

Vision detection and path planning of mobile robots for rebar binding

Bin Cheng | Lei Deng

School of Mechanical and Electrical Engineering, Xi'an University of Architecture and Technology, Xi'an, China

Correspondence

Lei Deng, School of Mechanical and Electrical Engineering, Xi'an University of Architecture and Technology, 710055 Xi'an, China.
Email: denglei1412@163.com

Funding information

Natural Science Basic Research Project of Shaanxi Province

Abstract

Focused on the problems of cumbersome operation, low efficiency, and high cost in the traditional manual rebar binding process, we propose a mobile robot vision detection and path-planning method for rebar binding to realize automated rebar binding by combining deep learning and path-planning technology. A MobileNetV3-SSD rebar binding crosspoints recognition model is built based on TensorFlow deep learning framework, and a crosspoints localization method combining control factor α and feature projection curve is introduced to achieve the localization of unbound crosspoints. In addition, A back-and-forth path-planning algorithm with priority constraints combined with dead zone escape algorithm based on improved A* is proposed to achieve complete coverage path planning of the working area and path transfer of the dead zone. In the field test of the robot prototype, the classification accuracy and localization accuracy reached 94.40% and 90.49%, and the robot was able to reach complete coverage path planning successfully. The experimental results show that the visual detection method can achieve fast, noncontact and intelligent recognition of rebar binding crosspoints, which has good robustness and application value. At the same time, the proposed path-planning method has higher efficiency in the execution of robot complete coverage path planning, and meets the basic requirements of path planning for rebar binding process.

KEY WORDS

A* algorithm, deep learning, image processing, MobileNetV3-SSD, mobile robots, path planning, rebar binding

1 | INTRODUCTION

The new round of technological revolution and industrial change with artificial intelligence, industry 4.0, and next-generation internet technology as the core is profoundly affecting traditional industries, and intelligent robotics brings historical opportunities for the development of technological innovation and management mode change in construction engineering.

In engineering construction, the reinforced concrete is an important form of structure, which is the main structure of the building and plays an important role in supporting the building. Before the reinforced concrete is molded, it is necessary to bind

several rebar crosspoints to form a rebar framework, thus constituting a larger rebar mesh (Spencer et al., 2019). Among them, whether the rebar crosspoints in concrete are firmly bound is directly related to the quality of concrete structure construction. Currently, most of the rebar binding methods are manual binding, and there are also small fully automatic rebar binding devices. Generally speaking, the total area of the rebar mesh is large, and the number of rebar crosspoints to be bound is large, which is limited by the physical strength of the workers, making manual binding laborious and inefficient. Small handheld rebar binding machine is mainly composed of four parts: body, special bobbin, battery box, and charger. It can not only solve the problem of efficiency, but also simple structure,

easy to carry, and can greatly reduce the labor intensity of workers. However, the rebar binding process still requires manual control, and the process can only be carried out by one person holding a binding machine for a single crosspoint. In addition, it also requires the staff to repeat the bending and squatting operation, which will cause damage to the staff's body, thus reducing the efficiency of the staff's tying. To improve economic efficiency and reduce the physical injury to workers at the rebar engineering site, and improve the degree of construction industrialization and intelligence, an adaptable, intelligent, and automated way of binding rebars is needed. Therefore, exploring a robot with an autonomous movement function is both a practical engineering need and the current trend of construction industrialization and intelligence.

In recent years, the rapid development of robotics and the deep integration of subject knowledge with the construction field have brought light to solve this problem (Gharbia et al., 2020; Melenbrink et al., 2020). A refractory spraying robot called SSR-1 (Ueno et al., 1986) was first developed by Shimizu Corporation in Japan in 1982 and is considered to be the world's first construction robot for building construction. Within the following decades, construction robots were highly valued and rapidly developed around the world. Depending on the functions used by construction robots, include bricklaying robots (Shi et al., 2023), painting robots (Wang et al., 2022), floor-tiling robots (Prabakaran et al., 2023), and construction measuring robots (Chen et al., 2019). Overall, construction robots can be mainly divided into robots modified in existing machine buildings and robots for specific construction operations, and thus unmanned construction operations (Pan et al., 2020).

The complex site environment of rebar works has high demands on the structure (Yang et al., 2020), perception (Galceran et al., 2013; Li et al., 2021), and control (Spong et al., 2006) of the robot. The structural design of the robot is particularly critical to adapt to the rebar mesh conditions and to meet the functions of rebar binding. Advanced Construction Robotics has developed a gantry rebar binding robot called Tybot. The robot is driven by a motorized frame, locates the rebar crosspoints by marking tags, and the binding is done by a gantry suspended robot arm. However, the Tybot is relatively clumsy and must be deployed at the construction site in advance. It is only suitable for large construction scenarios and is not conducive to scenarios that require the flexible deployment of rebar, such as indoor construction and prefabricated panel manufacturing scenarios. In 2017, Chiba Institute of Technology developed and designed T-iROBO rebar binding robot. The robot uses a two-way walking mechanism to move in the rebar mesh, and obtains real-time information on the crosspoints of rebar surfaces and the distance of obstacles through infrared detection methods to automate the rebar binding operation. Sweden's Skanska invented an ABB robot for the production of rebar steel cages. The robot uses multiple manipulators to work in tandem with ABB's offline programming simulation software, and the results show that the production time for the binding of rebar structures was reduced from 16 to 1 h per ton, greatly increasing the speed of manufacturing rebar structures.

Recognition and localization of rebar crosspoints is the core of the automatic binding process (Zhao et al., 2016). For the rebar image, there are two main image features: one is the overall linear feature and the other is the edge shape feature in detail. For the localization of rebar crosspoints, the analysis methods include line segment detection (Ying et al., 2010), corner point detection (Wu et al., 2015), and so forth. The most used line segment detection algorithms are Hough transform (Ying et al., 2010), RANSAC algorithm (Kim et al., 2021), and so on. In addition, corner point detection algorithms such as Harris corner point detection (An et al., 2022) are used. However, due to the diverse backgrounds of stored rebar, traditional image processing methods have limited generalization capability, which reduces the usefulness of detection methods.

With the in-depth study of deep learning theory, many scholars have carried out research on deep learning-based algorithms for target recognition, classification, and detection. Convolutional Neural Networks (CNNs) are the most widely used deep learning network in the field of image processing, and due to their superior performance, a series of models such as Region-based CNN (R-CNN; Girshick et al., 2014) and you only look once (YOLO; Redmon et al., 2016) have been gradually formed, which are used in the fields of target recognition, crack detection, and rebar counting (Joshi et al., 2022; Lee et al., 2019; Wang et al., 2019). Fang et al. (2018) used a data set containing 1500 images and trained an IFaster R-CNN model to achieve the recognition of workers and equipment at construction sites with an accuracy of 95%. Li et al. (2018) used the YOLO model for image classification to predict the area and size information of steel surface defects with an average precision and recall of 97.55% and 95.86%. Watfa et al. (2022) used the OpenCV image processing module to preprocess the images of crosspoints and introduced the Tiny-YOLO model to train the images to recognize the crosspoints in different backgrounds with an accuracy of 92.3%. Despite the current rapid development in the study of CNNs in terms of architecture and parameters, the network models are generally based on laboratory high-performance hardware support, which requires high costs in practical engineering applications and cannot provide recognition conditions for mobile devices. In 2016, Liu et al. (2016) combined the logistic regression idea of YOLO model and the anchor box mechanism of Faster R-CNN model to propose a Single Shot Multibox Detector (SSD) model, which uses VGG16 (Simonyan et al., 2014) as the base network, uses six scale feature layers for feature extraction of objects of different sizes, and sets multiple anchor boxes with various aspect ratios for edge regression, which improves the prediction accuracy of the model and reduces the complexity of the model to some extent (Zhang et al., 2022). Although the SSD model has good detection performance, it still suffers from delay due to a large number of VGG16 structural parameters. In recent years, Google introduced MobileNet, a light-weight network for mobile devices, and further developed MobileNetV3 network. Compared with VGG16, MobileNetV3 network can greatly reduce the complexity of the model and the number of parameters with similar accuracy (Howard et al., 2019).

Path planning is one of the key technologies to achieve robot intelligence, and is divided into point-to-point and complete coverage path planning. Point-to-point path planning requires the robot to plan a path with the shortest possible path and the least number of turns between the starting point and the ending point. However, for some specific works, such as window cleaning robots (Bisht et al., 2022), lawn robots (Wei et al., 2022), and ship repair robots (Muthugala et al., 2022), it is not only necessary to satisfy the above conditions but also to traverse the whole unobstructed work area and finally form a complete path from the starting point to the ending point in the work area (Bosse et al., 2007). Choset (2000) developed the boustrophedon cellular decomposition. The algorithm divides the map into several regions to be covered based on the boundaries of obstacles and uses Depth-First Search on the workspace for complete region coverage. An et al. (2020) proposed a complete coverage path-planning solution for circular mobile robots, which first divides the map into several square triangles and second determines the complete coverage path based on the size and extent of several square triangles.

