

Research Article

The Construction Method of the Digital Operation Environment for Bridge Cranes

Keyuan Zhao ¹, Qicai Zhou,¹ Xiaolei Xiong,^{1,2} and Jiong Zhao¹

¹School of Mechanical Engineering, Tongji University, Shanghai 201804, China

²Department of Mechanical and Automotive Engineering, Tongji Zhejiang College, Jiaxing 314000, China

Correspondence should be addressed to Keyuan Zhao; zhaokayuan@tongji.edu.cn

Received 27 February 2021; Revised 6 April 2021; Accepted 23 April 2021; Published 4 May 2021

Academic Editor: Abdul Qadeer Khan

Copyright © 2021 Keyuan Zhao 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.

To realize real-time detection of the operating environment of bridge cranes, this paper presents a three-dimensional mapping method using a binocular camera and laser ranging sensor. First, the left and right images are obtained by using the binocular camera, and the block matching is used for stereo correspondence to obtain the disparity image. Then, according to the reprojection matrix and the disparity image, the depth value of each pixel is obtained, and the depth image can be transformed into a three-dimensional point cloud of the current frame. Relying on the camera position data obtained by the laser ranging sensor, the three-dimensional point cloud generated from different positions is registered to obtain the three-dimensional point cloud of the crane's global operating environment. Finally, an octree map is used to represent the operating environment. To simulate the construction process of the operating environment of the bridge crane, an experimental platform was built, and experiments were carried out using the method proposed in this paper. The experimental results verified the feasibility of the method proposed in this paper.

1. Introduction

With the popularity of industrial automation, traditional manually operated bridge cranes can no longer meet the needs of unmanned and intelligent workshops. To solve this problem, relevant researchers have carried out research on various key technologies of intelligent operation or unmanned operation of bridge cranes, such as automatic spreader, hook antisway, operation status monitoring, and online fault diagnosis. However, if the bridge crane does not have real-time detection capabilities for its operating environment and only operates on a preset fixed route, it cannot sense the location and size of obstacles in the workshop and cannot perform obstacle avoidance behavior. This problem seriously affects the intelligent degree of bridge cranes.

In order to realize the independent operation of the bridge crane and have the ability to detect the operating environment, the real-time mapping algorithm needs the following contents:

- (1) Ordinary computing power: the onboard computer provides ordinary computing power. Onboard computers generally use industrial control computers with special features such as shock resistance and dust resistance. Such computers generally have ordinary computing power. Therefore, the algorithm can obtain real-time and high-precision calculation results with limited computing power.
- (2) Hardware cost: due to production cost constraints, it is unrealistic to use expensive sensors such as multiline lidar.
- (3) Map saving: bridge cranes work in a fixed and large range, so the saving format of the map needs to have a high compression rate.
- (4) Special mapping scene: the mapping scene of the unmanned car is that the camera movement direction is perpendicular to the phase plane to construct a 3D or 2.5D map. The AGV and the sweeping robot build a flat two-dimensional map. The mapping scene of the

bridge crane is that the camera moves in the x -axis direction and y -axis direction parallel to the phase plane, and the depth of the obstacle is estimated to build a three-dimensional map. Different from unmanned vehicles, the working place of the bridge crane is fixed within the workshop, so it can obtain its absolute position through fixed markings or laser ranging like some AGV positioning methods.

This paper starts from the foundation to build an environment perception method and proposes a binocular 3D mapping system for bridge cranes. The main contributions are as follows:

- (1) An algorithm is proposed, in which the camera moves in a horizontal plane and constructs a three-dimensional scene (as shown in Figure 1) from the top view perspective. The algorithm can solve the problem of three-dimensional mapping in the operating environment of bridge cranes.
- (2) Binocular camera is used as the main sensor of three-dimensional mapping of the bridge crane. The binocular camera has the advantages of low cost, long detection distance, and high resolution. Compared with other types of sensors, it is more suitable for the needs of bridge cranes for 3D mapping.
- (3) Laser positioning: this paper selects the laser ranging sensor, combined with the characteristics of the bridge crane, to measure the camera position. Two laser ranging sensors are used to measure the position of the cart and the trolley, respectively, to obtain the position of the binocular camera. Compared with other positioning methods, it has the advantages of less computing resources, low hardware cost, long measurement distance, and high accuracy.
- (4) Map compression: in this paper, an octree map is selected as the final saving form of the map, which greatly reduces the saving space of the map and facilitates the transmission and saving of the mapping results.
- (5) Real time: in the case of maintaining a certain accuracy, the algorithm with fast operation speed is used. Thus, the three-dimensional mapping method proposed in this paper can be real time on a common hardware platform.

Because the method proposed in this paper is aimed at a special object, namely, bridge crane, it is difficult to find a public data set suitable for 3D mapping scenes of bridge cranes. Therefore, we build an experimental platform according to the operating characteristics of bridge cranes. Experiments are carried out using the method proposed in this paper, the experimental results are analyzed, and the reasons for the errors are discussed.

2. Related Works

2.1. Machine Vision for Cranes. Compared with robots, the research and development of stereo vision in bridge cranes is relatively late, and some researchers have begun to explore it

in recent years. To realize the automatic operation of the bridge crane, the researchers have made some research studies. Most of them focus on the detection of the swing angle of the hook, and there are relatively few research works on the construction of the operating environment for crane navigation.

Osumi uses a camera to identify obstacles while marking them to prevent overshooting of the moving objects [1]. However, no research was carried out without marking, and no three-dimensional map of the operating environment was established. Wang and Chen designed the crane robot in the laboratory environment to realize the movement of the cart, the trolley, and the hook in the three coordinate directions, and used the binocular camera to realize the visual positioning and measurement functions of the target [2–4]. Their work has made a useful exploration for the unmanned control of bridge cranes, but their research content is limited to digital image target recognition, and they failed to carry out research in the direction of stereo vision. Ling proposed a moving object tracking strategy using machine vision and applied it to cranes. Yoshida [5] and Smoczek and Hyla [6] used a 3D camera as a sensor for antisway control. Szpytko used a binocular camera to obtain a three-dimensional map of the left and right image strokes of a single frame and discussed the accuracy of the map [7]. However, he failed to study the multiframe binocular image, which made the crane only have the three-dimensional condition of the object within the range of a single frame of the digital image. Xiaobo proposed a container crane automatic loading and unloading system that uses machine vision to realize the identification and positioning of containers, and uses barcode location technology and motor absolute encoders to realize the movement positioning of various operating mechanisms of the container crane [8]. Kim proposed a method of using machine vision to recognize steel coils [9]. The research results of Xiaobo and Kim realize the application of machine vision in the crane operation process and have made remarkable achievements in computer vision in crane obstacle avoidance, but the research objects are containers and steel coils with standardized dimensions, and the workplace is neatly stacked. For bridge cranes in workshops with many debris and different sizes and shapes of objects to be handled, the environment is more complex. Xiaobo's research results can provide a reference but cannot be applied to the operating environment of bridge cranes. Chen [10] and Cho [11] used a wide-angle camera and lidar to fuse to establish a working scene for a mobile crane. The research results provide a great reference for the three-dimensional mapping of the operating environment of bridge cranes. However, bridge cranes are different from mobile cranes. Bridge cranes work in fixed places. How to use this feature is worthy of further study in this paper. The problem of map compression in large scenes is not considered in Chen and Cho's study. And they did not consider the compression problem of large-scale scene maps.

