

Research Article

A Road Environment Prediction System for Intelligent Vehicle

Chao Ma,¹ Zhao Sun ,¹ Shanshan Pei,¹ Chao Liu,² and Feng Cui ³

¹Faculty of Information Technology, Macau University of Science and Technology, Macau 999078, China

²Research Base of Beijing Modern Manufacturing Development, Beijing University of Technology, Beijing 100124, China

³Beijing Smarter Eye Technology Co.Ltd, Beijing 100023, China

Correspondence should be addressed to Feng Cui; feng.cui@smartereye.com

Received 18 February 2021; Revised 16 March 2021; Accepted 26 March 2021; Published 30 April 2021

Academic Editor: Wenqing Wu

Copyright © 2021 Chao Ma et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The road environment prediction is an essential task for intelligent vehicle. In this study, we provide a flexible system that focuses on freespace detection and road environment prediction to host vehicle. The hardware of this system includes two parts: a binocular camera and a low-power mobile platform, which is flexible and portable for a variety of intelligent vehicle. We put forward a multiscale stereo matching algorithm to reduce the computing cost of the hardware unit. Based on disparity space and points cloud, we propose a weighted probability grid map to detect freespace region and a state model to describe the road environment. The experiments show that the proposed system is accurate and robust, which indicates that this technique is fully competent for road environment prediction for intelligent vehicle.

1. Introduction

The road environment prediction is an essential task for intelligent vehicle and robotic applications. As the basis of path planning [1], motion strategy [2], and collision avoidance [3], the road environment prediction focuses on whether the host vehicle or robot will pass through without collision [4].

In recent years, light detection and ranging (LiDAR) [5], cameras [6], and multisensor fusion technique are adopted to perceive the road environment. In literatures, LiDAR and cameras are devoted to the odometry method, in which the relative motion is estimated by matched points in continuous frames. The multisensors fusion technique includes visual sensors, inertial sensors, and precision map. The inertial sensors are used to estimate the relative pose based on laws of classical mechanics. Those methods are considered as passive sensing that has only detected road information when a vehicle drives into it. The precision map method relies on a global positioning system (GPS) signal that can predict the road path; however, it is not robust in tunnel or underground parking.

In this paper, we propose a flexible and robust road environment perception system. This system consists of a binoc-

ular camera and a low-power mobile platform. Because that software functions run in the terminal device rather than transmitting a large amount of data to the cloud [7], the proposed system is suitable for the Internet of Things (IoT) [8]. For the sake of real-time system in the low-power platform, we propose a multiscale stereo matching algorithm, weight probability grid map, and state model to describe road environment.

The traditional stereo matching algorithms, such as [9, 10], provide dense disparity map by predefined matching measures and semiglobal optimization, which consumes a lot of computing sources and takes a large of electric power. In this study, we propose a new stereo matching framework to adapt to the field-programmable gate array (FPGA) implement environment. The focus of the improvement is to reduce the computational cost, maintain the matching accuracy, and effectively perceive the target in different scales.

The traditional road environment perception methods, such as [11, 12], focus on the detection of freespace, which estimates the boundary of freespace region by geometric constraints in u -disparity or v -disparity space. The traditional position estimate methods, such as [13, 14], put forward motion estimation method based on the Bundle Adjustment

(BA) algorithm with feature points. It is necessary to select the points which are motionless relative to world coordinate system from the feature points. In this paper, we propose a weighted probability grid map to freespace detection. It is a robust and flexible strategy because we avoid the motion estimate that is different to match the feature point on static objects. In addition, we define a state model to describe the road environment, which describes the road status in the front of the host vehicle.

The main contributions of this work are summarized as follows.

- (i) A multiscale stereo matching algorithm is presented to reduce the computing cost and improve the accuracy
- (ii) Based on the disparity map, a weight probability grid map is proposed to detection the freespace region
- (iii) A state model is proposed to describe the road environment in the front of the host vehicle
- (iv) An efficient deployment programme is put forward to process our system at the low-power mobile platform in realtime

2. Related Work

2.1. Stereo Matching. Humenberger et al. [15] have presented a large sparse Census mask for the fast stereo matching algorithm. In this literature, authors compare the large sparse masks and small dense masks, which the experiment shows that the former has better performance.

Yang [16] has presented a new matching cost aggregation method to preserve depth edges and reduce the computing complexity. In this literature, the author suggests an adaptive cost aggregation technique based on pixel similarity. A bilateral filter is used to compute the aggregation cost with the spatial similarity and the range (intensity/color) similarity, respectively.

Zhang et al. [17] have presented a cross-scale cost aggregation framework to allow multiscale interaction in cost aggregation. In this literature, the authors consider the cost aggregation as a weighted least square optimization problem. Therefore, multiscale cost aggregation methods come from different similarity kernels in the optimization objective.

Mao et al. [18] have presented a robust deep learning solution for semidense stereo matching. In this literature, two CNN models are utilized for computing stereo matching cost and performing confidence-based filtering. Due to that the global information is provided, the model is suitable for dealing with the challenging cases, such as lighting changes and lack of textures.

However, the above algorithms cost a large number of computing resources, which is a risk to process in realtime. In this paper, we propose a multiscale stereo matching fusion algorithm. This algorithm is designed to reduce the computing cost of FPGA and process stereo matching in realtime.

2.2. Road Environment Perception. Qu et al. [19] have presented an improved V -disparity space algorithm to generate the confidence map to estimate the freespace. In this literature, the authors discuss a sub- V -disparity space method to avoid the assumption that the road is locally planar and the variance in the latitudinal slope is small. Based on the v -disparity space, road confidence map and obstacle confidence map are calculated for freespace estimation by dynamic programming.

Deilamsalehy and Havens [20] have presented a multi-sensor fusion method to improve the accuracy of the estimation and to compensate for individual sensor deficiencies. In this literature, the three sensors, inertial measurement unit (IMU), camera, and LiDAR, form a rigid body, where position estimates from the camera and LiDAR are used in the EKF as a measurement to correct the IMU pose estimation.

Xiao et al. [21] have presented a Bayesian framework and conditional random field to fuse the multiple features that includes 2D image and 3D point cloud geometric information. Besides, a Gaussian process regression is employed to enhance performance. In this literature, the results are outstanding compared to some relevant LiDAR-based methods when a conditional random field with color and geometry constraints is applied to make the result more robust.

Zheng et al. [22] have presented a low-cost GPS-assisted LiDAR state estimation system for autonomous vehicle. In this literature, a LiDAR is employed to obtain highly precise 3D geometry data and an IMU is used to correct point cloud misalignment. A low-cost GPS is used to refine the estimated LiDAR inertial odometry.

Cong et al. [23] have presented LiDAR-based simultaneous localization and mapping (SLAM) system embedding dynamic objects removal module to improve the pose estimation. In this literature, the authors remove the point clouds of moving objects to relieve their influence on the odometry, so that the precision relative pose is estimated.

In this paper, we propose a weighted probability grid map to freespace detection and a state model to describe the road environment. By describing of road state in front of the host vehicle, we predict the future road environment information and vehicle's estimation direction.

3. Proposed Method

3.1. Multiscale Stereo Matching Algorithm. In binocular stereo matching algorithm, feature similarity is an unsupervised matching method, such as census feature [15], structural similarity (SSIM) [10], and convolutional neural network (CNN) model [24]. In this study, our binocular system is set to synchronous exposure and same image signal processing (ISP) parameters. In order to tolerate the small grey difference between the two sensors in the imaging process, the SSIM method is adopted for stereo matching.