Hassan et al. (2020) proposed a novel decentralized multirobot coverage path-planning approach that is adaptive to unexpected stationary and moving obstacles while aiming to achieve complete coverage with minimal cost. Li et al. (2023) proposed a path-planning method of mobile robot based on an improved genetic algorithm, this method will break the chromosomes of both parents at the same node, and combine the optimal chromosome segments in each segment based on greed, so as to gather all excellent gene segments of both parents as much as possible in one individual, improve the quality of offspring after crossover operation, and accelerate convergence speed.

To address the above-mentioned problems, integrated with the environmental characteristics of regular distribution and many obstacles at the rebar project site, this paper combines industrial robotics, machine vision, path planning, and other technologies to carry out research on the key technology of automated rebar binding at the rebar project site, and develop a rebar binding robot system, which can realize the functions of walking, binding and obstacle avoidance in the flat rebar mesh environment, as well as realize the intelligent visual recognition and localization of rebar crosspoints during the robot's working process. The main contributions of this paper can be summarized as follows:

- (1) In view of the actual situation of the rebar project site and the functional requirements of the rebar binding robot, a robot body structure scheme for planar rebar binding is proposed, including important components such as body platform, upper rebar walking mechanism, lower rebar walking mechanism, binding actuator and image acquisition mechanism, and the working principle of the robot is elaborated.
- (2) A rebar binding crosspoints recognition and localization method combining MobileNetV3-SSD and feature projection curve is proposed for the intelligent recognition demand. The rebar binding crosspoints data set is constructed by data collection and

enhancement, and the MobileNetV3-SSD rebar binding crosspoints recognition model is built based on the TensorFlow deep learning framework to achieve automatic recognition and classification of bound/unbound rebar crosspoints. The extracted unbound crosspoint prediction frame area is image processed, and a localization method combining control factor α and feature projection curve is introduced to achieve localization of unbound crosspoints. The algorithm is evaluated through performance test experiments, and the optimal threshold values of the recognition model and localization method are determined according to the evaluation indexes.

- (3) An improved prioritized reciprocal complete coverage path-planning method and an improved dead zone escape algorithm of A-star are proposed for the full-region coverage binding requirement of the rebar binding robot. The common flat rebar mesh environment is analyzed, and the operational trajectory of the rebar binding robot is theoretically analyzed. A raster-based modeling method is selected to use the improved priority reciprocal complete coverage path-planning algorithm for the traversal coverage of the rebar mesh area. For the problem of getting into the dead zone in the algorithm, the improved A-star algorithm is used for point-to-point path planning to escape the dead zone.

The rest of this paper is organized as follows. Section 2 presents the proposed robot prototype. The recognition and localization method of rebar crosspoints are discussed in Section 3. We present the coverage path planning for robotics in Section 4. Section 5 discusses the results and findings. We conclude this paper in Section 6 with some suggested future directions.

2 | OVERVIEW OF THE PROPOSED ROBOT PROTOTYPE

In this paper, we propose a robot system to bind rebar crosspoints automatically. The hardware system of the robot includes the rebar walking mechanism, binding mechanism, and sensors. The software system consists of the crosspoints detection system, the path-planning system, and the binding control system.

2.1 | Hardware configuration

The hardware system of the rebar binding robot is shown in Figure 1, which is mainly composed of five parts: (1) the robot base and the upper rebar walking mechanism, (2) the lower rebar walking mechanism, (3) the binding mechanism, (4) the control system, and (5) the sensor system.

The rebar mesh is generally uniformly distributed in a horizontal and vertical pattern, and the binding crosspoints are located at the intersection of the mesh. The rebars can naturally serve as the walking track of the robot, completing the walking process.

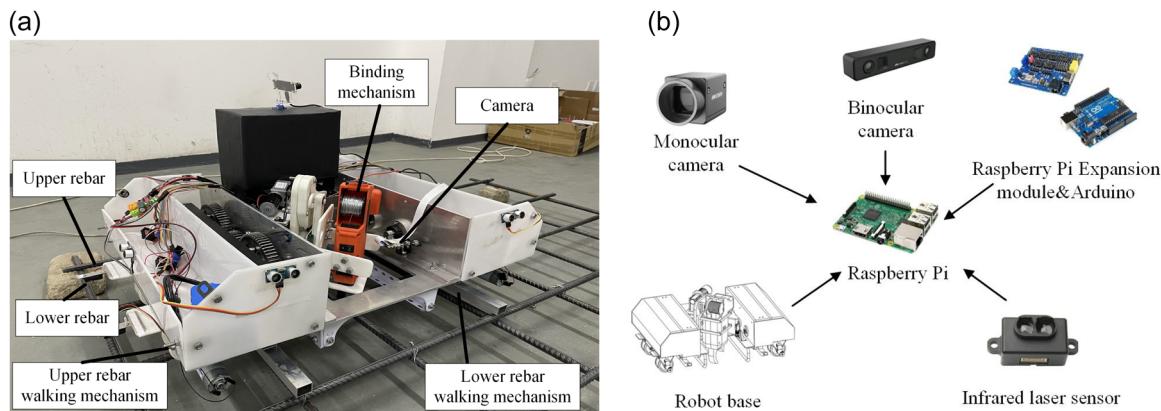


FIGURE 1 (a) Prototype of the proposed robot and (b) robot control and sensor.

Therefore, the walking process of the robot on the rebar mesh is divided into walking in the upper rebar direction and walking in the lower rebar direction, so that the rebar binding robot can move in four directions on the rebar mesh.

The robot uses a self-designed vehicle platform with dimensions of $700 \times 500 \times 500$ mm (length \times width \times height). It is equipped with four motor-driven adaptive wheel assemblies. The wheel motors drive the wheels to drive the whole robot along the upper rebar, and the adaptive connection assembly is used to adjust the spacing between the two wheels at the bottom of the car platform to accommodate different rebar spacing.

The lower rebar walking mechanism uses the flat quadrilateral principle of the planar linkage mechanism, which is oriented at 90° to the upper rebar walking mechanism and mounted in pairs in the side box. Through the lower rebar walking mechanism and the support frame of the car body platform can alternate the robot and drive the whole robot to walk along the lower rebar. The walking directions of the upper rebar walking mechanism and the lower rebar walking mechanism are mutually perpendicular to each other to realize the robot's vertical and horizontal walking on the rebar.

To realize the binding function, a binding mechanism is installed on the robot. Using the binding mechanism drive, the muzzle of the binding machine is aligned with the crosspoints where the rebar needs to be bound, and the rebar crosspoints are bound.

Robots need to work in a wide range of high temperatures for a long period of time, so it is necessary to choose an embedded device with low power consumption, low price, and small size. Commonly used embedded devices are Raspberry Pi and Jetson Nano. Comparing the two, the number of GPIO connectors and memory are the same, the central processing unit (CPU) of Raspberry Pi is a little bit stronger than that of Jetson Nano, but Jetson Nano has a graphics processing unit (GPU) and 128 CUDA cores, and the power consumption of Raspberry Pi is less than 50% of that of Jetson Nano. In terms of power consumption, Raspberry Pi consumes less than 50% of Jetson Nano; in terms of price, Raspberry Pi is 55.5% of Jetson Nano. To summarize, Raspberry Pi is more cost-effective than Jetson Nano and has better power consumption. Considering the

special needs of real-world scenarios, choose the Raspberry Pi. The control system uses Raspberry Pi 4B as the main control core, equipped with Raspberry Pi expansion module and microprocessor Arduino. Raspberry Pi communicates with Raspberry Pi expansion module and microprocessor, respectively, to realize sensor ranging and motor control functions.

The sensor system consists of three kinds of sensor devices, including monocular camera, binocular camera (RGB + RGB), and infrared laser sensor. The monocular camera is mounted on the inside of the body, and the camera lens is parallel to the horizontal plane, about 20 cm away from the rebar, for recognizing the rebar crosspoints. The monocular camera is fixed in the middle right position of the robot, facing directly above the rebar crosspoint. The robot can automatically adjust the determination of binding crosspoints within the camera's field of view based on the current running direction. In order for a rebar binding robot to safely pass through an obstacle without colliding with it, it is usually necessary to obtain obstacle distance information to complete the obstacle avoidance strategy. Binocular cameras are characterized by rich information, high accuracy, and low price. As for the RGBD camera, it uses structured light or TOF to obtain distance information, has a narrow measurement range, a small field of view, and is susceptible to daylight interference, so it is not suitable for use in outdoor environments. In summary, we use a binocular camera (RGB + RGB) for distance measurement. Meanwhile, to determine whether the robot has reached the boundary of the rebar mesh, infrared laser sensors are used to obtain the distance between the robot base and the rebar.

2.2 | Software system

A software system was designed and built for binding path planning, rebar crosspoints detection, and binding. As shown in Figure 2, the software system can be divided into the path-planning system, the rebar crosspoints detection system, and the binding control system according to its functions.