2.2. Machine Vision for Robots. Endres uses an RGB-D camera to build a high-precision three-dimensional map, which can be applied to small domestic robots [12]. Henry

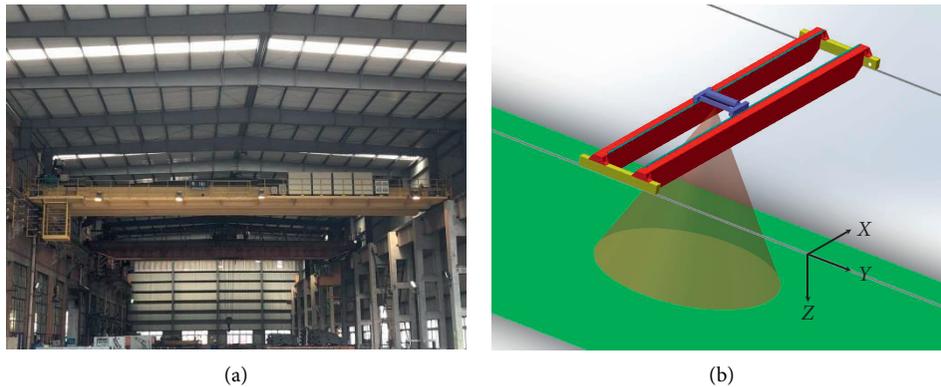


FIGURE 1: Three-dimensional mapping scene of the bridge crane.

used RGB-D cameras to construct a dense three-dimensional map of the indoor environment. A full 3D mapping system based on visual features and shape-based alignment combined with joint optimization algorithms is proposed [13]. Ryde proposed a multiresolution algorithm to align the 3D distance data stored in the occupied voxel list to facilitate the construction of 3D maps [14]. Smith used lidar to build a three-dimensional map [15]. Saarinen proposed a three-dimensional representation method for online real world maps based on two known representation methods: normal distribution transformation map and occupancy grid map [16]. Hornung proposed an open source framework to generate 3D environment models. The map is built based on octree and probabilistic occupancy estimation. An octree map compression method is proposed [17]. Schule uses a binocular camera to use 2.5D occupancy grid mapping method at night [18]. Yu proposed a two-dimensional occupancy grid map construction method based on stereo vision [19]. Brand introduced local obstacle maps that can be directly used for rapid local obstacle avoidance and path planning, and developed a stereo vision-based vehicle mapping system for 2D map navigation [20]. Peng proposed a three-dimensional map that uses stereo images to generate high-precision moon and Mars surfaces [21]. Bajracharya established a stereo vision near-field topographic mapping system. The mapping system combines stereo model-based outlier suppression and spatiotemporal filtering and achieves high robustness through a unique 2D/3D hybrid data structure [22]. Mou used unmanned surface vehicle equipped with binocular vision to detect the location of static obstacles and built the map of the obstacles. Finally, he completed mapping task of wide baseline stereo obstacles in marine environment [23]. Cavegn proposed an algorithm for automatic detection, classification, and mapping of road signs based on the depth information of stereo images [24]. Rankin's binary obstacle detection method is used in the terrain mapping algorithm for off-road navigation of unmanned ground vehicles [25]. Lin proposed a SLAM algorithm based on stereo images to solve the observability problem of moving entities and improve the accuracy of location, mapping, and tracking [26].

2.3. Summary of Research. By consulting the papers of machine vision in the field of crane and robotics, we can get the following conclusions:

- (1) Machine vision is rarely used in the field of bridge cranes, and most of it focuses on the movement detection of the hook in the antisway control. The operating environment mapping technology applied to mobile cranes cannot meet the working characteristics of the fixed scene of bridge cranes. For dimension standard obstacles such as steel coils and containers, the algorithm cannot be used for bridge cranes in more complex environments.
- (2) Machine vision has carried out a wealth of research in the field of robotics. Researchers use different types of sensors to build various algorithms that can meet different accuracy requirements. This paper proposed a solution for the compression problem of large maps.
- (3) Related research in the field of robotics has obtained relatively mature technology, which can provide valuable research ideas for three-dimensional mapping of bridge cranes. According to the characteristics of the operating environment of the bridge crane, a suitable algorithm is selected to solve the three-dimensional mapping problem of its operating environment.

3. 3D Mapping Method

3.1. Technical Framework. Figure 2 is a flowchart of the algorithm proposed in this paper, which is divided into the following three parts: binocular vision running at a high frequency of 10 Hz, camera location and mapping running at a low frequency of 1 Hz. Binocular vision is responsible for generating partial point clouds from the left and right images of the current single frame. Camera location uses two laser ranging sensors to measure the positions of trolley and cart, respectively, and then obtain the displacement of the camera installed on the trolley. The mapping is based on the camera displacement obtained by the camera positioning calculation. The partial point cloud obtained by binocular vision is registered, then the global point cloud is filtered, and finally, an octree map is generated.

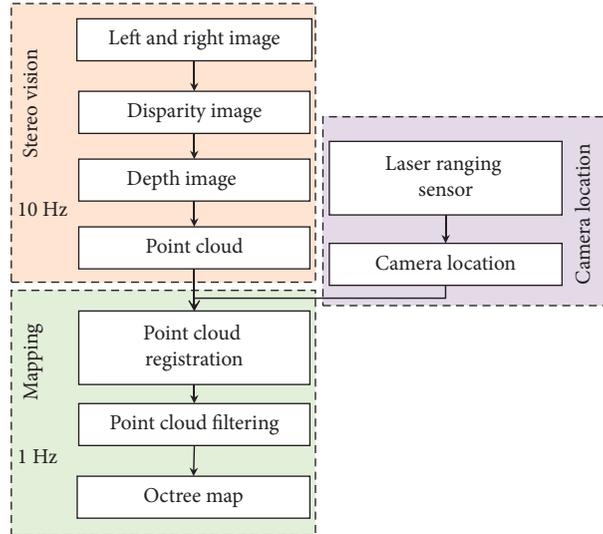


FIGURE 2: Algorithm flowchart.

For binocular vision, the binocular camera first captures the current field of view at the same time to obtain the left and right images. The block matching algorithm is used to convert the left and right images into a single disparity image. Each pixel in the disparity image represents the value of the difference of the corresponding spatial point of the pixel point in the left image and the right image. The reprojection matrix Q is used to calculate the three-dimensional depth for each two-dimensional point (x, y) in the disparity image and its corresponding given disparity, thereby converting the disparity image into a depth image. Each pixel value in the depth map is the depth value of the corresponding two-dimensional point (x, y) , so all two-dimensional points are reprojected to the three-dimensional space according to the depth image to generate the point cloud of the current frame. The generation process of the partial point cloud is shown in Figure 3.

Camera location uses two independently operating laser ranging sensors to obtain the location information of the camera. One of the laser ranging sensors is used to measure the position of the cart, and the other is used to measure the position of the trolley. The binocular camera is fixed on the trolley, and the coordinates of the binocular camera in the $x - y$ plane can be obtained according to the position information of the cart and the trolley returned by the two laser ranging sensors.

The mapping is based on the camera position information obtained by the camera location and the partial point cloud generated by a single frame of left and right images to generate a global point cloud by registering. Then, the global point cloud is filtered. Point cloud filtering includes sparse outlier point analysis and removal, and the use of the voxel grid method to achieve downsampling. Finally, the filtered point cloud image is transformed into an octree map.

3.2. Stereo Correspondence. Stereo correspondence is a process of matching a three-dimensional point in the left image and the right image and converting the left image and

the right image into a single disparity image. When the base distance is constant, the larger the difference between the left image and the right image of a 3D point in space is, the farther it is from the binocular camera; the smaller the difference is, the closer it is from the binocular camera. Therefore, the disparity image is obtained from the left image and the right image. Besides, combined with the camera internal parameters, the depth value of the object in the visible area can be calculated.