The SSIM method [25, 26] is proposed to perceive image quality, because this index considers the luminance ($l(\varphi, \phi)$), contrast ($c(\varphi, \phi)$), and structure ($s(\varphi, \phi)$) as shown in

Equation. (1), where φ is the index of row and ϕ is the index of column.

$$\begin{aligned} l(\varphi, \phi) &= \frac{2\mu_\varphi\mu_\phi + C_1}{\mu_\varphi^2 + \mu_\phi^2 + C_1} \\ c(\varphi, \phi) &= \frac{2\sigma_\varphi\sigma_\phi + C_2}{\sigma_\varphi^2 + \sigma_\phi^2 + C_2}, \\ s(\varphi, \phi) &= \frac{\sigma_\varphi\sigma_\phi + C_3}{\sigma_{\varphi\phi} + C_3} \end{aligned} \quad (1)$$

where μ , σ^2 , and σ denote the mean, variance, and covariance, respectively. Approximately, μ and σ^2 are viewed as the estimated luminance and contrast, respectively, and σ measures the tendency. C_1 , C_2 , and C_3 are constants. The general form of the SSIM index between u (disparity) and p (pixel coordinate in image) is defined as Equation (2):

$$\text{SSIM}(p, u) = l^\alpha(\varphi, \phi) c^\beta(\varphi, \phi) s^\gamma(\varphi, \phi). \quad (2)$$

In engineering practice, α , β , and γ are set as 1 and $C_1 = 0.01$, $C_2 = 0.03$, and $C_3 = C_2/2$. Therefore, the matching similarity value is simplified as Equation (3):

$$\text{SSIM}(p, u) = \frac{(2\mu_\varphi\mu_\psi + C_1)(2\sigma_{\varphi\psi} + C_2)}{(\mu_\varphi^2 + \mu_\psi^2 + C_1)(\sigma_\varphi^2 + \sigma_\psi^2 + C_2)}, \quad (3)$$

which satisfies the following conditions to be a matching cost:

- (i) *Boundedness.* $0 < \text{SSIM} \leq 1$, which indicates that the more similar the patch, the higher the value
- (ii) *Unique Extremum.* $\text{SSIM} = 1$ if and only if the two patches are the same

In order to adapt to the high dynamic matching cost aggregation method, we define the matching cost based on SSIM as Equation (4):

$$\text{cost}(p, u) = L(1 - \text{SSIM}(p, u)), \quad (4)$$

where $L = 255$ is the dynamic range of the pixel values for 8 bits/pixel greyscale images. Therefore, the maximum aggregation cost is less than 512 without penalty item by the semiglobal matching method. The purpose of transforming matching cost into integer type is to be more suitable for FPGA implementation. In this study, we limited the single matching cost to 255, which is represented by 8 bits. Therefore, the maximum matching cost is within a reasonable range in the semiglobal cost aggregation process, in which the maximum cost is limited to 10 binary digits in this system.

We set the penalty item to $P1 = 30$ and $P2 = 100$ in this system, which indicates an assumption that the disparity of adjacent pixels is smooth. In fact, the depth change is discontinuous in automatic driving, which leads to that the disparity fluctuates greatly and the precision decreases in the depth

discontinue pixel. In order to improve this problem, we propose to build a guide map to show where depth changes are possible [27]. Compared with the literature [27], we assume that the depth change always occurs at the edge of objects. In addition, the structure is obvious at the edge region, which makes it easier to get the real and accurate disparity.

First, we provide a one-dimensional Gaussian edge feature map (setting to $\mu = 0$ and $\sigma = 3$ in the system) to indicate where the obvious edge is and where is the stereo matching result is more credible. Then, the first derivative in the same direction is calculated to obtain the gradient response. Next, the same process is implemented in the tangent direction. Finally, the sum of the square root of two gradient responses is considered as a feature map, as shown in Figure 1.

Based on the feature map, the larger the response value is, the stronger the edge characteristics are, and the greater the possibility of depth change. According to the above method, we infer that the maximum aggregation cost between adjacent pixels is 610 ($255 + 100 + 255$) in one path. The sum cost in eight paths is less than 4880 (610×8), which is represented by 13 binary digits. Therefore, the maximum memory cost of a single pixel is limited within 16 binary digits (2 Bytes) by eight aggregation paths. To reduce the computing cost, we propose a stereo matching optimization method, which focuses on reducing the disparity search range. We construct the multiscale images by the image pyramid method. In this study, we propose three layers pyramid method. As shown in Figure 2, the large-scale layer images have the original resolution of 1280×720 . The middle-scale layer images are downsampled from the large-scale layer images, and the small-scale layer images are downsampled from the middle-scale layer images.

The structure features are obvious in the large-scale layer. Correspondingly, the remote details are completely preserved in the small-scale layer. At first, we propose to set the disparity search range as 16 pixels in the small-scale layer, which is equal to the search range of 64 pixels in the large-scale layer. Based on that, we calculate the small-scale disparity map by the SSIM algorithm in the small-scale layer. Then, we initialize the middle-scale disparity map by upsampling from the small-scale disparity map by the linear interpolation method. Because the small-scale disparity map has the same size as the small-scale layer, 320×180 , the size of initialized middle-scale disparity map is the same as that of the middle-scale layer, 640×360 . Next, we refine the middle-scale disparity map. Based on the initialized disparity map, we set the same search range as 16 pixels. However, the beginning of the search depends on the initialized disparity, in which the search range with 16 pixels is a symmetric interval centered on the initialized disparity. Finally, we repeat the above processes to obtain the large-scale disparity map.

At the calculation process of the multiscale stereo matching algorithm, the feature map is only adopted in the large-scale layer. The proposed stereo matching method is suitable for the FPGA to process:

- (i) There are many reusable operation modules because the rules and parameters are the same



FIGURE 1: An example of the feature map in stereo matching. Left: grey image. Right: feature map.

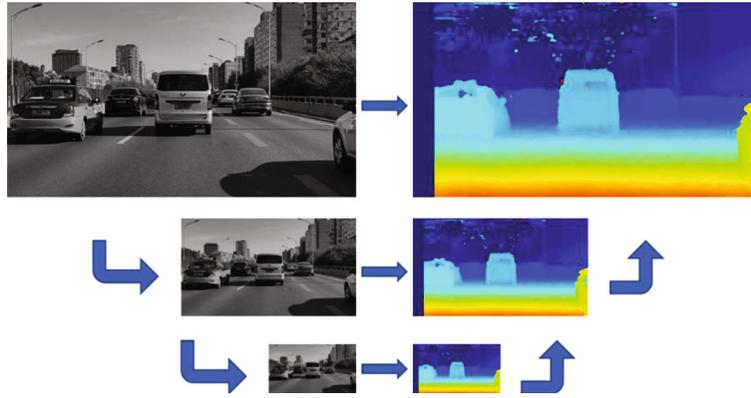


FIGURE 2: Multiscale images pyramid and disparity map. Top: the large-scale layer. Middle: the middle-scale layer. Bottom: the small-scale layer.