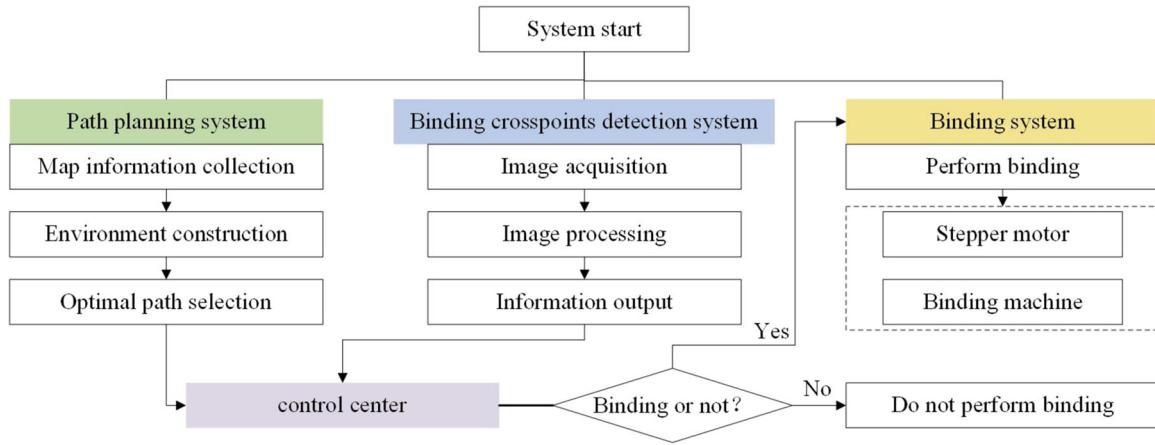


FIGURE 2 Architecture of the robot system.

Among them, the binding path-planning system receives the environmental information detected by the sensor and plans a complete coverage path from the starting point to the ending point. At the beginning of the work, the walking mechanism of the upper rebar fits with the upper rebar, and the motor drives the wheel and the robot to move along the upper rebar. During moving, the rebar crosspoints detection system recognizes whether the crosspoints are bound and transmits the information to the binding control system. The binding control system receives the unbound crosspoints and localization information, compiles the information, and drives the servo motor of the binding mechanism to rotate. In addition, if the binding control system receives the information about the bound crosspoints, the binding process is not performed. When the robot reaches the boundary of a single column of rebar mesh, the motor drives the lower rebar walking mechanism to make the robot span to the adjacent reinforcement column, repeat the above process, and realize the traversal and binding of rebar crosspoints through the planning results of all crosspoints.

3 | RECOGNITION AND LOCALIZATION METHOD OF REBAR CROSSPOINTS

The recognition and localization method of rebar crosspoints is one of the prerequisites and core technologies for the automation of rebar binding. To realize the intelligent recognition and localization of bound and unbound crosspoints during the process of the rebar binding robot, a method of rebar crosspoints recognition and location based on MobileNetV3-SSD and feature projection curve is proposed.

3.1 | Overview

To realize the recognition and localization of the rebar crosspoints, as shown in Figure 3, the robot starts to work with the upper rebar

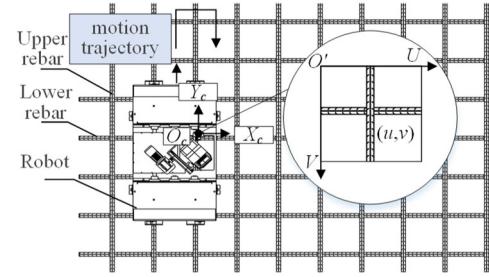


FIGURE 3 Robot working diagram.

walking mechanism and the motor drives the wheels and the robot to move along the upper rebar. During the movement, the vision system recognizes whether the crosspoint is bound or not and transmits the information to the control system. For the unbound crosspoint, the localization method is used to determine the point (u, v) of the crosspoint in the pixel coordinate system, and the (u, v) is converted to the point (x_c, y_c, z_c) in the camera coordinate system, and the conversion relationship is as follows:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}. \quad (1)$$

Where f_x , f_y , c_x , and c_y are the intrinsic parameters of the camera.

From the working principle of the robot, it is clear that the camera and the upper rebar remain constant for the distance along X_c and the height of Z_c . Therefore, the localization problem is transformed into finding v -coordinates, and y_c can be determined by the formula to locate the unbound crosspoint and guide the binding mechanism to perform binding.

On the basis of the above, combined with visual information, we developed a method to recognize and locate the rebar crosspoints. The framework of the rebar crosspoints recognition method is shown in Figure 4. Firstly, the camera is used for fast and

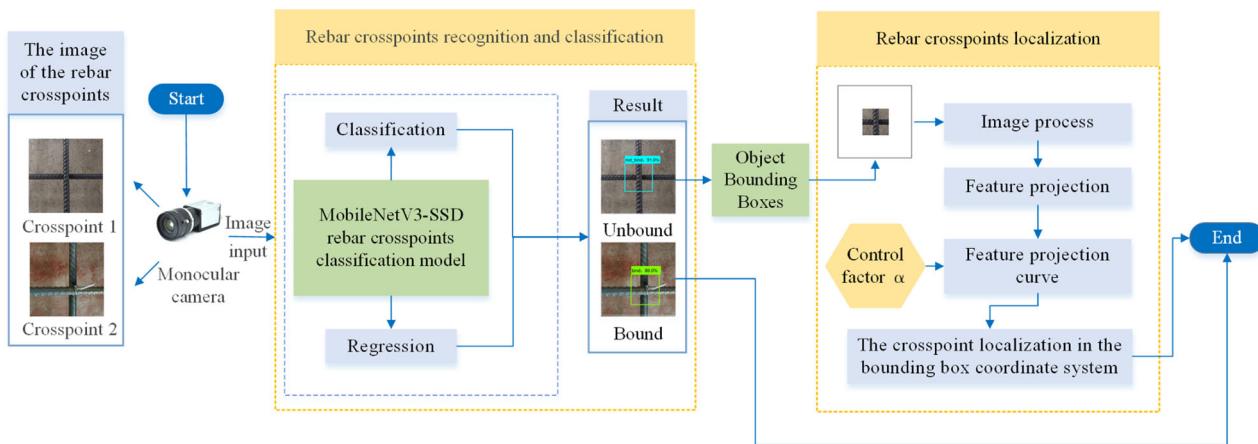


FIGURE 4 Recognition and localization method of rebar crosspoints. SSD, Single Shot Multibox Detector.

noncontact high-quality image acquisition of the target rebar image; then, the image is passed into the MobileNetV3-SSD rebar crosspoints classification model for feature extraction, so that it can use the multiscale feature map for crosspoints classification and bounding box regression to achieve intelligent recognition of bound/unbound crosspoints during the process of the rebar binding robot; finally, for the images with unbound crosspoint, the crosspoint features in the bounding box are further enhanced by combining the image postprocessing method, and the crosspoint localization in the bounding box coordinate system is completed by combining the horizontal feature projection curve and the control factor α . Finally, the original image mapping is performed to obtain the v-coordinate in the pixel coordinate system.

3.2 | Recognition and classification method of rebar crosspoints

3.2.1 | Data set preparation, augmentation, and split

The collection of images was performed in the Key Lab of Structure and Earthquake Resistance. The image acquisition was realized with high-definition digital cameras. The camera was fixed at a distance of 20–25 cm from the rebar, the video was captured in a vertically downward direction, and only one bound/unbound rebar crosspoint was included in a single image. To ensure the comprehensiveness of the sample types, the weather conditions of the collected samples were cloudy and sunny light conditions, and the background contained concrete floors, redwood boards, and so forth, which were consistent with the actual application scenarios. The 2000 images that meet the requirements are screened, and the image resolution is adjusted to 300 pixels \times 300 pixels, and some samples are shown in Figure 5.

The data set is split into three subsets: train, validation, and test, and the data set augmentation operations such as adjusting brightness and rotation are used to augment the train subset and

increase the robustness and generalization ability of the rebar crosspoints recognition model. The data set of rebar crosspoints is shown in Table 1.

3.2.2 | MobileNetV3 rebar crosspoints classification model

Despite the rapid development of architectures and parameters in the study of CNNs, the network models are generally based on high-performance hardware support in the laboratory, which requires high cost in practical engineering applications, and cannot provide recognition conditions for mobile devices. In 2016 Liu et al. (2016) combined the logistic regression idea of the YOLO model and the anchor frame mechanism of the Faster R-CNN model to propose a SSD model, which uses VGG16 as the base network, which improves to a certain extent the prediction accuracy of the model and reduce the complexity of the model. Although the SSD model has a good detection performance, the delay phenomenon still occurs due to the large number of structural parameters of VGG16. Compared with VGG16, the MobileNetV3 is able to greatly reduce the complexity of the model and the number of parameters under the condition of similar accuracy.

To classify the rebar crosspoints to distinguish the bound crosspoints and unbound crosspoints, this paper uses the network MobileNetV3 to extract features and combine it with the SSD detector to reduce the complexity of the model and improve the detection speed of the network. MobileNetV3-SSD consists of three parts: Base Layer, Extra Feature Layer, and Prediction Layer. The structure is shown in Figure 6.