The commonly used stereo correspondence algorithms include block matching (BM) and semiglobal block matching. Block matching is a fast and effective algorithm. It uses a small “sum of absolute difference” (SAD) window to search for matching points between the left and right images. Block matching only searches for high-texture points between two images [27]. Therefore, in a highly textured scene, such as an outdoor forest, each pixel has a calculable depth. In a low-texture scene, such as an indoor corridor, only a few points of depth can be calculated. Unlike the block matching algorithm, the semiglobal block matching (SGBM) algorithm uses the Birchfield–Tomasi metric for matching at the subpixel level. SGBM attempts to enforce a global smoothness constraint on the calculated depth information. It considers many one-dimensional smoothness constraints of the region of interest. BM is better than SGBM in computing speed, but SGBM has better reliability and accuracy than BM. The reliability and accuracy of the calculation results of the BM meet the reliability and accuracy requirements of the operating environment of the bridge crane. Therefore, this paper adopts the fast block matching to obtain the disparity image; the specific block matching process is shown in Figure 4. Each feature in the left image is traversed through column loops and row loops. The matching calculation is performed by sliding the SAD window. When the calculation is completed for all sliding windows of the one feature, the best match is selected. The best match is the smallest value among multiple calculation

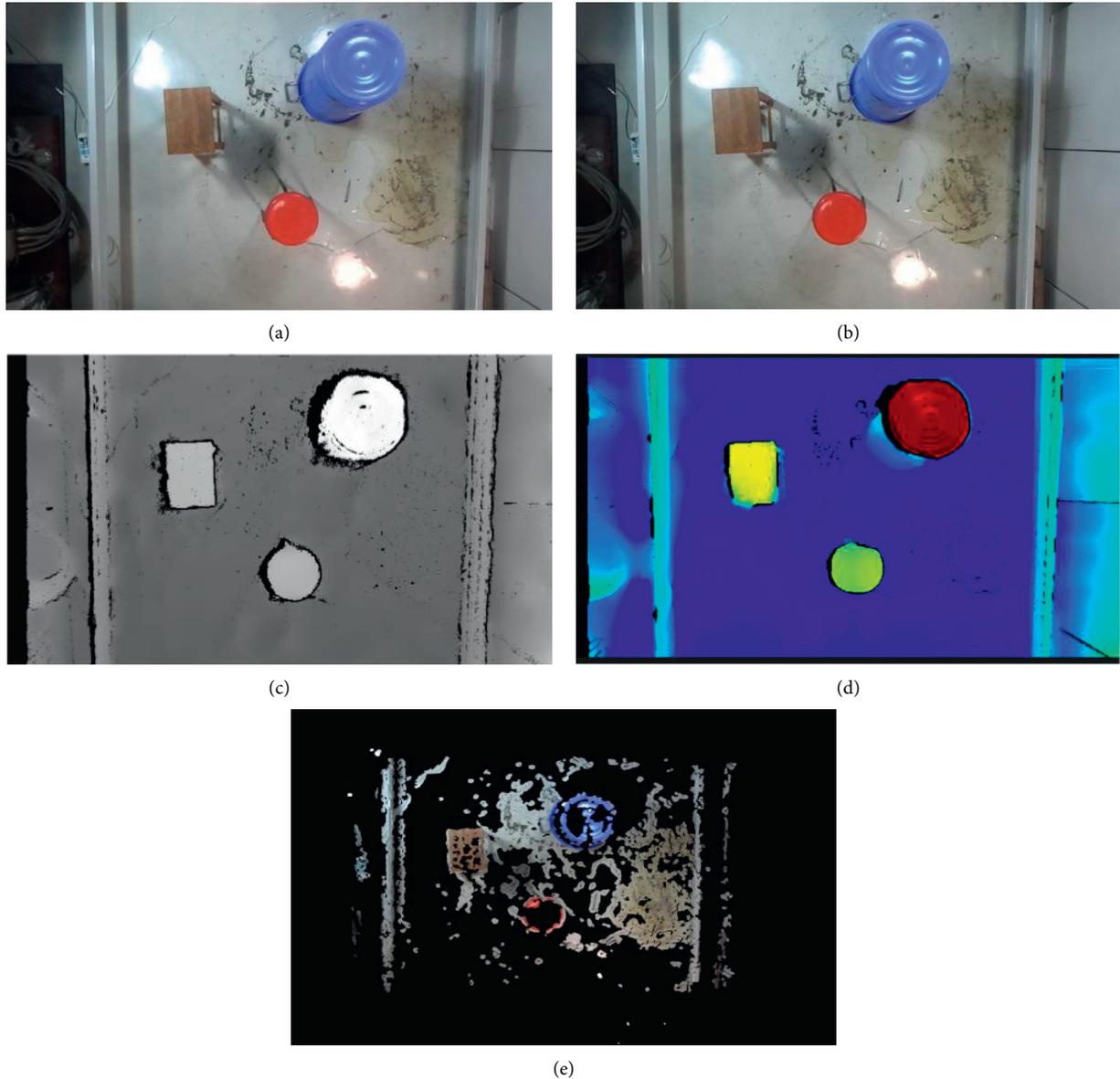


FIGURE 3: Generation process of the partial point cloud: (a) left image; (b) right image; (c) disparity image; (d) depth image; (e) point cloud.

results. After the matching calculation of all left images is completed, speckle filtering is used to obtain the final difference image.

The parameters that have a great impact on the disparity image results are block size, number of disparities, and speck window size. The parameter block size sets the size of the area around each pixel. The setting range is $[5, 255]$. Generally, it should be between 5×5 and 21×21 , and it must be an odd number. The larger the value is, the fewer false matches will be found. However, increasing the window will increase the computational cost of the algorithm. In addition, the larger the window, the smoother the disparity map, and some details may be lost. The parameter number of disparities is the maximum value of different disparities, which sets the range of search matching. The parameter speck window size sets the size of the speckle filter. The

speckle filter performs postfiltering on the matching results to obtain the output image.

Block matching algorithm has three steps:

- (1) Prefiltering to normalize the brightness of the image and enhance the texture
- (2) Using the SAD window to search for matching along the horizontal polar line
- (3) Postfiltering to eliminate bad corresponding matching

The specific matching process of Step (2) is as shown in Figure 5. For the pixel p_l in the left image, the windows with the size $w \times w$ around it are taken and marked as A ; n small blocks to the left along the epipolar line at the corresponding position in the right image are selected and marked as B_k ,

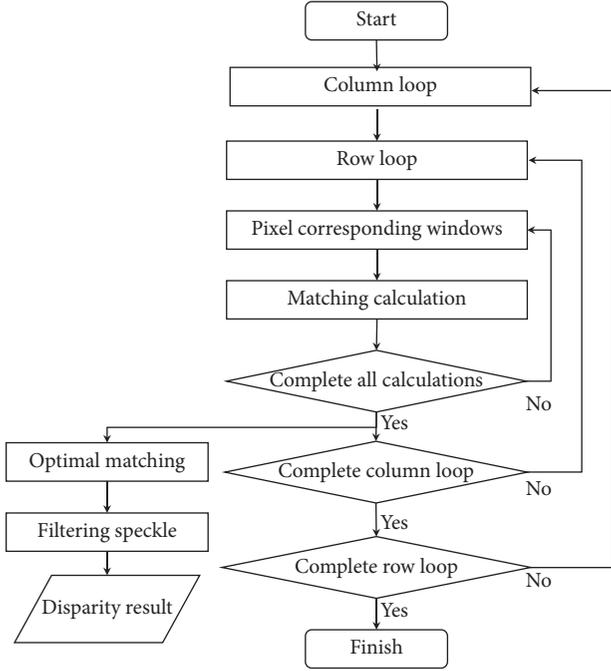


FIGURE 4: Disparity image process.

