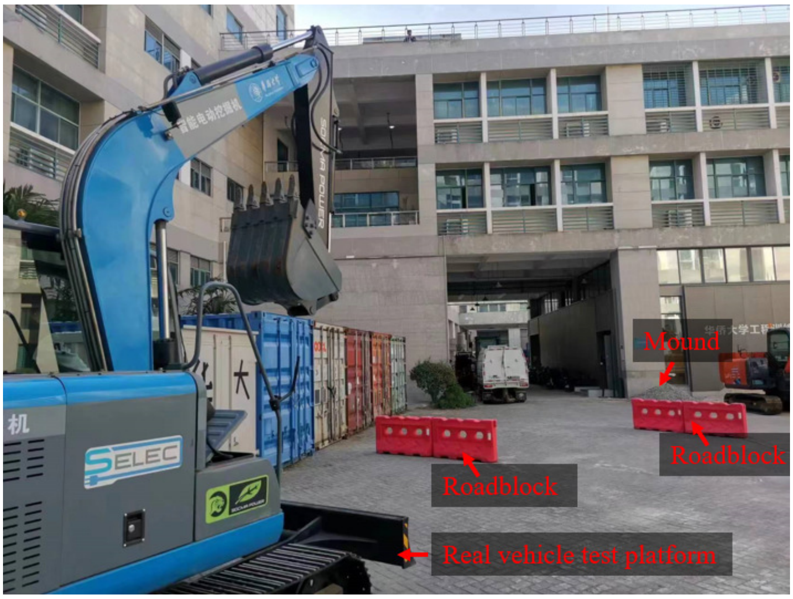


Figure 24. Autonomous path finding and obstacle avoidance test scenario of real vehicle.

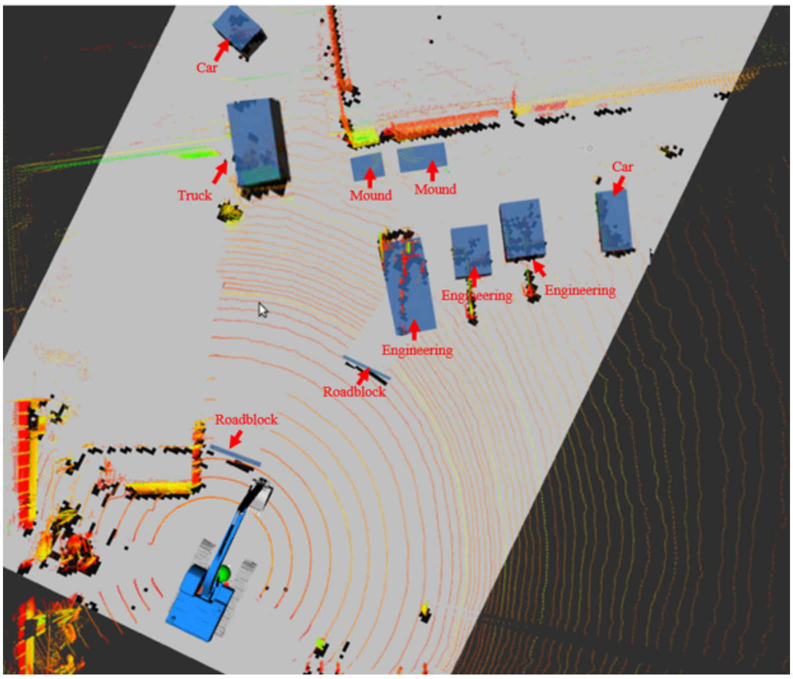


Figure 25. PointPillars algorithm real vehicle detection rendering.

After obtaining the position and orientation information of the starting point, the target point and the obstacle, combined with the prior information of the obstacle and the accessible area provided by the grid map, the hybrid A* algorithm is used to plan a global path that can reach the vicinity of the work object and can avoid the obstacle, as shown in Figure 26.