- Base Layer

In this paper, MobileNetV3 is used as the base layer of the rebar crosspoints classification model, which ensures accuracy and reduces the computational cost and parameter count of MobileNetV3.

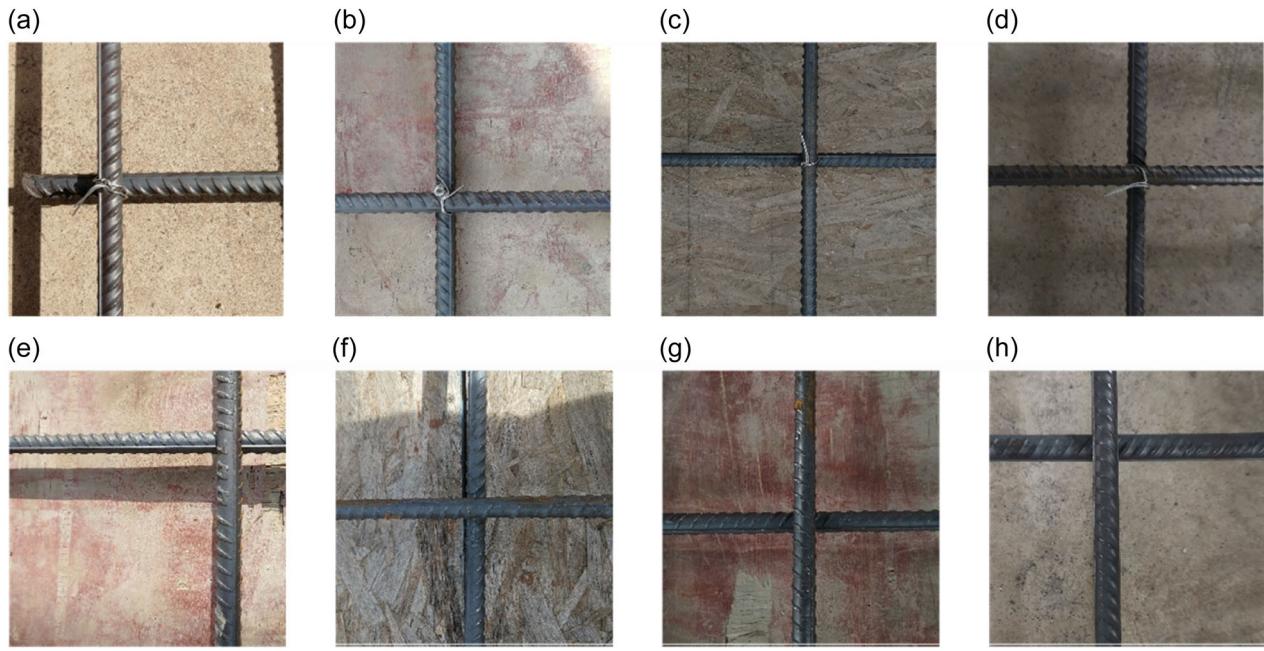


FIGURE 5 (a–d) Bound crosspoints in the data set and (e–h) unbound crosspoints in the data set.

TABLE 1 The data set of rebar crosspoints.

Data set	Number of images	
	Bound crosspoints	Unbound crosspoints
Train subset	1050	1050
Validation subset	300	300
Test subset	300	300

The depthwise separable convolution splits the standard convolution into depthwise convolution and pointwise convolution. The structure is shown in Figure 7.

We assumed that the size of the input feature map is $M_1 \times N_1 \times C_1 \times 1$, where M_1 is the image width, N_1 is the image height, and C_1 is the number of input channels. The feature map with size $M_2 \times N_2 \times C_2 \times 1$ is output by the convolution kernel with size $m \times n$. In the filtering process, the amount of calculation of standard convolution C_n can be expressed as

$$C_n = \sum_{c_1=1}^{C_1} \sum_{c_2=1}^{C_2} \sum_{m_2=1}^{M_2} \sum_{n_2=1}^{N_2} (mn). \quad (2)$$

The amount of calculation of depthwise separable convolution C_d can be expressed as

$$C_d = \sum_{c_1=1}^{C_1} \sum_{m_2=1}^{M_2} \sum_{n_2=1}^{N_2} (mn) + \sum_{c_1=1}^{C_1} \sum_{c_2=1}^{C_2} (M_2 N_2). \quad (3)$$

The ratio of the depthwise separable convolution C_d to the standard convolution C_n computation σ is

$$\sigma = C_d C_n^{-1} = C_2^{-1} + (mn)^{-1}. \quad (4)$$

When the number of output channels C_2 is large, the amount of calculation of depthwise separable convolution is reduced to $(mn)^{-1}$ of the standard convolution, which greatly improves the computation rate.

- Extra Feature Layer and Prediction Layer

Extra feature layer added on top of the base layer that will provide higher-level feature maps. Prediction layer that will locate and identify objects in these feature maps.

MobileNetV3-SSD rebar crosspoints classification model has 29 layers. Among them, Conv1–17 are MobileNetV3 base layer and Conv18–29 are extra feature layers. The outputs of the base layer Conv14, Conv17, and extra feature layers Conv20, Conv23, Conv26, and Conv29 are selected as the prediction layers to provide six different feature maps with sizes of 19×19 , 10×10 , 5×5 , 3×3 , 2×2 , and 1×1 . At the same time, bounding box with different aspect ratios and scales are generated by each pixel on the six feature maps. After that, the recognition result is obtained through nonmaximum suppression.

3.2.3 | Model training

The computer configuration used in this paper is: Intel Core i7-7700HQ CPU and NVIDIA GeForce GTX 1050Ti GPU, Python 3.6.1 programming language version, Tensorflow-GPU 1.14.0 deep learning framework, and OpenCV 3.4.5. In terms of parameter settings,

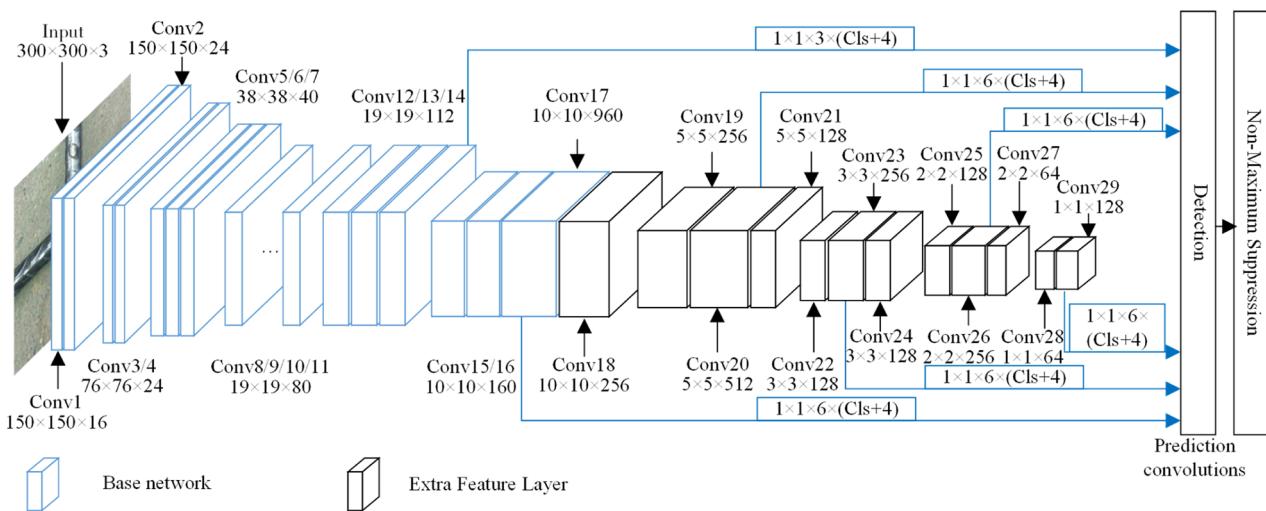


FIGURE 6 MobileNetV3-SSD rebar crosspoints classification model. SSD, Single Shot Multibox Detector.

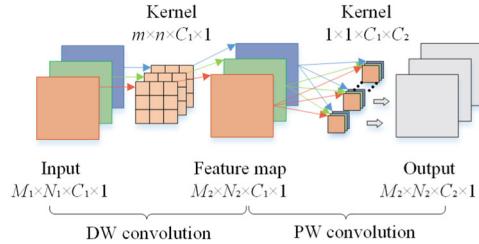


FIGURE 7 The structure of depthwise separable convolution.
DW, depthwise; PW, pointwise.