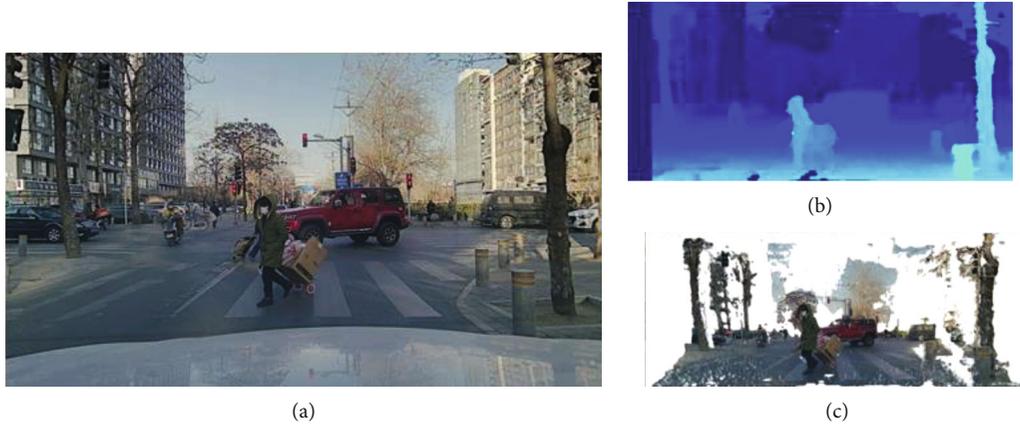


FIGURE 3: Disparity and 3D point cloud. (a) The left image. (b) Disparity map. (c) 3D point cloud.

- (ii) There is no need to cache all image data in memory at the same time, so the calculation process can be consistent with the data transmission process
- (iii) The algorithm is designed to process numerical multiplication and addition by a large number of fixed-point data

3.2. Road Prediction. In practice, we focus on two issues: (1) where is the obstacle and (2) what is the trend of road. We propose the grid projection method to predict the trend of

road. When obstacles occupy the road, the trend prediction will be hindered, but the freespace region should be correctly described. In this study, our method consists of three stages: grid projection, boundary search, and shape detection.

3.2.1. Grid Projection. Based on the disparity map, we calculate the 3D point cloud coordinate by Equation (5):

$$x = \frac{b(u - u_0)}{\text{disp}}, y = \frac{b(v - v_0)}{\text{disp}}, z = \frac{bf_{\text{eq}}}{\text{disp}} \quad (5)$$

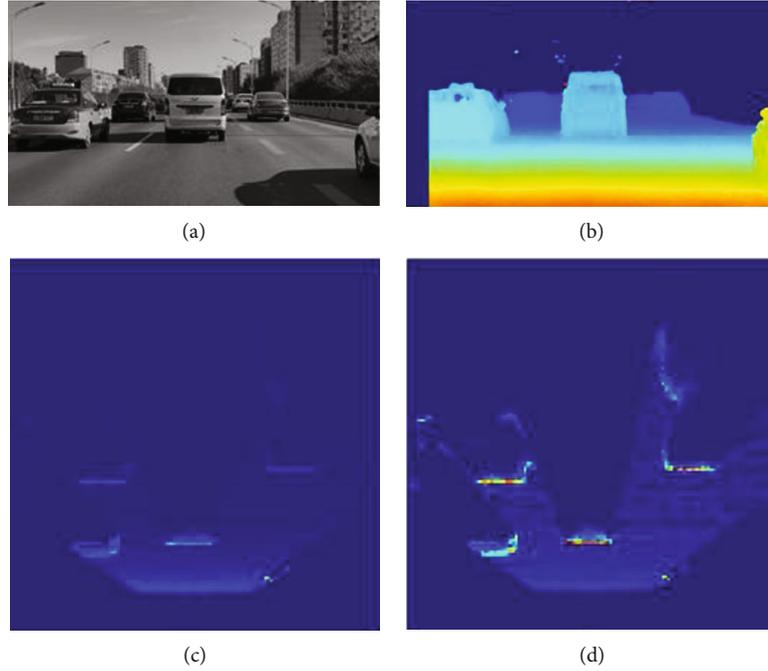


FIGURE 4: The process of probability grid map. (a) Left grey image. (b) Disparity map. (c) Grid projection. (d) Probability grid map.

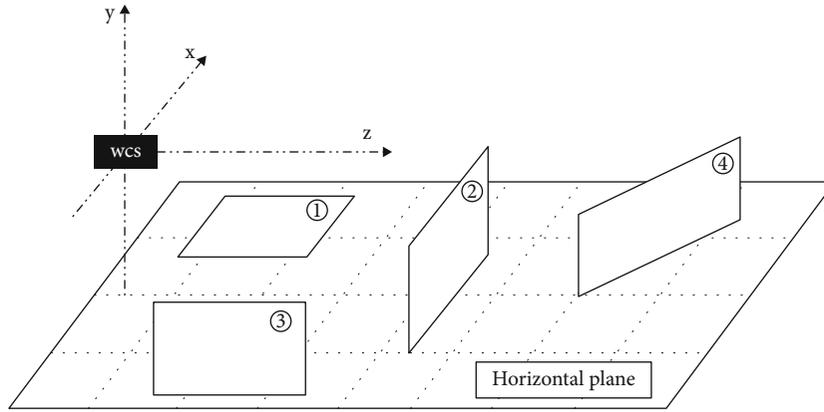


FIGURE 5: Road plane and three obstacle plane models in wcs.

where (x, y, z) is the 3D coordinate in the world coordinate system (wcs), (u, v) is pixel coordinate in image coordinate system (ics), (u_0, v_0) is the camera central point, disp is the disparity, b is the baseline of binocular camera, and f_{eq} is equivalent focal length. The disparity map and 3D points cloud are shown in Figure 3.

Inspired by pseudolidar data from visual [28], we build a projection space from bird eye view (BEV) in wcs. At BEV, the x -axis, which is parallel to the baseline of the binocular camera, represents the horizontal direction, and the y -axis, which is parallel to the optical axis of the camera and is positive horizontally to the front, represents the depth direction. In the projection space, the fixed size grid is divided as the basic projection unit, which the grid is a rectangle by setting $10 \text{ cm} \times 50 \text{ cm}$ in x -direction and z -direction. First, we divide point cloud data by grid mesh and drop out the outliers that points do not belong to the detection space $\Pi = \{(x, y, z) \mid y \in (-0.5, 2.5), z \in (0, 60)\}$. This constraint represents a priori

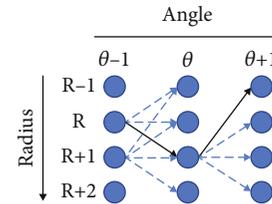


FIGURE 6: Directed acyclic graph with dynamic programming method. The black arrow indicates the best path.

assumption that the maximum height of the obstacles on road will not exceed 3 m, and the farthest observation distance is 60 m. Then, we count the number of points in each grid and set this value as the feature value of this grid. In this way, grids with lots of points cloud data have a larger accumulated number. As shown in Figure 4(c), we use color to represent the accumulated number, where the dark blue

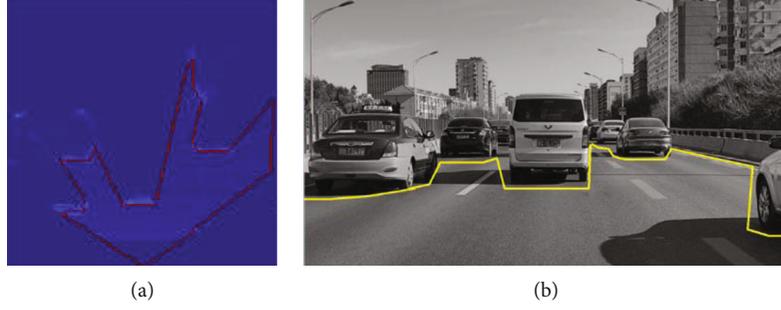


FIGURE 7: Freespace boundary. (a) Optimal path by DP. (b) Boundary in ics.