$k = 1, \dots, n$. The window A and block B are calculated as follows:

$$S(A, B)_{\text{SAD}} = \sum_{i,j} |A(i, j) - B(i, j)|. \quad (1)$$

The meaning of (1) is the sum of the absolute values of the difference between two blocks, so for n calculation results, the block B_{\min} with the smallest value among the calculation results is selected. Pixel p_r in the right image is the best matching result for pixel p_l , and p_l is the corresponding pixel of the block B_{\min} in the right image.

After finishing matching pixel of the left image with the right image, the disparity value can be calculated as follows:

$$d = u_1 - u_2. \quad (2)$$

After the above matching operation is carried out for all pixels in the left image, the result is a matrix with the same size as the left image. The matrix is the disparity image. The value of each element $d(u, v)$ is the disparity at the coordinate point (u, v) in the left image.

3.3. 3D Reconstruction. Three-dimensional reconstruction is the process of converting the disparity values in pixels into world coordinates in meters. As shown in Figure 6, there is a point P in space whose coordinate in the camera coordinate system is (x, y, z) . The pixel coordinates are relative to the upper-left of the left and right image. Pixels of point P in the left and right image are denoted by (u_1, v_1) and (u_2, v_2) , respectively. The centers of projection are at O_{C_l} and O_{C_r} . The distance between O_{C_l} and O_{C_r} is the base distance b of the camera [28].

After obtaining the pixel coordinates and the internal parameters of the binocular camera, the pixels in the two-

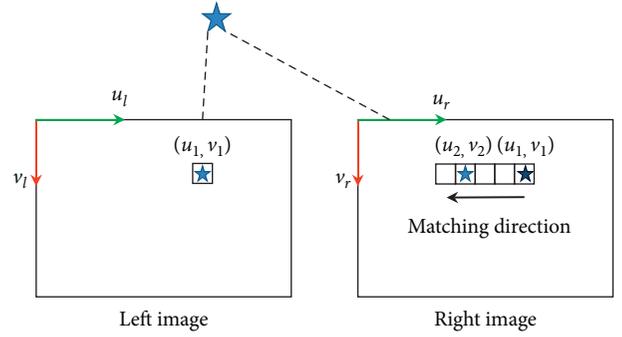


FIGURE 5: Matching process of the left image and right image.

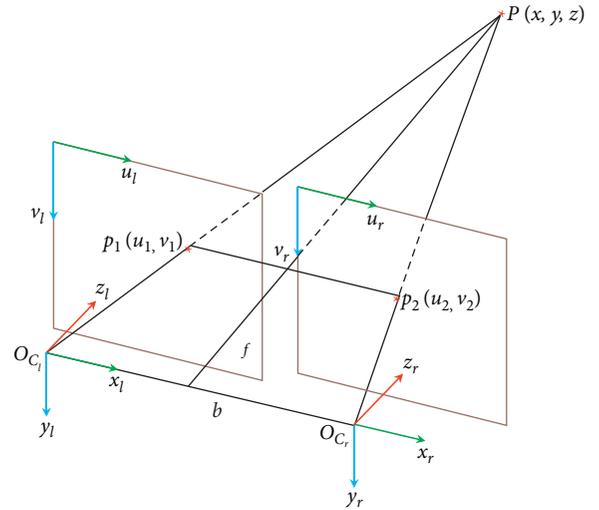


FIGURE 6: Principle of binocular triangulation.

dimensional image can be reprojected into the three-dimensional space. For the disparity d and corresponding two-dimensional points (u_1, v_1) , they can be obtained as follows according to the principle of similar triangles:

$$\frac{b}{b-d} = \frac{x}{x-u_1} = \frac{y}{y-v_1} = \frac{z}{z-f}. \quad (3)$$

The three-dimensional coordinates can be obtained by solving the calculation formula (3). The results are as follows:

$$\begin{cases} x = \frac{bu_1}{d}, \\ y = \frac{bv_1}{d}, \\ z = \frac{fb}{d}, \end{cases} \quad (4)$$

where f is the focal length of the left camera.

We save all generated three-dimensional space coordinate points into the point cloud, and we filter the point cloud. The point cloud filtering methods used in this article include StatisticalOutlierRemoval filter and VoxelGrid filter. Mismatches often cause errors in the estimation of the edge depth of the

object, so the StatisticalOutlierRemoval filter used in this article removes outliers caused by mismatches. The VoxelGrid filter uses voxelized grids to achieve downsampling, reducing the number of points. In the next position, the left and right eye images are taken again, and the above process is repeated to generate the point cloud of the new position. According to the position change data read by the laser ranging sensor, the original point cloud and the newly generated point cloud are registered. After the registration is completed, a three-dimensional point cloud of the working environment of the bridge crane, that is, the scene in the workshop, is generated. However, because bridge cranes tend to operate in a larger space, the saving space occupied by the generated point cloud files is often large, which is not suitable for file saving and transmission, and the point cloud files provide unnecessary details for the operation of bridge cranes, such as wrinkles on the surface of obstacles and shadows in the dark. Therefore, this paper introduces a flexible and highly compressed map form: octree map.

3.4. Camera Location and Point Cloud Registration.

Common robot location technologies include visual location technology, GPS location technology, and beacon location technology. Their advantages and disadvantages and application scenarios are shown in Table 1. Because the bridge crane works indoors, GPS location technology is not applicable. The visual location technology is to estimate the movement of the camera between continuous images to obtain the position data of the camera. It requires a lot of computing resources. Long-term use will produce a large cumulative error, and the accuracy is worse than that of the location technology using beacons.

Common methods for beacon location technology include magnetic nails, two-dimensional codes, barcode tapes, and laser ranging. Taking into account the reasons for the convenience of installation, this paper selects the laser ranging sensor to obtain the location data of the camera, as shown in Figure 7.

For a point P in space, it is P_w in the world coordinate, and it is P_c in the camera coordinate. Since the camera is moving, the movement of the camera can be considered as an Euclidean transformation, described by the transformation matrix $T \in SE(3)$, as shown in Figure 8. Generally, the left camera coordinate system is used as the camera coordinate system of the binocular camera, and the camera coordinate calculation is obtained as follows:

$$P_w = RP_c + t = TP_c, \quad (5)$$

where T can be represented as follows:

$$T = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (6)$$

The bridge crane only has three degrees of freedom: the movement of the cart, the trolley, and the hook. Therefore, the camera is fixed to the trolley and moves with the trolley,

so the camera only has linear movement with the cart and trolley movement. Thus, we can get that the rotation matrix R in formula (6) is the identity matrix, and the motion component in the z -axis direction is $t_3 = 0$. The motion transformation matrix of the binocular camera can be derived as follows:

$$T = \begin{bmatrix} I & t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_1 \\ 0 & 1 & 0 & t_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (7)$$

where t_1 and t_2 are the displacements of the cart and trolley, respectively, which are measured by the laser ranging sensors.

From equations (5) and (7), the point cloud generated by the images taken at different positions can be registered. The distance between the obstacle positions is related to the distance between the obstacle and the camera. When the distance between the obstacle and the camera increases, the distance between the shooting positions should also be appropriately increased. Too dense shooting positions will cause a great computational burden, and too sparse ones will reduce the robustness of the mapping results and even omit obstacles.

Because bridge cranes have high requirements for operational safety, this paper carries out multiple frames of the same object when building maps to enhance the robustness of the mapping results.