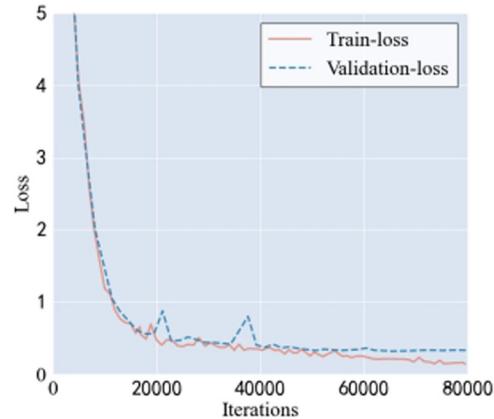


FIGURE 8 The loss of MobileNetV3-SSD. SSD, Single Shot Multibox Detector.

considering the computer performance and the convergence speed of the model, each iteration of the training image is set to 16, the number of iterations is set to 80,000, the momentum factor is 0.9, the weight decay is 0.0005, and the initial learning rate is 0.1. The learning rate update strategy uses the Cosine Annealing algorithm. The loss values of the network on the training and validation subsets are shown in Figure 8.

The loss values of training and validation subsets level off and the final loss value is about 0.15. It indicates that the network can output correct results for most of the samples, and the network model does not overfit during the training process and has good intersection recognition ability.

Meanwhile, to comprehensively evaluate the model performance, we use the Intersection Over Union (IoU), Precision (P), recall (R), and F_1 score for algorithm evaluation calculation.

IoU for two boxes G and D , as defined in Equation (5), is the ratio between the area of the intersection of the two boxes and the area of their union.

$$\text{IoU} = \frac{G \cap D}{G \cup D}. \quad (5)$$

Precision (P), recall (R), and F_1 score are shown in Equations (6)–(8).

$$P = \frac{T_p}{T_p + F_p}, \quad (6)$$

$$R = \frac{T_p}{T_p + F_N}, \quad (7)$$

$$F_1 = 2 \frac{P \cdot R}{P + R}. \quad (8)$$

T_p represents the number of real ground boxes correctly detected, F_N represents the number of real ground boxes which have not been detected, and F_p represents the number of wrong prediction boxes. The above equations are used to calculate the precision, recall, and F_1 score of each category. However, since the

classification in this paper includes bound and unbound crosspoints, the macro average is used as the overall performance indicator to evaluate the results, which is defined as the average value of the calculated index of each category, as shown in Equations (9)–(11):

$$P_{\text{macro}} = \frac{1}{n} \sum_{j=1}^n P_j, \quad (9)$$

$$R_{\text{macro}} = \frac{1}{n} \sum_{j=1}^n R_j, \quad (10)$$

$$F_{1-\text{macro}} = \frac{1}{n} \sum_{j=1}^n F_{1-j}, \quad (11)$$

where the P_j , R_j , and F_{1-j} , respectively, represent precision, recall, and F_1 score of each category, P_{macro} , R_{macro} , and $F_{1-\text{macro}}$, respectively, represent the average value of precision, recall, and F_1 score.

3.3 | Localization method of unbound crosspoints

3.3.1 | Image features of unbound crosspoints

After the recognition of the image of rebar crosspoints, it is necessary to locate the unbound crosspoint of rebar. After testing, it is found that the center coordinates of the prediction box and the coordinates of the actual rebar unbound crosspoint have certain deviations, which cannot meet the accuracy requirements. The object of study in this section is the prediction box image of unbound crosspoint, which has a complex and diverse background and often has uneven illumination due to light and rusting problems. At the same time, the surface texture of the rebar is more complex. Figure 9 shows the results obtained by using conventional image processing methods, such as grayscale processing, image binarization, and edge detection, and it can be seen that the image processing effect is not satisfactory due to the uneven illumination and complex texture features on the rebar surface.

To further obtain the feature information of the rebar surface, we perform feature analysis on the unbound crosspoint prediction box image, as shown in Figure 10. We take the line L_1 passing

through the lower rebar and the line L_2 overlapping with the left boundary of the upper rebar, describe and divide it with lines of different lines, and obtain the gray distribution curve of columns L_1 and L_2 (Figure 10b). By analyzing Figure 10b, we can get the following information: the gray value of the background area changes gently along the L_1 -direction, and the gray value of the thread, rust and light area changes greatly; The gray value of rebar boundary along L_2 -direction changes little; There is a large difference between the gray value of the background area and the boundary of the rebar. In terms of numerical value, the gray value of the former is greater than that of the latter. In conclusion, the distribution of grayscale values on the surface of unbound crosspoint of rebar is not very regular due to the influence of factors such as uneven illumination and interference of texture features. However, according to the characteristics that the gray value difference between the background area and rebar boundary is large, and the former gray value is larger than the latter, this paper adopts the localization method based on the feature projection curve.

3.3.2 | The localization method based on the feature projection curve

To achieve the localization of unbound crosspoints of rebar in the prediction box area, this section performs image processing on the prediction box area and combines the feature projection curve to accurately locate the v-coordinates of unbound crosspoints of rebar. We assume that the feature function of the prediction box image I is $I(x, y)$ and the pixel point coordinates of I are (x, y) . The flow of the localization method is shown in Figure 11, and the steps are as follows.

- (1) *Grayscale and image enhancement:* A weighted average method is used to grayscale the image I to reduce the subsequent operation time and avoid the interference of useless information in the image. The two-dimensional gamma function is constructed according to the brightness distribution characteristics to adaptively correct the image brightness, highlight the difference between the rebar boundary and the background area, and reduce the effect of illumination on the background grayscale.

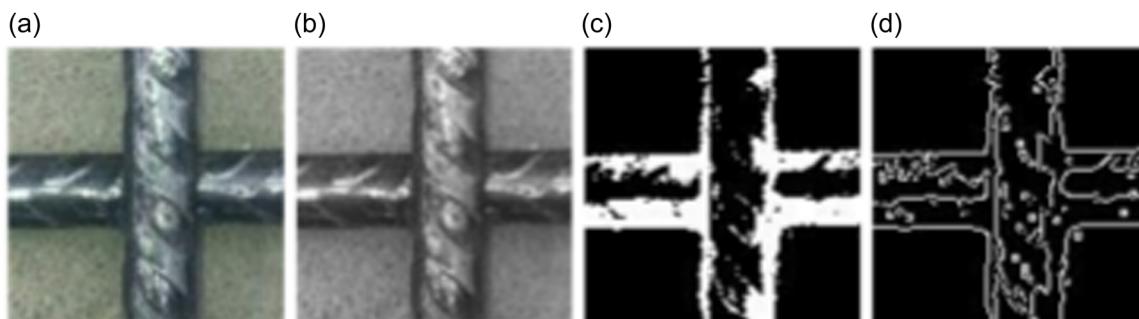


FIGURE 9 (a) Original, (b) grayscale processing, (c) image binarization, and (d) edge detection.

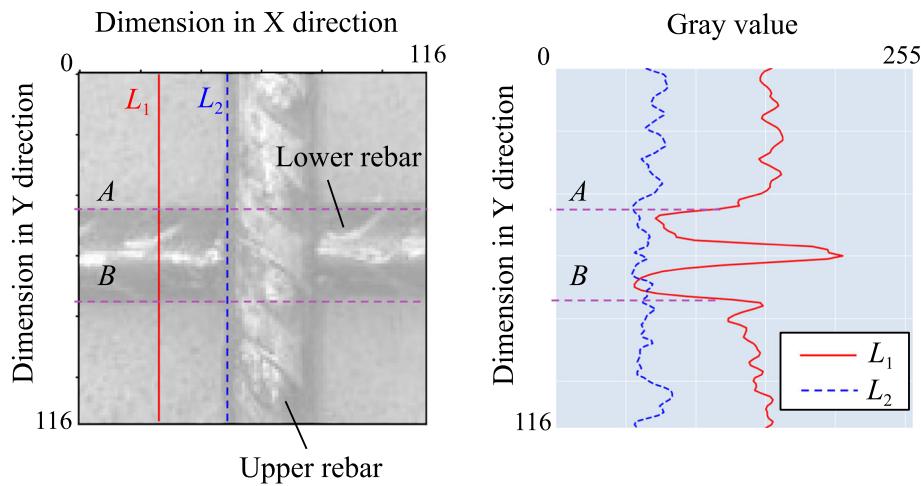


FIGURE 10 (a) Grayscale division of prediction box area and (b) grayscale distribution curve.

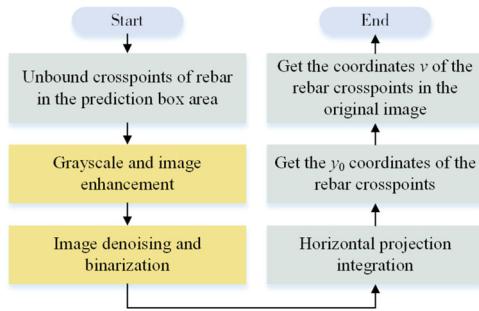


FIGURE 11 The flow of the localization method.