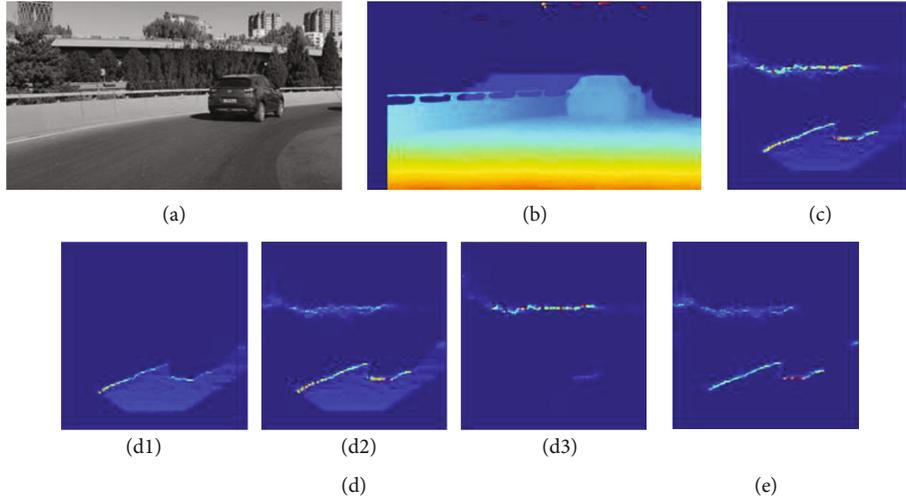


FIGURE 8: Multilayers grid projection. (a) Left image. (b) Disparity map. (c) PGM in Π . (d1) PGM in Π_1 . (d2) PGM in Π_2 . (d3) PGM in Π_3 . (d) WPGM in Π .

indicates zero and the brighter the color, the greater the accumulated number.

3.2.2. Boundary Search. Inspired by the stochastic occupancy grids method [29], grids with larger feature values represent that there is a greater possibility of an obstacle. On the contrary, grids with smaller feature values are more likely to represent the road region. In addition, grids with zero feature value indicate that it is occluded and cannot be observed.

By transferring the ics (disparity map) to vcs (grid projection), a unified physical scale is helpful to build a more robust mathematical model to solve the problem. In detection space Π , we consider the ground as a plane parallel to the xoz and obstacles as a plane vertical to the horizontal plane. As shown in Figure 5, obstacles can be approximated by three plane models in the detection space Π : perpendicular to the optical axis (marked ②), parallel to the optical axis and perpendicular to the horizontal plane (marked ③), and perpendicular to the horizontal plane and intersecting with the optical axis (marked ④). In addition, the road plane is another plane (marked ①) that parallel to the horizontal plane or intersecting at a smaller angle.

These plane models describe obstacles in different states. The plane ② represents obstacles perpendicular to the optical axis on the driving route, such as a vehicle in the same lane, which only the vehicle rear is visible. The plane ③ represents

obstacles parallel to the optical axis adjacent to the driving route, such as fences and barriers, which only the side is visible. The plane ④ represents obstacles that intersect the optical axis on the driving route, such as guardrails and walls in the curve road. In addition, plane ② and plane ③ are combined to describe adjacent discontinuous obstacles, such as cut-in vehicles, the vehicle side and rear are visible. Based on the above model, obstacles generate a large number of 3D point cloud data on its plane. Therefore, there are many accumulated points in the projection area of the obstacle plane in the grid projection.

Suppose that the basic detection unit on planes is a square, which represents that the obstacle is divided into many square units. In practice, the side length of the unit is consistent with the grid projection size in x -direction. At different distances, the number of point clouds generated by the detection unit conforms to the Equation (6):

$$N_{\text{unit}} = \left(\frac{a \times f}{z \times \mu} \right)^2, \quad (6)$$

where N_{unit} represents the number of 3D points in unit, a is the side length of unit ($a = 10$ cm in this study), f is the focal length of lens, μ is the pixel size of sensor, and z is distance in wcs ($0 \text{ m} < z \leq 60 \text{ m}$ in this study).

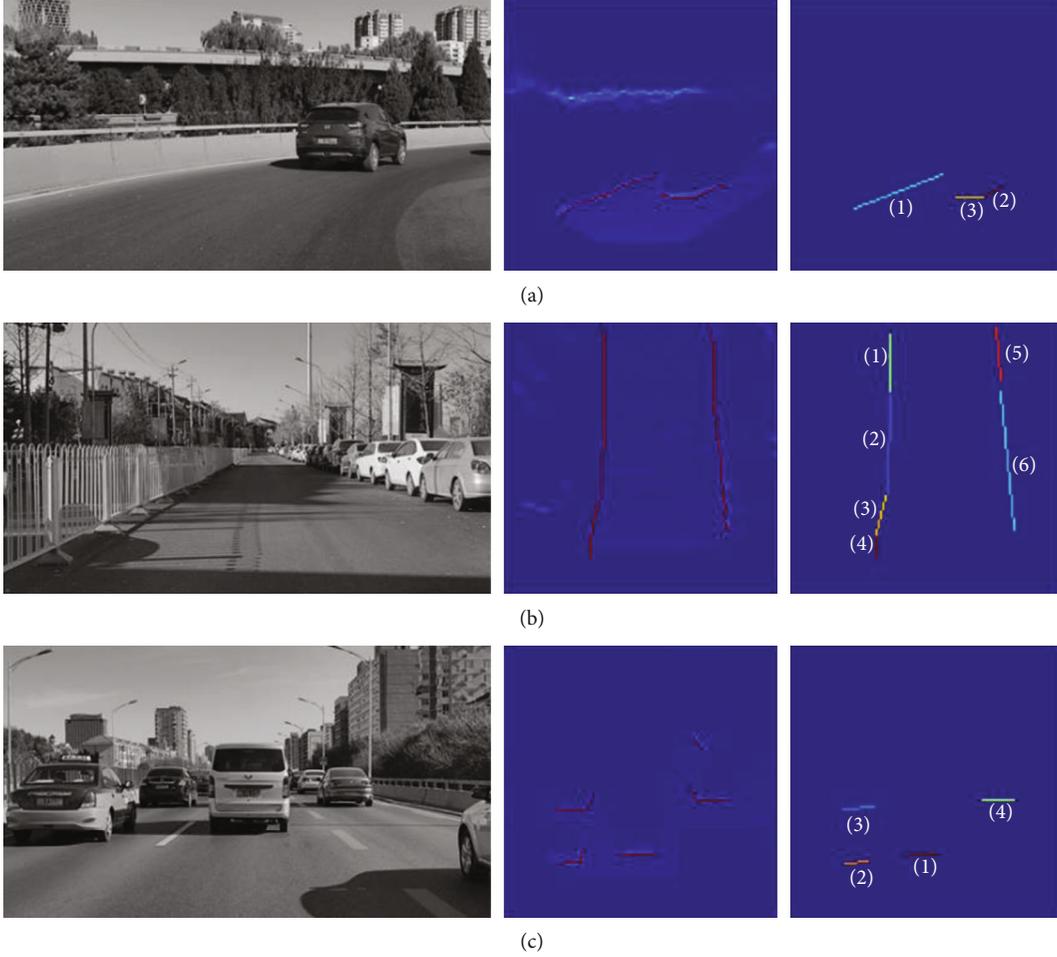


FIGURE 9: The process of shape detection. (a) In curved road. (b) In straight road. (c) With obstacles.