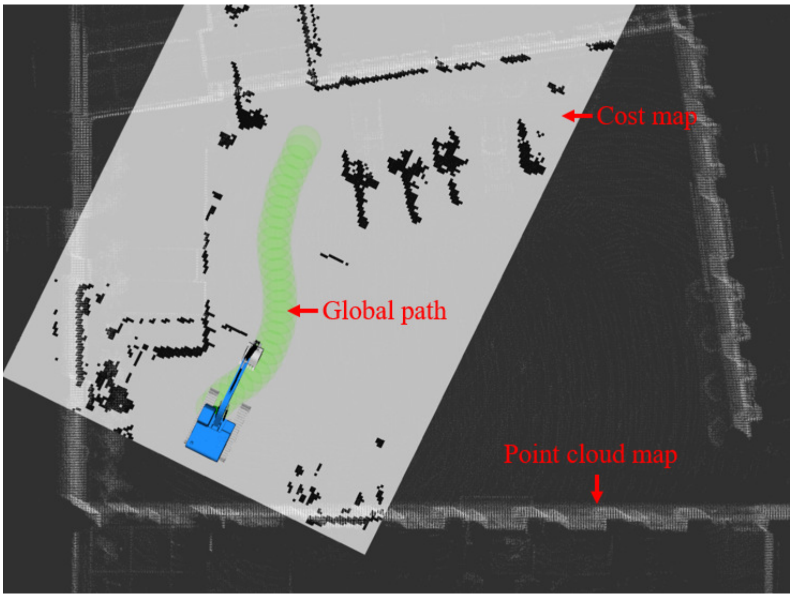


Figure 26. Rendering of global path generation of real vehicle.

After the hybrid A* algorithm has planned the global path, the improved pure pursuit algorithm proposed in this paper is used to control the motion of tracked construction machinery so that it can complete the tracking task of the reference global path. In the real vehicle path tracking test, the set tracking speed is 2 km/h, i.e., 0.56 m/s. Figure 27 shows the left and right track speed signals sent by the vehicle-mounted computing platform to the vehicle controller during the track tracking process of tracked construction machinery. After receiving the left and right track speed signals, the controller will convert them into the corresponding duty cycle, thus realizing the vehicle motion control.

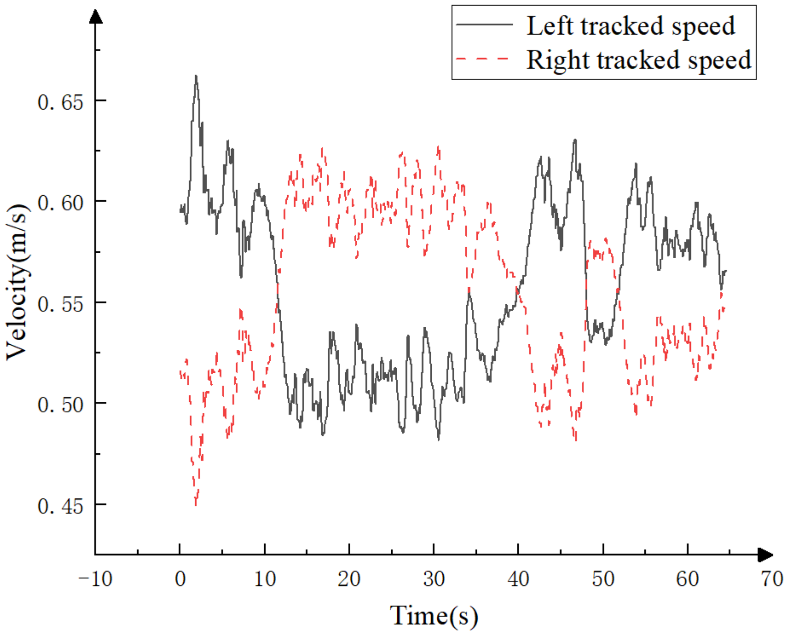


Figure 27. Tracking speed of left and right tracks for real vehicle autonomous path finding and obstacle avoidance.

The motion control accuracy of the whole vehicle is analyzed, and the results are shown in Figures 28 and 29. The analysis shows that the improved pure pursuit algorithm proposed in this paper has excellent performance in real vehicle motion control, and the tracking trajectory is basically consistent with the reference global path planned by hybrid

A*. The maximum tracking error in the X-axis direction is 0.154 m, and the average tracking error is 0.069 m; the maximum tracking error in the Y-axis direction is 0.123 m, and the average tracking error is 0.037 m, which meets the requirements of actual construction machinery for motion control accuracy. Moreover, the improved pure pursuit algorithm proposed in this paper can not only be applied to tracked construction machinery but also be adjusted in combination with the kinematics model of other construction machinery to complete the motion control of various construction machinery.