- (2) *Image denoising:* Gaussian filtering (convolution kernel is 3×3) is used to denoise the image, filtering out the useless noise in the image, while preserving the information of rebar edges and minimizing the weakening of edge features in the filtering process.
- (3) *Image binarization:* Segment the rebar region according to the grayscale difference between the background region and the rebar boundary. Since it is difficult to adapt fixed segmentation thresholds to different rebar project sites, the Ostu method is used to automatically select the best segmentation threshold T according to the current image to reduce the amount of image data. We assume that the eigenfunction value $I(x, y) = 1$ for the foreground (rebar) and the pixels are displayed as white, and the eigenfunction value $I(x, y) = 0$ for the background area is displayed as black.
- (4) *Horizontal projection integration:* Figure 12a shows the prediction box position and its binary image. We assume that the width and height of the original image are w_c and h_c , respectively; the prediction box image is w_0 and h_0 . The binary image of the prediction box containing the rebar information is projected along the horizontal direction, and the horizontal density function $P(y)$ of the image is as follows:

$$P(y) = \sum_{x=1}^{w_0} I(x, y), \quad y \in [1, h_0]. \quad (12)$$

The horizontal density function $P(y)$ is expressed as the sum of the eigenfunction values in row y . The horizontal feature projection curve is drawn according to the horizontal density function $P(y)$, as shown in Figure 12b. To determine the boundary of rebar, the projection segmentation threshold k is set. According to multiple tests, it is defined as the multiple of the maximum value of the horizontal density function $P(y)$. The equation is shown in Equation (13), where α represents the control factor:

$$k = \alpha \cdot \max(P(y)). \quad (13)$$

We took α as 0.50, 0.55, 0.60, 0.65, 0.70, 0.75, 0.80, 0.85, 0.90, and 0.95 for testing, and obtained the accuracy of locating unbound crosspoints in sunny, cloudy, redwood, and cement as shown in Figure 13. From Figure 13, it can be seen that the accuracy of unbound crosspoints localization is higher when α takes the value between 0.75 and 0.85. Therefore, the final value of α was determined to be 0.75.

In Figure 12b, we make the $P(y) = k$ intersect the horizontal feature projection curve. Starting from the origin of the coordinate system, there is the first intersection and the last intersection along the positive direction of the horizontal axis of the coordinate. The corresponding Y -coordinates are y_1 and y_2 , which is the boundary of the lower rebar. As shown in Figure 12c, we take the midpoint of y_1 and y_2 to get the y_0 -coordinates of the rebar crosspoints. As shown in Figure 12d, we restore the y_0 -coordinates of the rebar crosspoints in the prediction box area to the pixel coordinate system of the original image, and get the coordinates v of the V -axis direction of the rebar crosspoints in the original image to complete the localization.

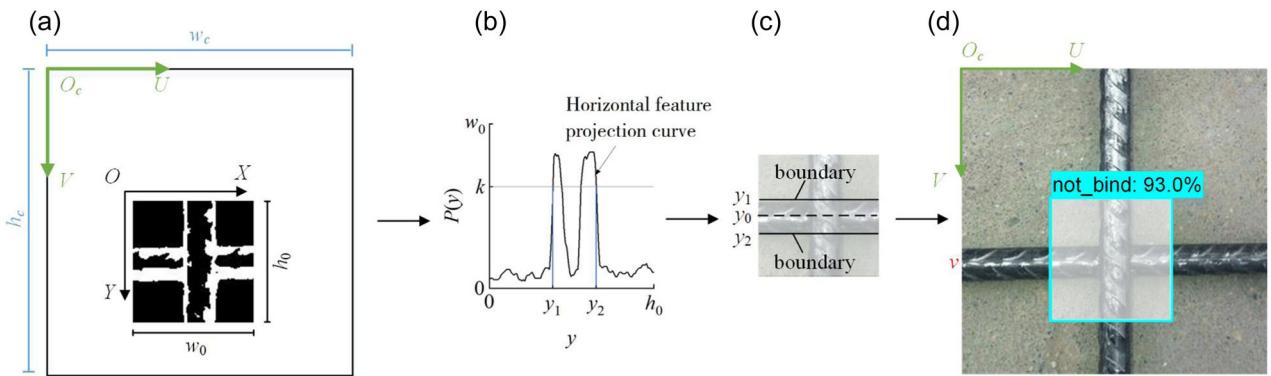


FIGURE 12 (a) Binary image, (b) horizontal feature projection curve, (c) rebar boundary localization, and (d) the localization of the original image.

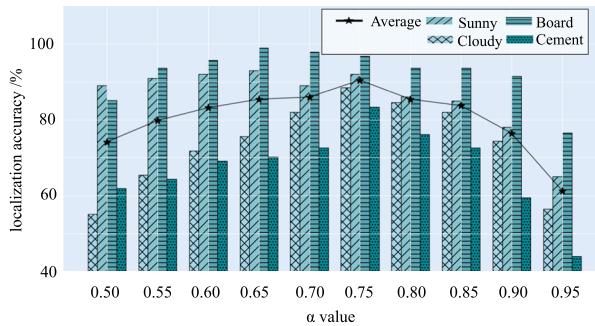


FIGURE 13 Localization accuracy of unbound crosspoints with different control factors.

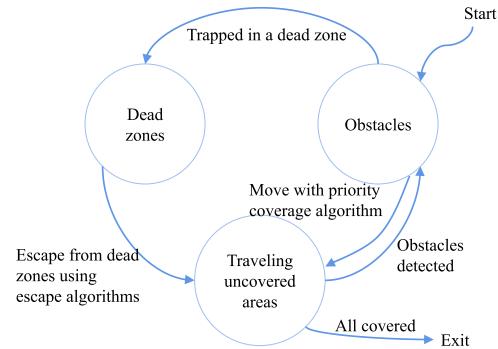


FIGURE 14 The proposed path-planning method.

4 | COVERAGE PATH PLANNING FOR ROBOTICS

The working environment of the rebar binding robot is a regional rebar mesh. There are obstacles such as cement columns and gullies in the rebar mesh, which makes it impossible for the robot to traverse an area in a fixed way, resulting in a complete coverage path-planning strategy to avoid obstacles. This section mainly studies the reasonable planning of the rebar binding route according to the actual task under the condition that the environmental information is known, such as the shape, size, obstacles, crosspoints, and their spacing requirements.

4.1 | Working environment and algorithm framework of robot

To determine the optimal solution for the path of the rebar binding robot, the motion characteristics of the rebar binding robot are first analyzed. The rebar binding robot is mostly driven by the wheel system in the operation process. When working, the robot slides forward in the upper rebar by driving four-wheel systems with the

same walking speed, and after the single column of rebar is bound, the robot starts the lower rebar walking mechanism, at which time the robot needs to overcome gravity to make the robot base be lifted across to the neighboring column of rebar. The robot consumes more power and has longer running time along the lower rebar than along the upper rebar.

An ideal rebar binding path should enable the robot to cover all crosspoint areas of the rebar indiscriminately and only once for the crosspoint areas at a minimum cost (energy consumption and time spent). The algorithm flow is shown in Figure 14. We assume that the robot has three situations: traversing uncovered areas, falling into dead zones, and encountering obstacles. First, the robot starts to move from the corner boundary of the environment. At this time, the environmental boundary is regarded as an obstacle, and the robot moves according to the priority coverage algorithm. When the robot leaves the obstacle area, the robot continues to walk on the rebar mesh according to the priority algorithm. When the robot is surrounded by obstacles and covered areas, the robot falls into a dead zone. At that time, the escape algorithm is used to escape from dead zones. Finally, the robot completes the complete coverage path planning.

4.2 | Path-planning algorithm

4.2.1 | The back-and-forth path (BFP) planning algorithm under priority condition constraints

To translate realistic environments and path-planning problems into a form of representation that can be understood by computers, it is first necessary to model the environment. The representation method of grid map (Ma et al., 2015) is to divide the environment where the robot is located into several grids of the same size. Each component unit represents a part of the environment. According to the corresponding value of each grid, we can determine whether the grid is occupied by obstacles. If a grid is not occupied by obstacles, it is a free grid. The grid type is described by a value of 0, and is usually represented by white squares on the map. If a grid is occupied by obstacles (or part of obstacles), it is an obstacle grid. The grid type is described by a value of 1, which is usually represented by black squares on the map. We assume that the length of the environment is L , the width is W , and the side length of the grid is D , so the total number of grids is $(L \times D/W \times W)$. Due to the similarity between the rebar grid and the grid distribution, and generally square, this paper takes the grid side length D as the rebar grid size. At the same time, the obstacle area is expanded in the grid map according to the actual size of the obstacle, that is, the boundary of the obstacle belongs to the safe area.

According to the motion characteristics of the robot, the motion direction of the robot is defined as up, down, left, and right. At the same time, according to the environment model in the robot-centered environment map window design priority order, determine the priority of domain grid search, so as to determine the coverage path. The information of the grid will be recorded in the matrix row i , column j , defined as $G(i, j)$.

This paper designs three priority conditions for path planning: (1) The priority in the direction of the starting point is higher than that in the direction away from the starting point. (2) Under the condition of meeting 1, the priority of horizontal direction is higher than that of vertical direction. (3) Under the condition that the above two rules are met, if there is no grid to move freely in the moving direction, the priority condition strategy will be stopped, otherwise, the algorithm priority condition will be followed.