In the detection space Π , the height constraint is less than h m. Therefore, the maximum of 3D points is $h \times N_{\text{unit}}/a$ in a grid. Furthermore, we construct a probabilistic occupancy grid, where we divide the accumulated number by the maximum. As shown in Figure 4(d), we use the color to indicate the probability. The dark blue represents low probability and red represents high probability. In polar coordinates, we transform the probability grid map (PGM) into a graph model, where the nodes represent elements in PGM with fixed angle resolution and radius resolution. The value of the node is the probability of the grid, the column index is angle, and the row index is radius. Therefore, the road boundary search problem is transformed into a dynamic programming problem to find the path with maximum probability. We consider the graph as a directed acyclic graph (DAG) that the direction is from left to right or opposite. In theory, the two directions have the same path. In this study, we propose a semiglobal optimization method to solve the dynamic programming problem.

As shown in Figure 6, the node represents the probability of the grid. The horizontal axis is the angle of the field of view, where the resolution is 1 degree. The vertical axis is the radius, where the resolution is the same as that of the grid

in z direction. The FOV and grid projection space Π are not completely coincident, so the number of nodes in each column is not consistent. We propose a semiglobal energy function in Equation (7).

$$E(p | p \in N) = P_{p-1}(\theta, R) + T(p-1, p) + P_p(\theta, R), \quad (7)$$

where N is the best path that the summary of probability is maximum. $P_p(\theta, R)$ represents the probability of $p(\theta, R)$, and this point belongs to the best path ($p \in N$). $T(p-1, p)$ represents the penalty term between adjacent column nodes, in which the $p-1$ is previous nodes and p is current nodes. The penalty term is defined as Equation (8).

$$T(p-1, p) = \begin{cases} 0 & |R(p-1) - R(p)| = 0 \\ T_1 & |R(p-1) - R(p)| \leq 1 \\ T_2 & \text{otherwise} \end{cases} \quad (8)$$

The penalty term represents an assumption that the path between adjacent columns is smooth in DAG. $R(p-1)$ is the radius index of node in the previous column and $R(p)$ is in

current column. Obviously, the penalty term is constrained to $-1 \leq T_2 < T_1 < 0$. Finally, the optimization function is defined as Equation (9).

$$N(p) = \arg \min \sum E(p). \quad (9)$$

The purpose of parameter setting is that the row coordinate changes of adjacent column nodes should be smooth. In this study, we set the $T_1 = -0.1$ and $T_2 = -0.3$. As a result, we obtain an optimal path in DAG, where the sum of probability exists maximum. We convert the node index into the coordinate of PGM, where grids are considered as boundary points of freespace in wcs. In addition, the boundary points in wcs are back projected into ics through Equation (6). As shown in Figure 7, the red line in the probability grad map is the optimal path and the yellow line is the freespace boundary in ics.

In practice, we employ the multilayer grid projection to overcome that case where the near obstacle is lower than the far obstacles. Three layers are divided by height with $-0.5 \text{ m} \sim 0.5 \text{ m}$ (subspace Π_1), $0.5 \text{ m} \sim 1.5 \text{ m}$ (subspace Π_2), and $1.5 \text{ m} \sim 2.5 \text{ m}$ (subspace Π_3). As shown in Figure 8(a), the height of the near barrier is lower than the far greenbelt, so that the probability of barrier is smaller than the probability of greenbelt in Figure 8(c). We propose a weight energy function as Equation (10).

$$E(p | p \in N) = \omega_{p-1} P_{p-1}(\theta, R) + T(p-1, p) + \omega_p P_p(\theta, R), \quad (10)$$

where ω_p is the weight of node p that the coordinate of this point is (θ, R) and $0 < \omega \leq 1$. The $p-1$ indicates the previous node. The capital P indicates the penalty item. The ω in Π_1 is greater than ω in Π_2 , and ω in Π_2 is greater than ω in Π_3 . In this study, we set weight as $\omega = 1$ in Π_1 , $\omega = 0.6$ in Π_2 , and $\omega = 0.3$ in Π_3 . As a result, the Figure 8(e) shows the weight probability grid map (WPGM) that the best path corresponds to our expectations, where the high brightness represents a large probability.

3.2.3. Shape Detection. Based on the three obstacle models, we detect geometric constraints on the multilayer grid projection. In practice, we propose to identify the geometry feature by prior. The plane ②, in Figure 5, is a segment parallel to the x -axis in the WPGM, the plane ③ is another segment parallel to the z -axis, and the plane ④ is an oblique line or curve. Therefore, we detect these geometry features on the boundary to identify if there any obstacle exists and road environment, for example, vehicle is going to drive into straight road or curved road.

To reduce the noise, we discard the grid whose probability is less than 0.1. The result is shown in Figure 9. The left column is the grey images, the middle column is the feature points in WPGM, and the right column is the result of shape detection.

In the middle column of Figure 9, the red points are feature points that make up the shape feature map (SFM). These feature points are used to progressive probabilistic Hough

TABLE 1: Describe the road environment by state model.

	Curve road	Straight road	Obstacles
(a)	1	0	1
(b)	0	6	0
(c)	0	0	4

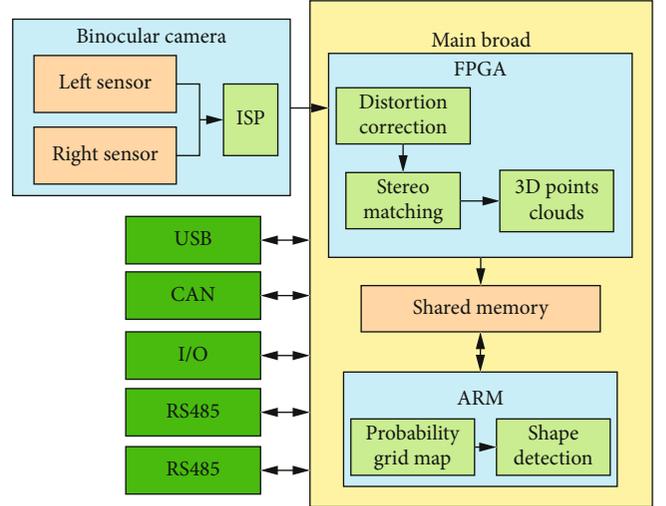


FIGURE 10: System hardware platform.

TABLE 2: System hardware parameters.

Items	Parameters	Items	Parameters
Baseline	120 mm	Focal length	8.26 mm
Pixel size	$4.2 \times 10^{-3} \text{ mm}$	Resolution	1280×720
Horizontal FOV	30°	Vertical FOV	22°

transform, in which the coordinates use pixel dimensions in the SFM. For example, in the right column of Figure 9(a), the different color line represents different slopes: the slope of line (1) is 0.3862, the slope of line (2) is 0.5062, and the slope of line (3) is 0.0082. We consider the line (3) is modelled as the plane ②, while the line (1) and line (3) are modelled as plane ④. However, the plane ② is usually considered to be an independent obstacle, such as a vehicle. Because line (2) and line (3) are connected, they are regarded as independent obstacle in the wcs. In addition, plane ④ is usually considered continue obstacle, such as guardrails and walls in the curve road. The line (1) is regarded as a guardrails or walls in the curve road.