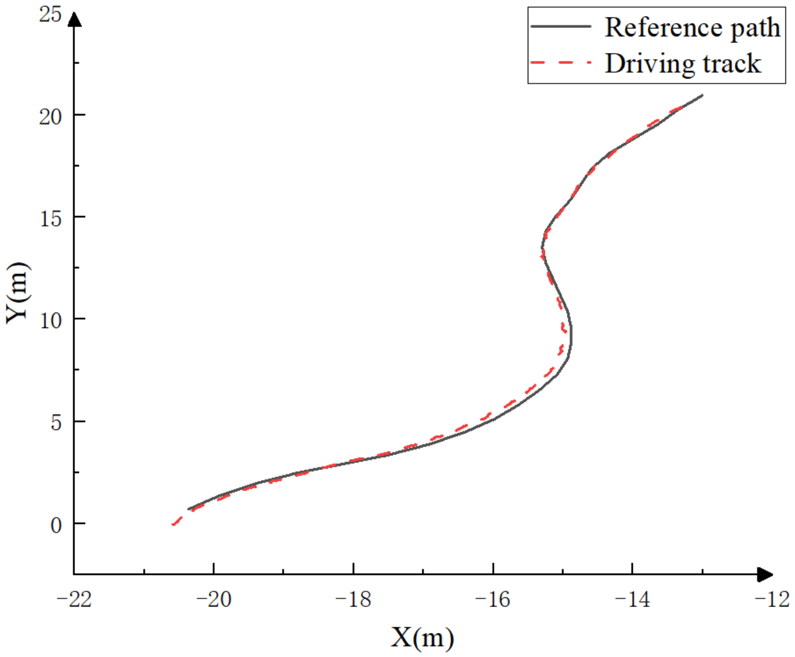


Figure 28. Comparison diagram of autonomous path finding and obstacle avoidance trajectory of real vehicle.

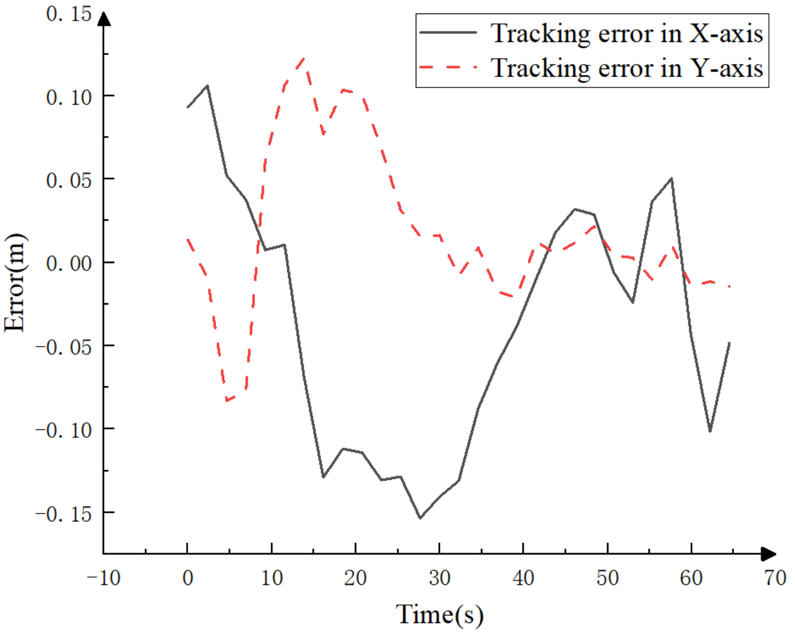


Figure 29. Autonomous path finding and obstacle avoidance trajectory tracking error of real vehicle.

7. Conclusions

In order to realize the autonomous path finding of construction machinery, this paper uses the PointPillars algorithm to sense the object and its position, and uses the hybrid A* algorithm to generate a global path to the object independently. Finally, based on the improved pure pursuit algorithm proposed in this paper, it completes the motion control and tracks tracking of the whole vehicle.

In view of the problem that there are few data sets conforming to the working conditions of construction machinery at present, this paper uses the SuSTechPoints annotation software to produce the construction machinery point cloud data set, and uses the PointPillars algorithm to train the data set, and verifies it on the simulation and real vehicle test platform. Aiming at the problem that the traditional path planning algorithm cannot plan a smooth path that meets the kinematics of construction machinery, this paper uses the hybrid A* algorithm to take the kinematics characteristics of construction machinery into account in the path planning, effectively solving the problem of uneven path planning by the common path planning algorithm.