Figure 15 shows the schematic diagram of the BFP planning algorithm under priority condition constraints, and I-IV in the figure indicate the priority order of grid traversal, that is, I is the first, II is the second, and so on. When the robot is at position 1, the priority order is established in the planning window according to the above algorithm, as shown in the planning window 1 in the figure. At this time, the priority of the upper grid of the robot is the highest, so the robot moves to position 2. When the robot is at position 2, the priority order is established as shown in the planning window 2, and the robot's right grid has the highest priority, and the robot moves to position 3, and so on.

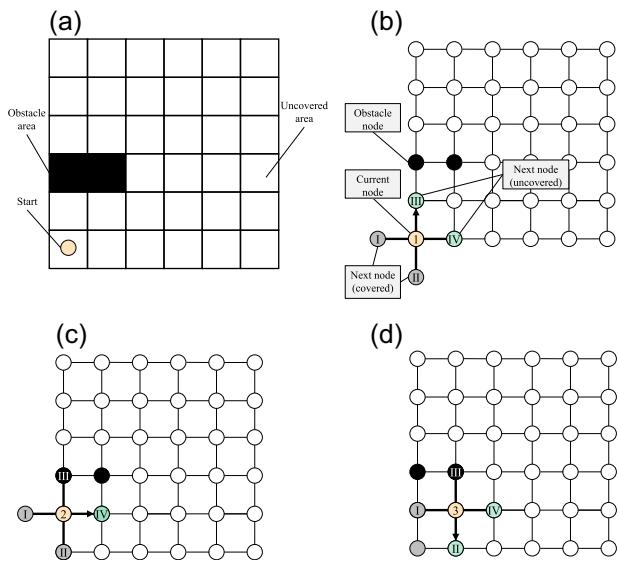


FIGURE 15 (a) An example grid map, (b) planning window 1, (c) planning window 2, and (d) planning window 3.

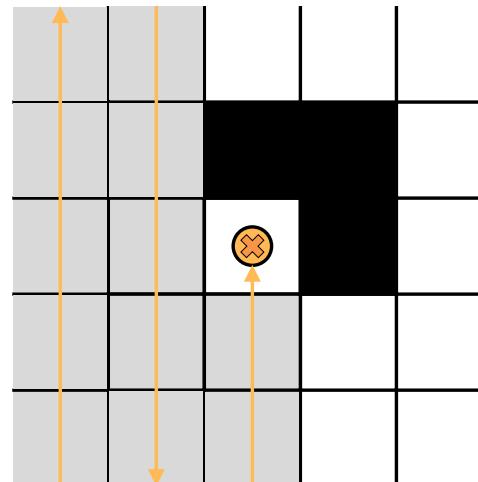


FIGURE 16 Trapped in the dead zone.

4.2.2 | A dead zone escape method based on improved A* algorithm

In the process of planning the path of the rebar tying robot, it sometimes enters the dead zone position, that is, the grid around the current rebar binding robot is occupied by obstacles or areas that have been traversed, as shown in Figure 16.

After the robot is trapped in the dead zone, the BFP planning algorithm under priority condition constraints can no longer be adopted, so the robot should escape the dead zone with the optimal path as soon as possible. Therefore, A* algorithm (Guruji et al., 2016; Li et al., 2022) is introduced to avoid invalid search paths to a certain

extent and improve search efficiency. The cost function of a node in A* algorithm is expressed as

$$f(n) = g(n) + h(n), \quad (14)$$

where $f(n)$ represents the estimation function from the initial node to the target node, $g(n)$ represents the actual cost from the initial node to n node, and $h(n)$ represents the estimated cost of the best path from n node to the target node. The selection of the above functions directly determines whether the algorithm can find the optimal solution.

We take the depth from the initial node of $g(n)$ to the n node, as shown in Figure 17. The initial node is node 1, then the $g(n)$ value of node 4 is 1, the $g(n)$ value of node 8 is 2, and so on. $h(n)$ are shown in Equation (15), where n_s represent the initial node, n_e represent the target node.

$$h(n) = |n_s(x) - n_e(x)| + |n_s(y) - n_e(y)|. \quad (15)$$

When the estimation function is calculated using the traditional A* algorithm, the four subnodes around the node based on the node do not produce penalty values. However, in the actual motion of the robot, if the robot changes direction frequently, it will inevitably increase the energy loss of the robot, and the energy loss of the robot in the upper rebar direction is smaller than that in the lower rebar direction.

To reduce the energy loss caused by frequent direction changes, a penalty function $o(n)$ is added to the valuation function, and the penalty value is determined by calculating the orientation relationship between the parent and child nodes. The improved A* algorithm estimation function is as follows.

$$f(n) = g(n) + h(n) + o(n), \quad (16)$$

$$o(n) = \begin{cases} 0, & x_{node} - x_{next} = 0, \\ k \times (g(n) + h(n)), & x_{node} - x_{next} \neq 0, \end{cases} \quad (17)$$

where $f(n)$ represents the penalty time factor coefficient, which is taken as 0.3 in this paper, x_{node} is the horizontal coordinate of the parent node, and x_{next} is the horizontal coordinate of the child node; when the parent node and the child node are in the same upper rebar direction, that is, the horizontal coordinates are the same, the penalty function $o(n)$ is 0. Conversely, the

horizontal coordinates are different, and the penalty function $o(n)$ is added.

5 | EXPERIMENTS

To evaluate the effectiveness of the proposed algorithm, this paper establishes a platform for the rebar binding robot and experimentally verifies the mechanical structure motion, vision detection algorithm, and path-planning algorithm of the robot, respectively.

5.1 | Motion experiment of rebar binding robot

In this paper, the design guidelines of the rebar binding robot are based on modularization to do the implementation of the function, according to the modularization into the System integration and testing, the following will explain the structure of the control test of the operating function module.

The motion experiment of the rebar binding robot is shown in Figure 18. The robot switches between the upper and lower rebar walking mechanism to complete the driving action across the rebar mesh, and starts the real-time rebar binding work from the first column of rebar in the lower left corner in the square rebar mesh area as a unit. At the initial point of the column, the upper rebar traveling mechanism is attached to the upper rebar, and the motor drives the wheels and the robot to move along the upper rebar. At the same time, the ultrasonic sensors around the rebar binding robot and the binocular vision sensors mounted on the top detect whether there are obstacles around the robot in real time; the monocular camera detects whether there are rebar binding crosspoints in real time; and the infrared laser ranging sensors of the edge detection detect whether it reaches the edge in real time.

If an obstacle is detected around the rebar binding robot, the obstacle avoidance operation is performed, and the direct current (DC) geared motor traveling on the upper and lower rebars moves and stops to control the rebar binding robot to avoid obstacles through forward and reverse rotation; if no obstacle is detected, the obstacle avoidance operation is not performed, and the rebar binding robot continues to advance along the upper rebar; If the edge infrared laser ranging sensor detects that the edge has been reached, the upper rebar traveling DC gear motor stops, at which time the lower rebar traveling DC gear motor is driven so that the rebar binding robot crosses over to the neighboring upper rebar as a whole and travels a preplanned path.

The robot performs the binding as shown in Figure 19. If the monocular camera detects a rebar binding cross crosspoint in real time, the upper rebar traveling DC gear motor stops moving to stop the rebar binding robot, and the DC gear motor of the rebar binding machine rotates positively to realize binding of the binding crosspoint, and at the end of the binding, the DC gear

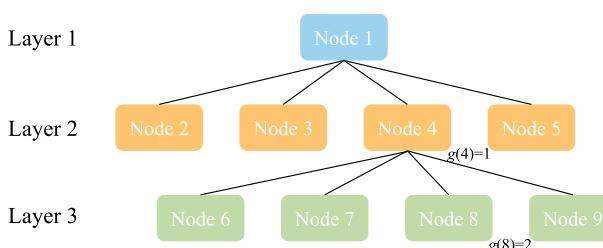


FIGURE 17 Spatial depth of $g(n)$.

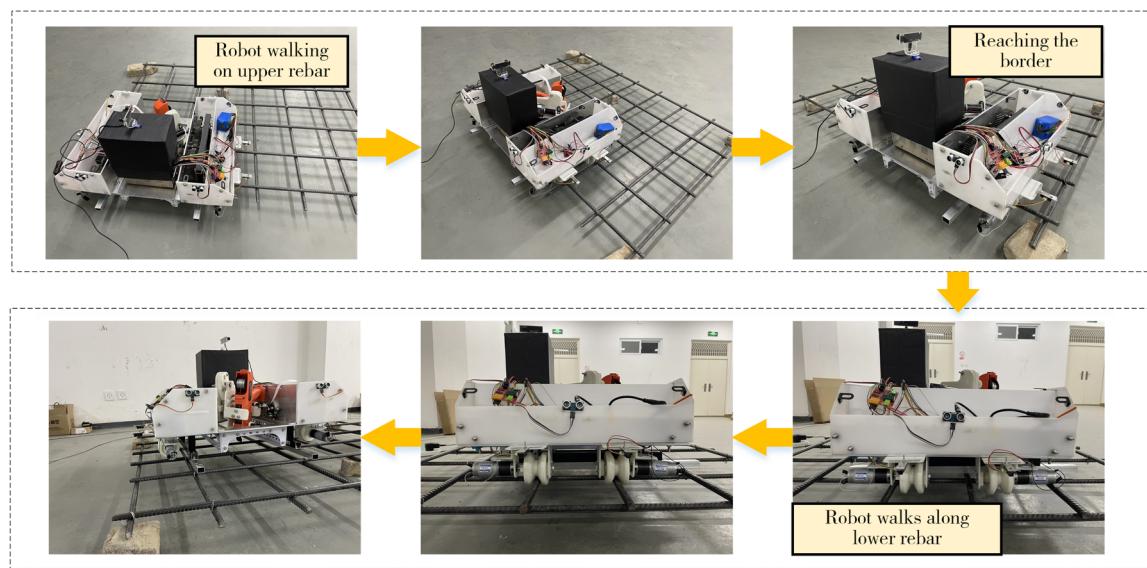


FIGURE 18 The motion experiment of rebar binding robot.