We describe the shape feature by different plane models so that the complex road environment is classified into finite-state models. Table 1 describes Figure 9. The (a) scene is described as 1 curve road model and 1 obstacle model, the (b) scene is described as 6 straight road models, and the (c) scene is described as 4 obstacle models. Therefore, the (a) scene is predicted to be going to drive into curve road and there is an obstacle. The (b) scene is predicted to drive on a straight road without obstacle. The (c) scene is predicted to

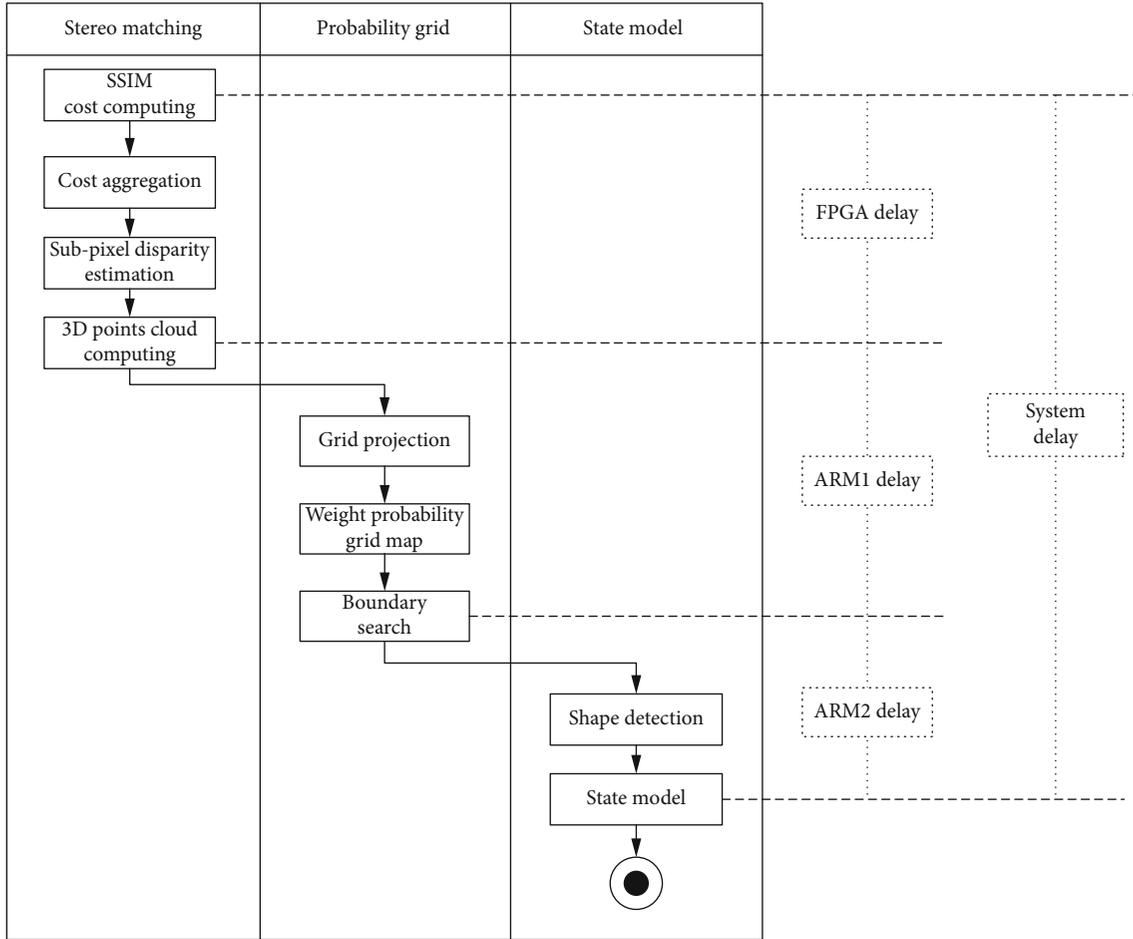


FIGURE 11: System software flow framework.

that there are many obstacles in front of the vehicle and other road information is invalid.

4. System Design

4.1. Hardware Architecture. The proposed system is used for forward sensing of automatic driving or advanced driver assistance system (ADAS). In practice, the system is generally installed in the narrow space between the windshield and the inside rearview mirror. It requires that the space volume of the hardware system must be small. The system hardware design architecture is shown in Figure 10.

The lens focal length is 8.26 mm and the image resolution is 1280×720 . The embedded computing core is a Xilinx Z-7020 system on chip (SoC) that includes an Artix-7 field-programmable gate array (FPGA) and two A9 advanced RISC machines (ARM). Based on this hardware platform, the system's full load power is 6 W without any cooling system. In addition, its volume is only $17 \text{ cm} \times 8 \text{ cm} \times 4 \text{ cm}$. Other system parameters are shown in Table 2.

4.2. Software Architecture. In the algorithm function, we divide three parts: stereo matching, probability grid, and state model, as shown in Figure 11. The stereo matching module

TABLE 3: Evaluation of stereo matching in KITTI.

Method	Accuracy (%)	Efficiency
Our proposed	93.22	1.00
Origin MPV	94.43	3.05
SGBM	92.36	—
ELAS	91.76	—

includes SSIM cost computing, cost aggregation, subpixel disparity estimation, and 3D point cloud computing. The probability grid module includes grid projection, weight probability grid map, and boundary search. The state model module includes shape detection and state module.

The three modules are processed by parallel computing on three processing units. The frequency of data processing depends on the slowest one of the three modules, and the system delay is equal to the sum of the three modules. In the proposed system, the FPGA delay is 66 ms, the ARM1 delay is 16 ms, and the ARM2 delay is 21 ms. Therefore, the frequency of the system is 15 fps and the system delay is 103 ms.

TABLE 4: Evaluation of stereo matching in private dataset.

Test case	Sunny	Cloudy	Night	Backlight
Accuracy (1.0)	78.94%	80.06%	58.68%	78.49%
Accuracy (0.5)	51.00%	51.33%	32.47%	50.11%
Accuracy (0.3)	33.84%	32.42%	19.54%	31.62%
Accuracy (0.1)	11.97%	11.13%	6.66%	10.84%
Convergence (pixel)	4.66	4.15	5.73	5.37

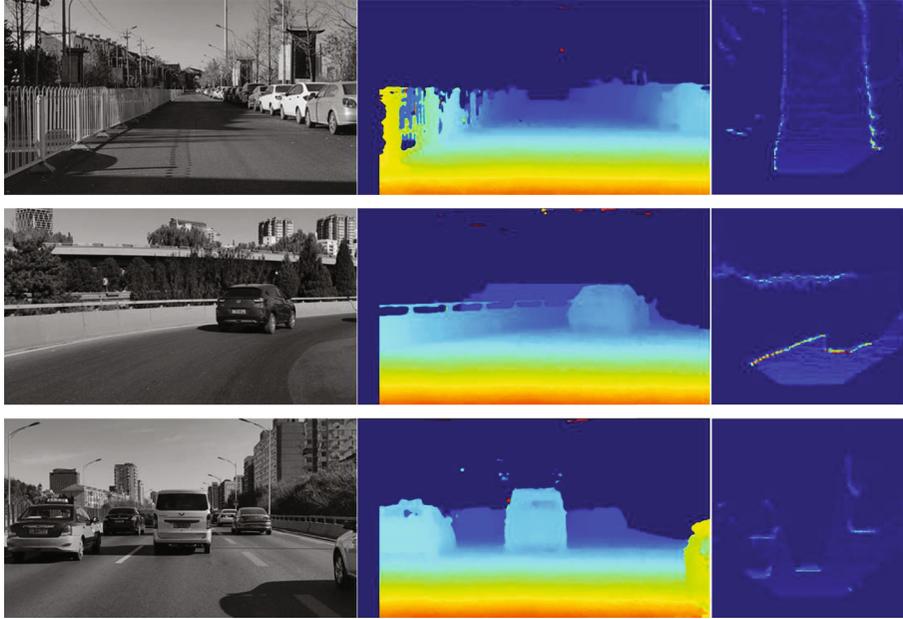


FIGURE 12: The WPGM in different road environment. Grey images, disparity space, and weight probability grid map.