Aiming at the problem that the existing pure pursuit algorithm is not applicable to the motion control of tracked construction machinery, this paper improves the traditional pure pursuit algorithm based on the kinematics model of tracked construction machinery, and carries out corresponding verification through simulation and real vehicle tests. The test results show that the improved pure pursuit algorithm proposed in this paper can perform better motion control of tracked construction machinery, and the accuracy of motion control can reach about 10 cm, It can well meet the requirements of construction machinery for motion control accuracy. At the same time, the improved pure pursuit algorithm proposed in this paper can also make appropriate adjustments to the kinematics characteristics of other construction machinery, so that it can be applied to the motion control of various construction machinery.

During the experiment, some problems with existing algorithms were also found. First, when using the PointPillars algorithm for target detection, there were problems such as the incorrect orientation of individual target detection boxes, and the size of the detection box did not match the actual object size. Subsequent work will focus on the post-processing part of target detection, while further optimizing the existing detection network to improve its detection efficiency and accuracy. Second, the current hybrid A* algorithm still has some limitations. It can only be used for global path planning in small and medium-sized scenarios, and cannot be used for path planning in large mine scenarios. The existing hybrid A* algorithm will be further improved in the future to improve its search efficiency.

Author Contributions: Conceptualization, J.W.; methodology, J.W. and Z.F.; software, J.W. and C.L.; validation, J.W. and C.L.; formal analysis, J.W. and Z.F.; investigation, Y.Y.; resources, T.L. and H.R.; data curation, J.W. and Z.F.; writing—original draft preparation, J.W.; writing—review and editing, H.R.; visualization, J.W. and H.R.; supervision, T.L. and H.R.; project administration, J.W., T.L. and H.R.; funding acquisition, T.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by National Natural Science Foundation of China (Grant No. 52275055 & 52175051), Key projects of natural science foundation of Fujian Province (Grant No. 2021J02013), and Fujian University industry university research joint innovation project plan (Grant No. 2022H6007 & 2022H6028).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lin, T.; Yao, Y.; Xu, W.; Fu, S.; Ren, H.; Chen, Q. Driverless walking method of electric construction machinery based on environment recognition. *Jixie Gongcheng Xuebao* **2021**, *57*, 42–49. [CrossRef]