3.5. Octree Map. Several partial point clouds were obtained using the methods in Sections 3.2 and 3.3, and the partial point clouds were registered in Section 3.4 to obtain a large global point cloud. If the PCD file of the point cloud is used as the final saving format of the 3D mapping result, a large amount of saving space will be needed. A single frame of $1280 * 720$ will generate 921,600 spatial points, resulting in a large PCD file. However, the large PCD file provides a lot of unnecessary details. For example, the texture information of the obstacle surface is not useful for the three-dimensional perception of the crane operating environment. Therefore, it is necessary to use a method to compress the map while ensuring the quality of the map.

There are many kinds of representation of maps in robotics; the commonly used are scale maps, topological maps, and semantic maps. Scale maps have real physical scales, such as grid maps and point cloud maps and are often used in SLAM and path planning. The topological map only contains the connectivity and distance of different locations and does not have a real physical scale. Each location is represented by a point, and edges are used to connect adjacent points. For example, in the subway route map, it is described whether station A is connected to station B. Semantic map is a map form that uses a collection of tags to represent each location and road in the scene and is often used for human-computer interaction. To realize the three-dimensional perception of the operating environment of the bridge crane and prepare for the further intelligent obstacle

TABLE 1: Comparison of location techniques.

	Visual location	GPS location	Beacon location
Accuracy	Low	Medium	High
Scenario	Outdoors/indoors	Outdoors	Fixed place
Computing resources required	High	Low	Low



FIGURE 7: Laser ranging sensor.

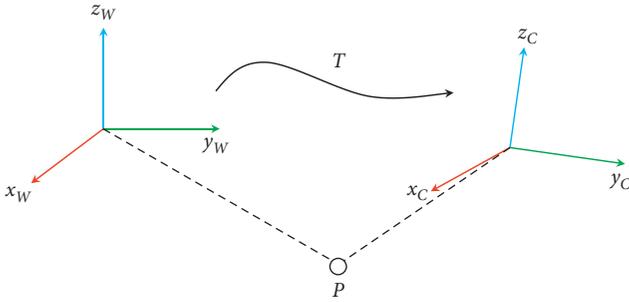


FIGURE 8: Coordinate transformation.

avoidance operation, this paper selects the octree map which can be used in the scale map for path planning of the problem of map selection.

The octree map is a commonly used and easy-to-compress map form in navigation, and it is a tree structure that describes a three-dimensional space. The first node of the octree is the root node. For an octree containing a 3D point cloud model, the bounding box of the 3D point cloud model is its root node. Each node in the octree corresponds to a spatial cube, and each internal node (nonleaf node) has 8 child nodes. These 8 child nodes use the center of the parent node as the bifurcation center to divide the space where the parent node is located into 8 small cubes. When building an octree on the 3D point cloud model, those nodes on the surface of the model will be divided into details until reaching the specified number of layers. If a spatial region contains only one data point or the region reaches the maximum depth specified by the octree, it is represented by a leaf node of the octree. If a spatial region contains more than one data point, the region is divided into eight equal parts. In this way, the recursive segmentation of the entire data cube is completed cyclically. Figure 9 shows the process of recursively splitting a data cube and the corresponding octree structure. The definition of the octree is as follows:

$$\text{octree} = \begin{cases} v_i = \frac{1}{8}v_{i-1}, d_i = \frac{1}{2}d_{i-1}, h_i = h_{i-1} + 1, & n > 2, \\ \text{end of separation,} & n \leq 2 \text{ or } d_i = d_{\min}, \end{cases} \quad (8)$$

where v_i , d_i , and h_i are the volume, side length, and depth of the small cube in the i th division; n is the number of objects contained in the small cube; and d_{\min} is the minimum side length that the cube can be divided into.

The value of the side length of the voxel should be selected according to the requirements of the mapping accuracy. If the side length is too large, most of the space in the voxel is empty, and the octree map cannot describe the shape of the object well. If the side length is too small, it will cause redundancy in the octree map. Too many depth levels will also make the octree map file larger. According to Srinivasan Ramanagopal's research [29], an appropriate value can be obtained using the following formula:

$$d_{\min} = 0.1 \times D, \quad (9)$$

where D is the distance from the camera to the obstacle.

In the three-dimensional environment of actual production and life, objects are often in contact with each other, and blank areas are often connected together. In the octree model, the information about whether it is occupied is stored in the node. When all the child nodes of a block are occupied or not occupied, there is no need to expand the node. For example, when the map is blank at the beginning, only one root node is required, not a complete tree. Therefore, most octree nodes do not need to expand to the leaf level. Consequently, the octree saves a lot of space compared to the point cloud.

4. Experiment and Discussion

The three-dimensional mapping problem of the operating environment of a bridge crane is that a camera moves in a horizontal plane to construct a three-dimensional scene from a top view. It is difficult to find a public data set suitable for the 3D mapping scene of the bridge crane. Therefore, this paper builds an experimental platform according to the operating characteristics of the bridge crane and performs the experimental verification according to the method proposed in this paper. The error was calculated by the experimental results, and the factors affecting the experimental results were analyzed.

4.1. Experimental Design. Based on the algorithm proposed in this paper and the operating characteristics of the bridge crane, an experimental platform is built, as shown in

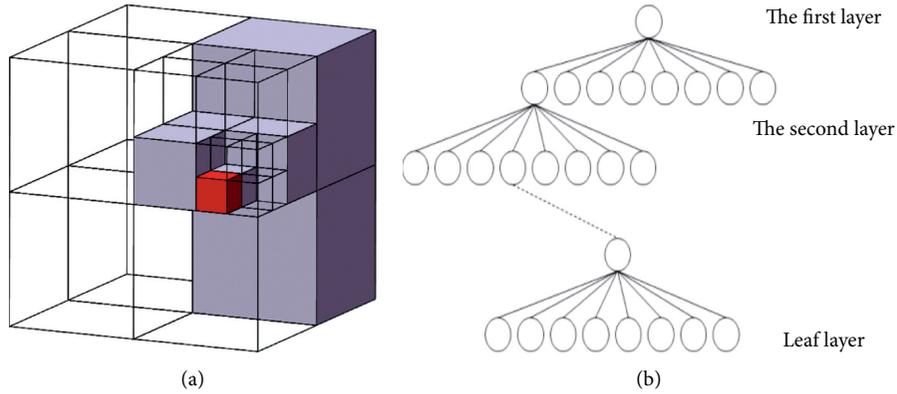


FIGURE 9: Diagram of the octree.

Figure 10. The traveling distance of the cart is 3 m, and the traveling distance of the trolley is 2 m. The binocular camera is fixed on the trolley. Two laser ranging sensors measure the displacement of the cart and the trolley, respectively. The experimental process is shown in Figure 11.

To make the mapping cover the working area of the experimental platform, six taken positions are selected, as shown in Figure 12, so that the mapping result of the same obstacle can be made up of at least 4 frames of point clouds at different positions. The base distance of the binocular camera used is 120 mm, and the image resolution is 1280×720 .

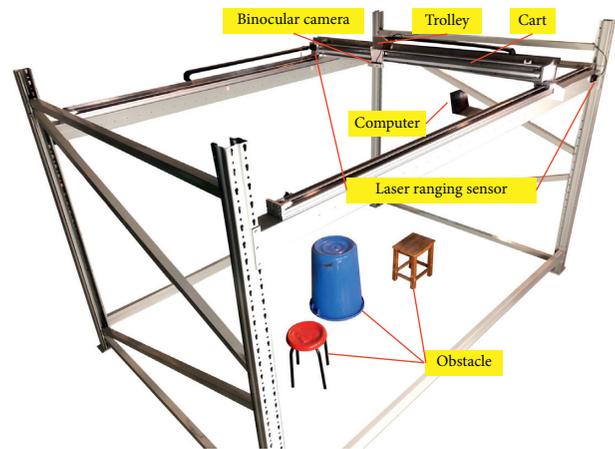


FIGURE 10: Experimental platform.