5. Experiment and Analysis

5.1. Stereo Matching. We evaluate our stereo matching method by two comparative experiments: efficiency and accuracy. By compared on KITTI dataset [30] as benchmark data set, our multiscale stereo matching algorithm (93.22%) is lower average error rate than classical unsupervised stereo matching algorithm, such as SGBM [9] (92.36%), ELAS [31] (91.76%), and origin MPV algorithm [10] (94.43%). The accuracy of our method is higher than SGBM and ELAS. Although our method is lower than the origin MPV algorithm in accuracy, its computing efficiency is higher than the origin MPV algorithm. The detail is shown in Table 3. Based on the running times of our proposed algorithm as the benchmark, the efficiency is defined as the running time of the comparison algorithm divided by the benchmark. Therefore, the efficiency of our algorithm is 1.

In addition, we test the multiscale stereo matching method in the private dataset, where we focus following items: subpixel disparity accuracy, light condition, and point cloud distribution.

- (i) *Subpixel Disparity Accuracy.* When the absolute value of disparity error is less than 1.0 (0.5, 0.3, 0.1) pixel, the matching disparity is correct. The

TABLE 5: Evaluation of freespace detection in private dataset.

Method	Running time (ms)	Recall (%)
Hautiere et al. [33]	30	95.0
Xin et al. [32]	18360	91.7
Ours	16	93.5

index is defined as the corrected number divided by the total pixels

- (ii) *Light Condition.* The disparity accuracy is evaluated by the different light condition, such as sunny, cloudy, night, and backlight
- (iii) *Convergence.* Assuming the distribution of point clouds being Gaussian model, disparity's variance represents the convergence degree of the point clouds

The result is shown in Table 4. The subpixel disparity accuracy represents a more precise ability of the stereo matching algorithm, which the large proportion in high precision range indicates the disparity estimation algorithm is close to ground truth (GT). The proposed multiscale stereo matching method has the highest precision in sunny and

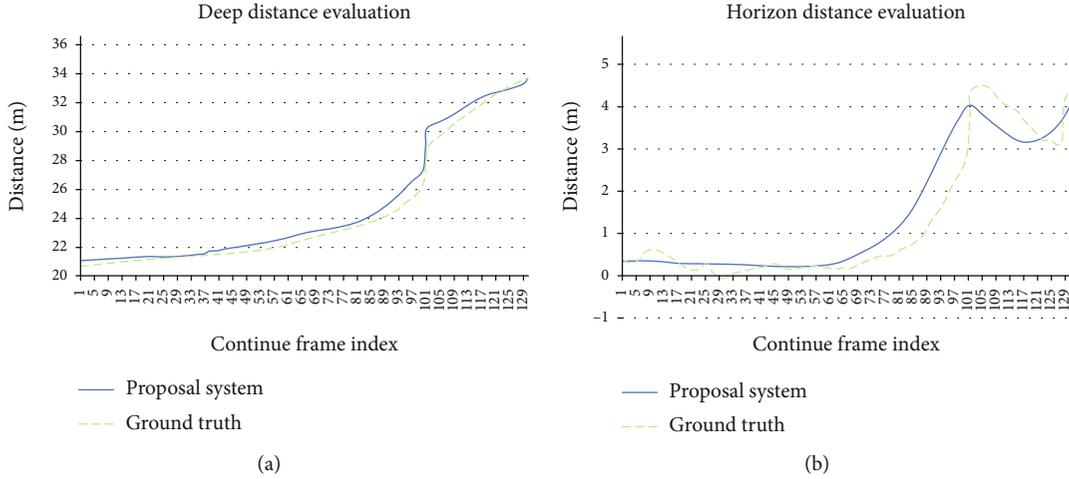


FIGURE 13: Obstacle prediction in deep and horizon distance.

TABLE 6: Evaluation of state model prediction in private dataset.

	Curve road	Straight road	Obstacle	Precision
Curve road	293	7	4	96.38%
Straight road	1	591	0	99.83%
Obstacle	5	2	96	93.20%
Recall	97.67%	98.50%	96.00%	—

cloudy, because the imaging quality is best in this weather. The edge will become unclear due to overexposure in backlight, and the texture features will be blurred in dark environment, such as at night. Therefore, the subpixel disparity accuracy decreases slightly in the backlight, but significantly at night. In addition, the variance indicates the aggregation degree of disparity on the same obstacle plane. The small variance represents that the disparity estimation is stable and the point clouds converge in space. The experimental results show that the proposed method is stable and robust under good light conditions.

Figure 12 shows middle processing to explain our multi-scale stereo matching algorithm and weight probability grid map. The left column is grey images, the middle column is a disparity map corresponding to points cloud space, and the right column is a weighted probability grid map corresponding to points cloud space.

5.2. Road Prediction. We evaluate the proposed system by following experiments: freespace detection, obstacle prediction, road environment prediction, and system performance.

Based on the private dataset, we label the GT of freespace on images. We evaluate the freespace by recall method, in which the index represents the similarity between the detection result and GT. Recall = 100% indicates that the detection result is completely consistent with the true value, while IoU=0 indicates completely inconsistent. The result is shown in Table 5, where our system is compared by two classic methods to show that our method is advanced in efficiency and result. Our system is more efficient than Xin et al. [32] method in running time and recall. Compared to

Hautiere et al. [33] method, we have halved our running time but lost only 1.5% of recall. The experiment shows that our system is robust and stable in freespace detection.

The obstacle prediction is one of the tasks based on the state model. We evaluate the results of obstacle prediction by the number and location of obstacles based on our private dataset. The GT of obstacles is labelled manually, where we focus on independent objects such as vehicles and pedestrians. We count the number of obstacles in the state model to compare with GT, so the average recall rate of obstacle detection of obstacles is 98.3% in private dataset. As shown in Figure 13, the left shows the deep distance and the right shows the horizon distance, where the deep distance is represented by the vertical axis and horizon distance is represented by the horizontal axis on WPGM. We employ the root mean squared error (RMSE) to evaluate the error of relative distance by Equation (11):

$$\text{RMSE} = \frac{1}{n} \sqrt{\sum_{i=1}^n (\text{data}_i - \text{GT}_i)^2}, \quad (11)$$

where data_i is observations and n is the total number of test dataset. The RMSE in the deep distance is 0.42 m and 0.46 m in the horizon distance. The experiment shows that our method is robust and stable in obstacle prediction.

We evaluate the state model by precision and recall as Table 6, where the first row represents the GT, and the first column represents the observations. We provide 1000 images including 600 straight road state, 300 curve road state, and 100 obstacle state. Precision and recall index are employed

TABLE 7: Running time of modules.

Hardware unit Module	FPGA			ARM1		ARM2	
	Stereo matching	Point cloud	WPGM	Path planning	Shape detection	State model	
Running time	55 ms	11 ms	12 ms	4 ms	20 ms	1 ms	
Delay time	66 ms			16 ms		21 ms	

TABLE 8: Chip temperature test.

Ambient temp (°C)	Chip temp (°C)	Load power (W)
-15	40~50	6
15	50~60	6
45	60~75	6
65	75~90	6

to evaluate state model prediction. The experiment shows that our system is sensitive to road environment prediction.

In addition, we test the system performance by running time and power cost. As shown in Table 7, we record the running time of three hardware units. The largest time cost is stereo matching and point cloud in FPGA, where its running time is 66 ms. The frame rate of the system is decided by this module to 15 fps. The system delay time is a summary of three modules, which is 103 ms.