2. Lang, A.H.; Vora, S.; Caesar, H.; Zhou, L.; Yang, J.; Beijbom, O. Pointpillars: Fast encoders for object detection from point clouds. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 12697–12705. [CrossRef]
3. Dolgov, D.; Thrun, S.; Montemerlo, M.; Diebel, J. Path planning for autonomous vehicles in unknown semi-structured environments. *Int. J. Robot. Res.* **2010**, *29*, 485–501. [CrossRef]
4. Wallace, R.; Stentz, A.; Thorpe, C.; Moravec, H.; Whittaker, W.; Kanade, T. First Results in Robot Road-Following. In Proceedings of the 9th International Joint Conference on Artificial Intelligence, Los Angeles, CA, USA, 18–23 August 1985; pp. 1089–1095.
5. Zhou, Y.; Tuzel, O. Voxelnet: End-to-end learning for point cloud based 3d object detection. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4490–4499. [CrossRef]
6. Yan, Y.; Mao, Y.; Li, B. Second: Sparsely embedded convolutional detection. *Sensors* **2018**, *18*, 3337. [CrossRef]
7. Zhang, Y.; Xiang, Z.; Qiao, C.; Chen, S. Accurate and real-time object detection based on bird's eye view on 3d point clouds. In Proceedings of the International Conference on 3D Vision (3DV), Quebec City, QC, Canada, 16–19 September 2019; pp. 214–221. [CrossRef]
8. Li, E.; Wang, S.; Li, C.; Li, D.; Wu, X.; Hao, Q. Sustech points: A portable 3d point cloud interactive annotation platform system. In Proceedings of the IEEE Intelligent Vehicles Symposium, Las Vegas, NV, USA, 19 October–13 November 2020; pp. 1108–1115. [CrossRef]
9. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.; Berg, A. Ssd: Single shot multibox detector. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 21–37. [CrossRef]
10. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988. [CrossRef]
11. Dijkstra, E.W. A note on two problems in connexion with graphs. *Numer. Math.* **1959**, *1*, 269–271. [CrossRef]
12. Likhachev, M.; Ferguson, D. Planning long dynamically feasible maneuvers for autonomous vehicles. *Int. J. Robot. Res.* **2009**, *28*, 933–945. [CrossRef]
13. Stentz, A. Optimal and efficient path planning for partially known environments. In *Intelligent Unmanned Ground Vehicles: Autonomous Navigation Research at Carnegie Mellon*; Springer: Berlin/Heidelberg, Germany, 1997; pp. 203–220. [CrossRef]
14. Zhang, T.; Zhu, Y.; Song, J. Real-time motion planning for mobile robots by means of artificial potential field method in unknown environment. *Ind. Robot* **2010**, *37*, 384–400. [CrossRef]
15. Kavraki, L.; Svestka, P.; Latombe, J.C.; Overmars, M. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Robot. Autom.* **1996**, *12*, 566–580. [CrossRef]
16. LaValle, S.M.; Kuffner, J.J., Jr. Randomized kinodynamic planning. *Int. J. Robot. Res.* **2001**, *20*, 378–400. [CrossRef]
17. Kuffner, J.J.; LaValle, S.M. An efficient approach to single-query path planning. In Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, CA, USA, 24–28 April 2000; pp. 473–479. [CrossRef]
18. Gammell, J.D.; Srinivasa, S.S.; Barfoot, T.D. Informed RRT: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. In Proceedings of the International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 2997–3004. [CrossRef]
19. Karaman, S.; Walter, M.R.; Perez, A.; Frazzoli, E.; Teller, S. Anytime motion planning using the RRT. In Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 1478–1483. [CrossRef]
20. Lefevre, S.; Carvalho, A.; Borrelli, F. A learning-based framework for velocity control in autonomous driving. *IEEE Trans. Autom. Sci. Eng.* **2015**, *13*, 32–42. [CrossRef]
21. Gorinevsky, D.; Kapitanovsky, A.; Goldenberg, A. Neural network architecture for trajectory generation and control of automated car parking. *IEEE Trans. Control Syst. Technol.* **1996**, *4*, 50–56. [CrossRef]
22. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95—International Conference on Neural Networks IEEE, Perth, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948. [CrossRef]
23. Dorigo, M.; Birattari, M.; Stutzle, T. Ant colony optimization. *IEEE Comput. Intell. Mag.* **2006**, *1*, 28–39. [CrossRef]
24. Holland, J. Genetic algorithms. *Sci. Am.* **1992**, *267*, 66–73. [CrossRef]
25. Bojarski, M.; Testa, D.D.; Dworakowski, D.; Firner, B.; Flepp, B.; Goyal, P.; Lawrence, D.J.; Monfort, M.; Muller, U.; Zhang, J.; et al. End to end learning for self-driving cars. *arXiv* **2016**, arXiv:1604.07316. [CrossRef]
26. Konar, A.; Chakraborty, I.G.; Singh, S.J.; Jain, L.C.; Nagar, A.K. A deterministic improved Q-learning for path planning of a mobile robot. *IEEE Trans. Syst. Man Cybern. Syst.* **2013**, *43*, 1141–1153. [CrossRef]
27. Cruz, G.C.S.; Encarnação, P.M.M. Obstacle avoidance for unmanned aerial vehicles. *J. Intell. Robot. Syst.* **2012**, *65*, 203–217. [CrossRef]
28. Aguilar, W.G.; Casaliglla, V.P.; Pólit, J.L. Obstacle avoidance based-visual navigation for micro aerial vehicles. *Electronics* **2017**, *6*, 10. [CrossRef]
29. Hermand, E.; Nguyen, T.W.; Hosseinzadeh, M.; Garone, E. Constrained control of UAVs in geofencing applications. In Proceedings of the 2018 26th Mediterranean Conference on Control and Automation (MED) IEEE, Akko, Israel, 1–4 July 2018; pp. 217–222. CrossRef]
30. Choset, H.; Burdick, J. Sensor-based exploration: The hierarchical generalized voronoi graph. *Int. J. Robot. Res.* **2000**, *19*, 96–125. [CrossRef]

31. Reeds, J.; Shepp, L. Optimal paths for a car that goes both forwards and backwards. *Pac. J. Math.* **1990**, *145*, 367–393. [CrossRef]
32. Thrun, S.; Montemerlo, M.; Dahlkamp, H.; Stavens, D.; Aron, A.; Diebel, J.; Fong, P.; Gale, J.; Halpenny, M.; Hoffmann, G.; et al. Stanley: The robot that won the DARPA Grand Challenge. *J. Field Robot.* **2006**, *23*, 661–692. [CrossRef]
33. Xie, X.; Zhu, C.; Xie, M. Visual tracking control of SCARA robot system based on deep learning and Kalman prediction method. *Int. J. Hydromechatron.* **2021**, *4*, 384–396. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.