4.2. Experimental Equipment. The performance information of the equipment used in the experiment is shown in Table 2. The sliding table and bracket are both customized and produced by the manufacturer after requesting.

4.3. Experimental Process. (1) Experimental preparation:

Connect the binocular camera and the two laser ranging sensors to the computer through a USB extension cable, respectively. Turn on the power, turn on the computer, and enter the Ubuntu system.

(2) Experimental operation:

In the stereo matching step, the block size is set to 9×9 , and the disparity number calculation function is as follows:

$$\text{disparity number} = \left(\left(\frac{\text{imgSize} \cdot \text{width}}{8} \right) + 15 \right) \& - 16. \quad (10)$$

Here, $\text{imgSize} \cdot \text{width}$ is the width of the processed image. The image resolution in the experiment is 1280×720 , so $\text{imgSize} \cdot \text{width}$ is 1280. Use formula (10) to calculate the disparities number to be 160. The speck window size of the speckle filter is 100.

Open the serial port assistant and start to collect the position data of cart and trolley through the laser ranging sensor.

Open Terminal and run the environment mapping program written. The motor starts to drag the camera to take pictures at positions 1 to 6. After the shooting is completed, a global point cloud file and an octree file are generated.

(3) Result analysis:

Use `pcl_viewer` and `octovis` to open the point cloud file and the octree file, respectively. Besides, error analysis is also carried out.

4.4. Experimental Result. According to the proposed experimental steps, the experimental platform is used to carry out the experiment. Choose cylinders, boxes, and a chair as obstacles in the crane's working environment.

The global 3D point cloud map of the experimental results is shown in Figure 13. The color octree map is shown in Figure 14. The distance between the camera and the obstacle is about 1 m. According to the calculation method of equation (9), the appropriate voxel side length d_{\min} should be 0.1 m. The results show that obstacles and obstacle-free areas in the operating space are effectively constructed in both the point cloud map and the octree map. The color information is provided in the color octree map, and it is

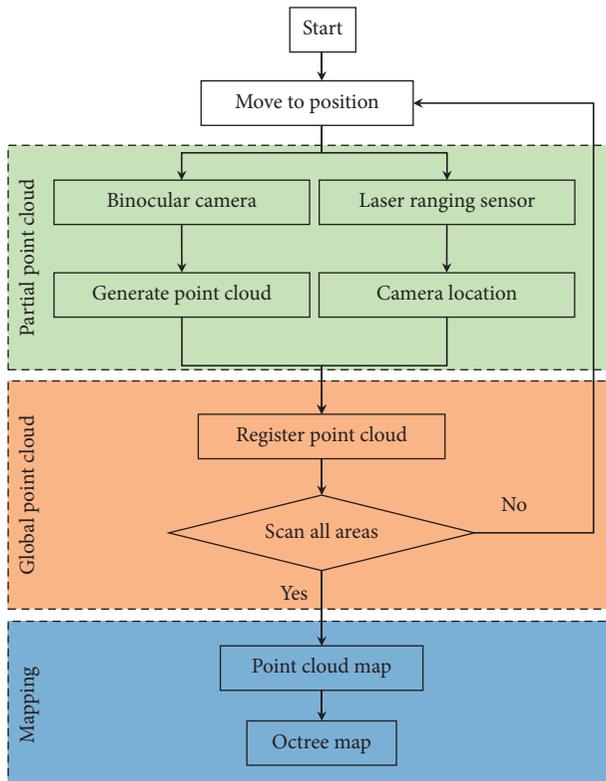


FIGURE 11: Experimental process.

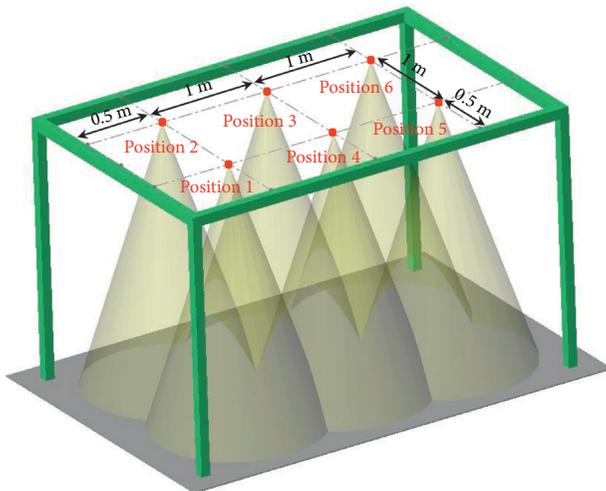


FIGURE 12: Photo location.

saved in a common file (.ot format). The colorless octree map has no color information and is saved as a binary file (.bt format). Therefore, compared with the colorless octree map, the color octree map is more intuitive and is mostly used for human-computer interaction and other purposes. The colorless octree map has less saving space than the color octree map. In the experimental results, the point cloud map size is 1.3 M, the color octree map is 1.1 M, and the colorless octree map is 665 kB. The octree map greatly compresses the saving space of the point cloud map, which is convenient for data transmission and saving. The user of

the octree map can choose whether to have color information according to the mapping requirements and file size requirements.

In this paper, 10 sets of experiments are carried out with the proposed method on the same obstacle in the same environment. The average time for partial point cloud processing is 0.0968 s, and the average time for point cloud filtering and octree map generation is 0.8773 s. That is, the method used in this paper can achieve a partial point cloud generation frequency of 10 Hz and an octree map update frequency of 1 Hz. In order to evaluate the influence of the obstacle surface roughness and the different parameter values used in the algorithm of this paper on the accuracy of the mapping results, this paper uses the normalized error as the evaluation index. The experimental results are analyzed. And the key data are calculated by the method proposed in this article: the calculation result of the obstacle height in the z -axis direction. Besides, the normalized error of the three obstacle mapping results is obtained, as shown in Figure 15, where obstacle 1 is a blue barreled obstacle, obstacle 2 is a red round stool, and obstacle 3 is a brown square stool.

The error analysis of the three obstacles is shown in Table 3. It can be seen from the table that the maximum error, the minimum error, and the mean error of obstacle 1 and obstacle 2 are approximately equal. The errors of obstacle 3 are the smallest among the three obstacles, and the mapping result is the best.

4.5. Experimental Analysis. The analysis of the experimental results shows that the main reasons that affect the mapping results are (1) the illumination conditions and the surface roughness of the obstacles; (2) the point cloud filter parameters; and (3) the accuracy of the laser ranging sensors.

Obstacles 1 and 2 are made of plastic, and obstacle 3 is made of wood. The surface roughness of obstacles 1 and 2 is almost the same. The surface of obstacle 3 is rougher than obstacles 1 and 2. Under the same illumination conditions, obstacles 1 and 2 reflect more light than obstacle 3, which forms some bright spots on the photo. In the block matching process, the bright spots cause more mismatches and affect the final mapping accuracy. As shown in Figure 16, there are several white bright spots on the highest part of the point cloud image of the red stool, but there is no white area on the upper surface of the stool entity, so the white bright spots in the point cloud are mismatches caused by reflected light. Since the position of the bright spot relative to the stool entity in the photo will change with the change of the camera position, the depth estimation of the bright spot is often wrong, and the reconstruction result of the bright spot often appears above the ideal reconstruction result of the obstacle, resulting in relatively large errors. Wooden obstacles with rougher surfaces reflect less ambient light, so the reconstruction results are more accurate.