Finally, we test the power cost and the chip temperature. Our system is installed between the front windshield and the rear-view mirror, so it must meet the requirements of GB/T 28046.4, which stipulates that the maximum operating temperature of the system shall not exceed 90°C. Table 8 shows that the maximum temperature in full load power meets the standard requirement.

6. Conclusions

In this study, we propose a low-power road environment prediction system, which the proposed system consists of a binocular camera and a low-power computing unit. Our contribution includes three points as follows. Firstly, a multi-scale stereo matching algorithm is proposed for hardware computing. Next, we propose a weighted probability grid map-based points cloud. Finally, the plane model and state model are proposed to describe the road environment. Our work proves that the existing technology achieves the function requirement under the low-power constraint. Experiments show that the proposed system is robust and sensitive to road environment prediction and the performance meets the mandatory standards in practice. In future work, our study provides a benchmark for obstacles recognition and path planning.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Key R&D Program of China (Grant No. 2018AAA0103103), the Science and Technology Development Fund, Macao SAR (no. 0024/2018/A1), and the Research Fund of Guangdong-Hong Kong-Macao Joint Laboratory for Intelligent Micro-Nano Optoelectronic Technology (No. 2020B1212030010).

References

- [1] Y. Shi, Q. Li, S. Bu, J. Yang, and L. Zhu, "Research on intelligent vehicle path planning based on rapidly-exploring random tree," *Mathematical Problems in Engineering*, vol. 2020, 14 pages, 2020.
- [2] M. Shahjalal, M. Hossan, M. Hasan, M. Z. Chowdhury, and J. Y. M. Le NT, "An implementation approach and performance analysis of image sensor based multilateral indoor localization and navigation system," *Wireless Communications and Mobile Computing*, vol. 2018, 13 pages, 2018.
- [3] L. Claussmann, M. Revilloud, D. Gruyer, and S. Glaser, "A review of motion planning for highway autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 5, pp. 1826–1848, 2019.
- [4] Z. Wang and H. Zhang, "A novel approach for free space detection using u-disparity and dynamic programming," in *2016 5th International Conference on Computer Science and Network Technology (ICCSNT)*, pp. 391–395, Changchun, 2016.
- [5] X. Zhang, J. Lai, D. Xu, H. Li, and M. Fu, "2d lidar-based slam and path planning for indoor rescue using mobile robots," *Journal of Advanced Transportation*, vol. 2020, 14 pages, 2020.
- [6] K. Zhou, X. Meng, and B. Cheng, "Review of stereo matching algorithms based on deep learning," *Computational Intelligence and Neuroscience*, vol. 2020, 12 pages, 2020.
- [7] C.-H. Hsu, S. Wang, Y. Zhang, and A. Kobusinska, "Mobile edge computing," *Wireless Communications and Mobile Computing*, vol. 2018, 3 pages, 2018.
- [8] S. Raza, S. Wang, M. Ahmed, and M. R. Anwar, "A survey on vehicular edge computing: architecture, applications, technical issues, and future directions," *Wireless Communications and Mobile Computing*, vol. 2019, 19 pages, 2019.
- [9] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, 2007.
- [10] Q. Xie, Q. Long, and S. Mita, "Integration of optical flow and multi-path-viterbi algorithm for stereo vision," *International Journal of Wavelets, Multiresolution and Information Processing*, vol. 15, no. 3, article 1750022, 2017.
- [11] Z. Hu and K. Uchimura, "Uv-disparity: an efficient algorithm for stereovision based scene analysis," in *IEEE Proceedings. Intelligent Vehicles Symposium*, pp. 48–54, Las Vegas, NV, USA, 2005.

- [12] A. Harakeh, D. Asmar, and E. Shamma, "Ground segmentation and occupancy grid generation using probability fields," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 695–702, Hamburg, Germany, 2015.
- [13] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: an open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.
- [14] K. Tateno, F. Tombari, I. Laina, and N. Navab, "Cnn-slam: real-time dense monocular slam with learned depth prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6243–6252, Honolulu, HI, USA, 2017.
- [15] M. Humenberger, C. Zinner, M. Weber, W. Kubinger, and M. Vincze, "A fast stereo matching algorithm suitable for embedded real-time systems," *Computer Vision and Image Understanding*, vol. 114, no. 11, pp. 1180–1202, 2010.
- [16] Q. Yang, "A non-local cost aggregation method for stereo matching," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1402–1409, Providence, RI, USA, 2012.
- [17] K. Zhang, Y. Fang, D. Min et al., "Cross-scale cost aggregation for stereo matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1590–1597, Columbus, OH, USA, 2014.
- [18] W. Mao, M. Wang, J. Zhou, and M. Gong, "Semi-dense stereo matching using dual cnns," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1588–1597, Waikoloa, HI, USA, 2019.
- [19] L. Qu, K. Wang, L. Chen, Y. Gu, and X. Zhang, "Free space estimation on nonflat plane based on v-disparity," *IEEE Signal Processing Letters*, vol. 23, no. 11, pp. 1617–1621, 2016.
- [20] H. Deilamsalehy and T. C. Havens, "Sensor fused three-dimensional localization using imu, camera and lidar," in *2016 IEEE Sensors*, pp. 1–3, Orlando, FL, USA, 2016.
- [21] Z. Xiao, B. Dai, H. Li et al., "Gaussian process regression-based robust free space detection for autonomous vehicle by 3-d point cloud and 2-d appearance information fusion," *International Journal of Advanced Robotic Systems*, vol. 14, no. 4, article 1729881417717058, 2017.
- [22] L. Zheng, Y. Zhu, B. Xue, M. Liu, and R. Fan, "Low-cost gps-aided lidar state estimation and map building," in *2019 IEEE International Conference on Imaging Systems and Techniques (IST)*, pp. 1–6, Abu Dhabi, United Arab Emirates, 2019.
- [23] Y. Cong, C. Chen, J. Li, W. Wu, S. Li, and B. Yang, "Mapping without dynamic: robust lidar-slam for ugv mobile mapping in dynamic environments," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 43, pp. 515–520, 2020.
- [24] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *The journal of Machine Learning Research*, vol. 17, no. 1, pp. 2287–2318, 2016.
- [25] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, vol. 2, pp. 1398–1402, Pacific Grove, CA, USA, 2003.
- [26] W. Lai, J. Huang, Z. Hu, N. Ahuja, and M.-H. Yang, "A comparative study for single image blind deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1701–1709, Las Vegas, NV, USA, 2016.
- [27] M. Park and K. Yoon, "Learning and selecting confidence measures for robust stereo matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 6, pp. 1397–1411, 2018.
- [28] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell, and K. Q. Weinberger, "Pseudo-lidar from visual depth estimation: bridging the gap in 3d object detection for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8445–8453, Long Beach, CA, USA, 2019.
- [29] H. Badino, U. Franke, and R. Mester, "Free space computation using stochastic occupancy grids and dynamic programming," in *Workshop on Dynamical Vision, ICCV*, vol. 20, Rio de Janeiro, Brazil, 2007.
- [30] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, Providence, RI, USA, 2012.
- [31] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Asian Conference on Computer Vision*, pp. 25–38, Berlin, Heidelberg, 2010.
- [32] L. Xin, J. Song, Y. Chen, and J. Hu, "Robust free-space detection in urban roads based on msr extraction using gradient images," in *2018 37th Chinese Control Conference (CCC)*, pp. 4141–4146, Wuhan, 2018.
- [33] N. Hautière, J. P. Tarel, H. Halmaoui, R. Brémond, and D. Aubert, "Enhanced fog detection and free-space segmentation for car navigation," *Machine Vision and Applications*, vol. 25, no. 3, pp. 667–679, 2014.