For the same situation, polarizers can be used to reduce the mapping error caused by ambient light, or a surface light source can be used instead of a point light source to improve the lighting conditions, to obtain better mapping results.

This paper uses the `pcl::StatisticalOutlierRemoval` filter in the point cloud filtering, the number of adjacent points analyzed

TABLE 2: Experimental equipment.

Equipment	Performance parameter	Remarks
Binocular camera	Depth range: 0.5–20 m, baseline: 120 mm	
Laser ranging sensor	40 m range, accuracy: ± 3 mm	
Sliding table module	Cart traveling distance: 3 m, trolley traveling distance: 2 m, speed of both: 0.1 m/s	Used to carry the camera to do translational movement in the x -axis and y -axis directions
Computer	CPU: i5-7300 HQ, RAM: 8 GB	
USB extension cord	USB: 3.0, length: 10 m	Used to connect sensors and laptops
Bracket	3.4 m long, 2.5 m wide, 2.2 m high	

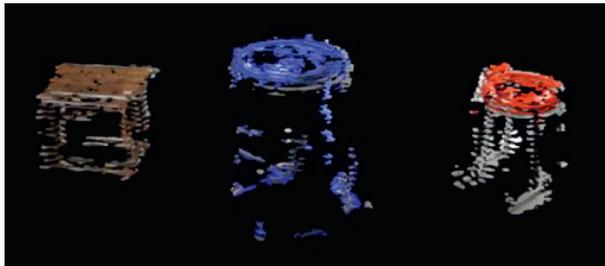


FIGURE 13: Point cloud mapping results.

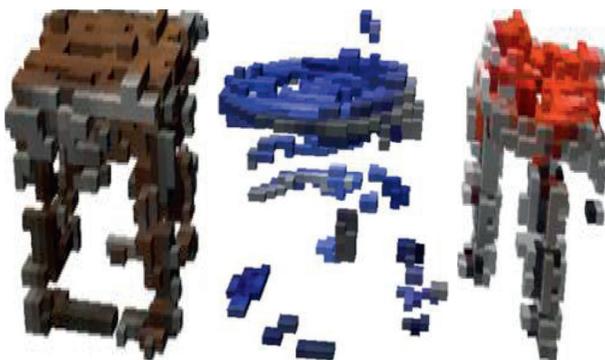


FIGURE 14: Results of octree mapping.

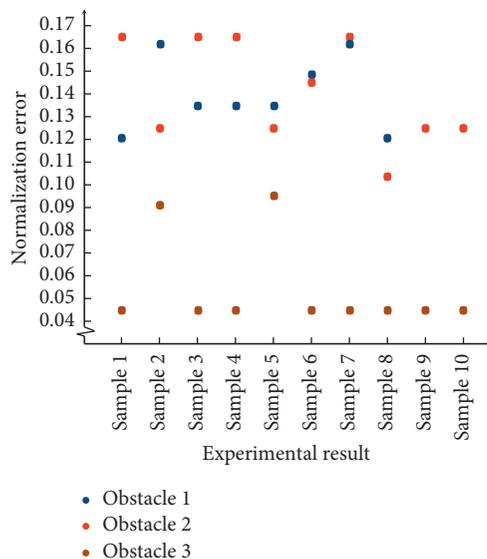


FIGURE 15: Normalization error.

TABLE 3: Error analysis.

	Maximum error (%)	Minimum error (%)	Mean error (%)
Obstacle 1	16.21	12.16	14.168
Obstacle 2	16.66	10.41	14.162
Obstacle 3	9.52	4.54	5.461

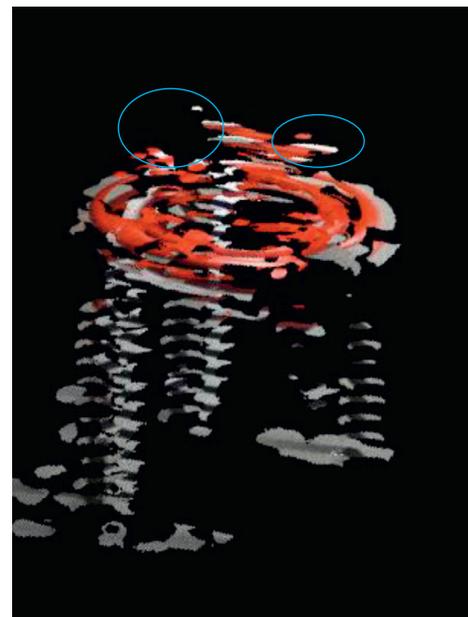


FIGURE 16: Mismatch caused by light spot.

for each point is set to 50, and the standard deviation multiple is set to 1. These parameter values cause if the distance of a point exceeds the average distance by more than one standard deviation, the point will be marked as outlier and will be removed. The filtering effect of StatisticalOutlierRemoval filter on outliers is shown in Figure 17. The left picture is the point cloud before filtering, and the right one is the point cloud after filtering. We can see that the point set caused by the mismatch becomes smaller or even disappears. The parameters of this filter are adjusted appropriately, the number of adjacent points is set to 60, and the standard deviation is set to 0.8. The error statistics of the mapping results are shown in Table 4. Comparing the mapping results before and after the parameter modification, it can be found that the reduction of the normalization error is at

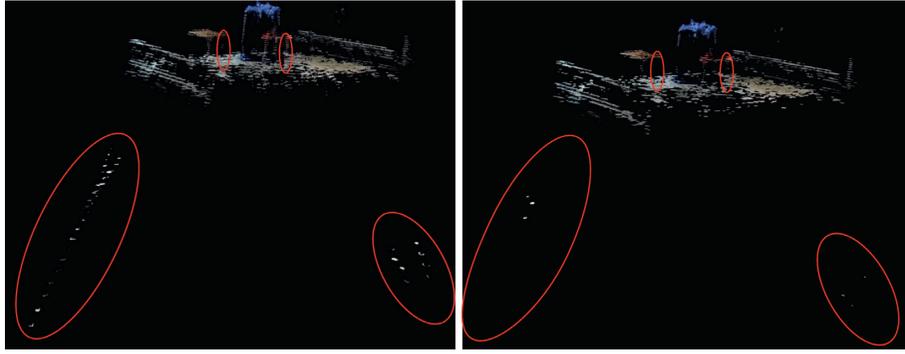


FIGURE 17: Outlier filtering effect.

TABLE 4: Error analysis.

	Before parameter modification			After parameter modification		
	Maximum error (%)	Minimum error (%)	Mean error (%)	Maximum error (%)	Minimum error (%)	Mean error (%)
Obstacle 1	16.21	12.16	14.168	13.17	10.54	12.522
Obstacle 2	16.66	10.41	14.162	12.76	9.33	10.981
Obstacle 3	9.52	4.54	5.461	8.468	4.04	4.579

the cost of reducing the resolution. It can be seen that each error has been reduced to varying degrees, but the generated mapping results have lost some details compared to those before modifying the parameters mapping results. Some texture details in the point cloud before the parameter adjustment become inconspicuous or even disappear after the parameter adjustment. Therefore, when setting the filter parameters, they should be adjusted according to the requirements of the mapping accuracy and the lighting conditions. For the crane operating environment, when the obstacles are mostly regular objects, the number of adjacent points can be increased appropriately and the standard deviation can be decreased. In this case, reducing the texture details of the obstacle will not affect the mapping effect. For scenes with higher requirements on the surface texture of obstacles, the number of adjacent points should be reduced and the standard deviation should be increased. In this way, more detailed mapping results can be obtained. For bridge cranes, which require high safety equipment, the robustness of the mapping results can be improved.

The measured accuracy of the laser ranging sensor selected in this paper is ± 3 mm within the range of 0–3 m. This error will cause the same object to be misaligned in the x -axis direction and y -axis direction during point cloud registration, which will affect the final point cloud mapping accuracy.

5. Conclusions

In this paper, aiming at the problem of three-dimensional mapping of the operating environment of bridge cranes, a mapping method in a fixed scene is established by using a binocular camera and laser ranging sensor. To realize this three-dimensional mapping method, the following work has been done:

- (1) The mathematical model of binocular mapping is established, which is suitable for the three-

dimensional perception of the operating environment of the bridge crane. The camera moves in the x -axis and y -axis directions and measures the depth in the vertical z -axis direction.

- (2) The binocular camera is used to build the 3D map of the obstacles within the field of vision to form a partial point cloud, the laser ranging sensor is used to locate the position of the binocular camera, the partial point cloud is registered according to the position information to a global point cloud, and finally, the octree map is used to represent the global point cloud.

In this paper, an experimental platform is built to verify the method by mapping several different sizes and shapes of obstacles. The results show that a normalized accuracy of 4.54%–16.66% is achieved. In addition, the analysis of the calculation time shows that the framework can run in real time when the partial point cloud and workspace update loop are running at 10 Hz and 1 Hz, respectively.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the China National Key R&D Program during 13th Five-Year Plan Period (Grant no. 2017YFC0704000).

References

- [1] H. Osumi, M. Kubo, S. Yano, and K. Saito, "Development of tele-operation system for a crane without overshoot in positioning," in *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5799–5805, Taipei, Taiwan, 2010.
- [2] Z. Chen, Y. Zhou, F. Wang, and Y. Zhou, "Structural design and experimental study of the crane avoidance warning visual system," in *Proceedings of the 2013 3rd International Conference on Consumer Electronics, Communications and Networks*, pp. 605–608, Beijing, China, November 2013.
- [3] Z. K. Chen, H. Y. Wang, W. H. Wang, and F. B. Wang, "System design of crane robot based on binocular stereo vision," *Applied Mechanics and Materials*, vol. 303, pp. 1691–1694, 2013.
- [4] W. Fubin, "Design of crane robot," *Machine Tool & Hydraulics*, vol. 43, no. 21, pp. 50–57, 2015.
- [5] Y. Yoshida and H. Tabata, "Visual feedback control of an overhead crane and its combination with time-optimal control," in *Proceedings of the IEEE/ASME Int. Conf. Adv. Intell. Mechatronics, AIM*, pp. 1114–1119, Delft, The Netherlands, July 2008.
- [6] J. Smoczek, J. Szpytko, and P. Hyla, "The application of an intelligent crane control system," *IFAC Proceedings Volumes*, vol. 45, no. 24, pp. 280–285, 2012.
- [7] J. Szpytko and P. Hyla, "Disparity compute methods IN three-dimensional scene reconstruction for overhead travelling crane work space visualization," *Journal of KONES Powertrain and Transport*, vol. 19, no. 3, pp. 421–428, 2015.
- [8] L. Xiaobo, C. Wenming, and G. Peng, "Research and design of automatic handling system for container cranes," *Journal of Computational and Applied Mathematics*, vol. 35, 2015.
- [9] S. Kim, S. Choi, S. Won, and H. Jeong, "The coil recognition system for an unmanned crane using stereo vision," in *Proceedings of the 2004 IEEE Industrial Electronics Conference*, vol. 2, pp. 1235–1239, Busan, South Korea, 2004.
- [10] J. Chen, Y. Fang, and Y. K. Cho, "Real-time 3D crane workspace update using a hybrid visualization approach," *Journal of Computing in Civil Engineering*, vol. 31, no. 5, Article ID 04017049, 2017.
- [11] Y. Cho, C. Wang, M. Gai, and J. W. Park, "Rapid dynamic target surface modeling for crane operation using hybrid LADAR system," in *Proceedings of the 2014 Construction Research Congress*, pp. 1053–1062, Atlanta, GA, USA, 2014.
- [12] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard, "3-D Mapping with an RGB-D camera," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 177–187, 2014.
- [13] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, "RGB-D mapping: using Kinect-style depth cameras for dense 3D modeling of indoor environments," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 647–663, 2012.
- [14] J. Ryde and H. Hu, "3D mapping with multi-resolution occupied voxel lists," *Autonomous Robots*, vol. 28, no. 2, pp. 169–185, 2010.
- [15] M. Smith, I. Posner, and P. Newman, "Adaptive compression for 3D laser data," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 914–935, 2011.
- [16] J. P. Saarinen, H. Andreasson, T. Stoyanov, and A. J. Lilienthal, "3D normal distributions transform occupancy maps: an efficient representation for mapping in dynamic environments," *The International Journal of Robotics Research*, vol. 32, no. 14, pp. 1627–1644, 2013.
- [17] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: an efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, 2013.
- [18] F. Schule, M. S. Kramer, R. Schweiger, K.-D. Kuhnert, and K. Dietmayer, "Night-vision stereo grid mapping for digital map localization," in *Proceedings of the 2014 IEEE Intelligent Vehicles Symposium Proceedings*, Dearborn, MI, USA, June 2014.
- [19] C. Yu, V. Cherfaoui, and P. Bonnifait, "Evidential occupancy grid mapping with stereo-vision," in *Proceedings of the 2015 IEEE Intelligent Vehicles Symposium (IV)*, Seoul, South Korea, August 2015.
- [20] C. Brand, M. J. Schuster, H. Hirschmüller, and M. Suppa, "Stereo-vision based obstacle mapping for indoor/outdoor SLAM," in *Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Chicago, IL, USA, September 2014.
- [21] M. Peng, Z. Yue, Y. Liu, and K. Di, "Lunar and Mars orbital stereo image mapping," in *Proceedings of the Remote Sensing of the Environment: The 17th China Conference on Remote Sensing*, Hangzhou, China, August 2010.
- [22] M. Bajracharya, J. Ma, M. Malchano, A. Perkins, A. A. Rizzi, and L. Matthies, "High fidelity day/night stereo mapping with vegetation and negative obstacle detection for vision-in-the-loop walking," in *Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3663–3670, San Diego, CA, USA, November 2013.
- [23] X. Mou and H. Wang, "Wide-baseline stereo-based obstacle mapping for unmanned surface vehicles," *Sensors*, vol. 18, no. 4, pp. 1085–1114, 2018.
- [24] S. Cavegn and S. Nebiker, "Automated 3D road sign mapping with stereovision-based mobile mapping exploiting disparity information from dense stereo matching," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XXXIX-B4, pp. 61–66, 2012.
- [25] A. L. Rankin, A. Huertas, and L. H. Matthies, "Stereo vision based terrain mapping for off-road autonomous navigation," in *Proceedings of the SPIE 7332, Unmanned Systems Technology XI*, Zhangjiajie, China, April 2009.
- [26] K. H. Lin and C. C. Wang, "Stereo-based simultaneous localization, mapping and moving object tracking," in *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010)*, pp. 3975–3980, Taipei, Taiwan, October 2010.
- [27] A. Kaehler and G. Bradski, *Learning OpenCV*, O'Reilly Media, Inc., Newton, MA, USA, 2012.
- [28] P. Corke, *Robotics, Vision and Control*, Springer International Publishing, Cham, Switzerland, 2017.
- [29] M. Srinivasan Ramanagopal, A. P.-V. Nguyen, and J. Le Ny, "A motion planning strategy for the active vision-based mapping of ground-level structures," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 1, pp. 356–368, 2018.