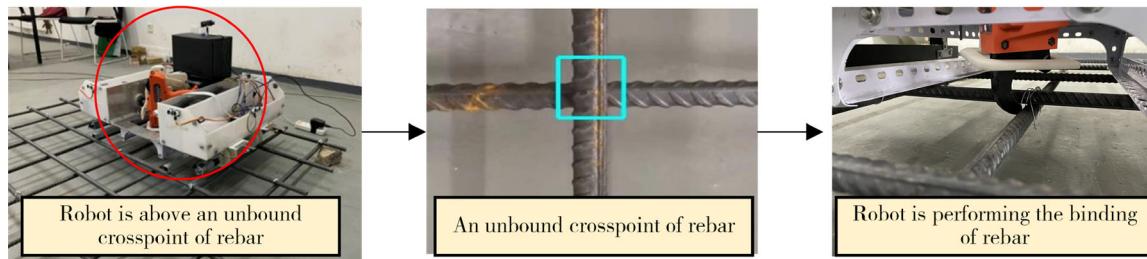


FIGURE 19 Rebar binding robot performs visual recognition and binding of rebar crosspoints.

motor of the rebar binding machine reverses back to its original position, and the DC gear motor of the upper rebar traveling continues to move, and the rebar binding robot advances along the upper rebar; and if there is no detection of the rebar binding crosspoint, the rebar binding robot continues to advance along the upper rebar.

To better understand the working performance of our robot, we compared the robot with manual binding and handheld binding machines. A rebar mesh of $1.5\text{ m} \times 1.2\text{ m}$ was selected for the experiment. The distance between the rebar mesh was $0.2\text{ m} \times 0.2\text{ m}$, and the number of effective binding crosspoints for the robot was 24. We set the start time of binding as the time when robots or humans start binding, and take the completion of crosspoint binding as the end time. The time consuming of 3 methods and 24 crosspoints is calculated. To reduce the randomness of binding, we conducted three experiments on the three methods and took their average values. Table 2 shows the differences between the various rebar binding methods. We counted the time taken to complete the binding of rebar crosspoints under different conditions. After comparison, we found that manual binding and handheld binding machines cannot perform continuous work, and the time to bind a single crosspoint of

our robot is less than the remaining two methods, which indicates that our designed robot has some advantages.

5.2 | Recognition and localization experiment of rebar crosspoints

5.2.1 | The analysis of rebar crosspoints recognition and classification

To further verify the reliability of MobileNetV3-SSD for recognition of rebar crosspoints, the VGG16-SSD and the Faster-R-CNN were compared with MobileNetV3-SSD. The same training platform and configuration information of MobileNetV3-SSD were used for training, and the comparison was made on the same data set.

Table 3 shows the parameter statistics of the recognition effect of the four methods for rebar crosspoints. In terms of the number of model parameters, the size of the MobileNetV3-SSD model is 9.10 MB, which is reduced by 89.39% compared with VGG16-SSD, and reduced by more than 90% compared with the Faster R-CNN and YOLOV3 models. In terms of recognition effect, the P_{macro} and

TABLE 2 Comparison of rebar binding methods.

Type	Running speed (m/s)	Continuous working time (h)	Binding time (s)	Single binding time (s)
Manual binding	-	Depends on physical strength	97.92	4.08
Handheld binding machine	-	Depends on physical strength	69.12	2.88
Our robot	Upper rebar (0.053 m/s); Lower rebar (0.029 m/s)	Depends on battery capacity	50.4	2.10

TABLE 3 Comparison between different recognition models.

Model	P _{macro} (%)	R _{macro} (%)	Time (ms)	Size (Mb)
VGG16-SSD	86.30	88.00	76	85.70
MV3-SSD	95.40	96.70	24	9.10
Faster-R-CNN	96.64	96.82	160	432.79
YOLOV3	80.89	82.10	89	206.50

Abbreviations: R-CNN, Region-based Convolutional Neural Network; SSD, Single Shot Multibox Detector; YOLO, you only look once.

R_{macro} of MobileNetV3-SSD are improved by 9.10% and 8.70%, both of which are higher than the original model, but the accuracy is slightly lower than that of Faster-R-CNN. The reasoning speed of the image is reduced by 52 ms compared with VGG16-SSD and 65 ms compared with the YOLOV3 model, which meets the requirements of small equipment requiring small floating point calculation and high precision. MobileNetV3-SSD is more suitable for deployment on mobile robots and has good real-time performance.

Meanwhile, the IoU and confidence values can have some influence on the model performance. The IoU is an important factor that directly affects the results. Table 4 shows the changes in various evaluation indexes under different IoU thresholds, and it can be found that the P_{macro} and R_{macro} of the model decrease as the IoU threshold rises. At the IoU thresholds of 0.3 and 0.4, the model has a higher F_{1-macro}, but the results are not satisfactory due to the small intersection ratio of the prediction box and the ground truth box. On the other hand, when the threshold value is 0.6, the ability of the model to recognize crosspoints drops sharply by 2.1% and continues to drop with the increase of the threshold value, and the recognition performance is poor. Since the purpose of the experiment is to recognize the existence and distribution of crosspoints, the requirement of the intersection is not high. In summary, the IoU = 0.5 was chosen as the criterion to determine whether the crosspoints were correctly recognized. Table 5 shows the changes of various evaluation indexes under different confidence values. As the confidence value keeps increasing, P_{macro} increases, R_{macro} decreases, and F_{1-macro} keeps increasing. When the confidence value is greater than 0.5, the F_{1-macro} decreases. In this paper, to obtain higher accuracy, the confidence value is taken to be 0.5, when P_{macro} and R_{macro} reach 95.40% and 96.70%.

Table 6 shows the statistical parameters related to the crosspoint recognition results under different weather conditions, and Figure 18 shows the recognition of some samples. From the results, under

TABLE 4 Comparison of recognition results under different Intersection Over Union (IoU).

IoU	P _{macro} (%)	R _{macro} (%)	F _{1-macro} (%)
0.3	95.70	97.00	96.40
0.4	95.70	97.00	96.40
0.5	95.40	96.70	96.00
0.6	93.30	94.50	93.90
0.7	85.00	83.90	84.40

TABLE 5 Comparison of recognition results under different confidence thresholds.

Confidence thresholds	P _{macro} (%)	R _{macro} (%)	F _{1-macro} (%)
0.3	94.30	96.70	95.50
0.4	95.10	96.70	95.90
0.5	95.40	96.70	96.00
0.6	95.80	95.30	95.60
0.7	96.50	92.80	94.60

sunny conditions (Figure 20a), the precision and recall rates of bound crosspoints are 97.20% and 91.30%, and the precision and recall rates of unbound crosspoints are 95.50% and 98.00%, respectively; under cloudy conditions (Figure 20b), the precision and recall rates of bound crosspoints are 96.10% and 98.00%, and the precision and recall rates of unbound crosspoints are 93.10% and 99.30%, respectively. The effects of uneven illumination and high-intensity illumination lead to a low recall rate of crosspoint recognition. The effects of noise and low-intensity illumination cause insufficient model feature learning and low precision rate. The F_{1-macro} under sunny and cloudy conditions are 95.45% and 96.55%, indicating that MobileNetV3-SSD has some adaptability to light unevenness and noise interference.

The experiment also analyzed images without accurately identifying categories. Figure 20c shows an image of a typical misidentified rebar binding crosspoint. It can be observed that under conditions of good lighting and simple background, the model still misrecognizes crosspoint types. In the first two of Figure 20b, although their confidence scores reached 0.84 and 0.86, the category discrimination was incorrect. The reason for this is that there is a

certain degree of reflection on the surface of the rebar, while there are transverse and longitudinal ribs on the surface of the threaded steel, which make the threads and excessively fine tie wires approximately mixed under uneven light conditions. The confidence scores of the last two in Figure 20b are 0.68 and 0.57, respectively. Although the prediction box identifies based on the characteristics of the rebar crosspoint, it is currently unknown whether the wire tie part exists due to the node being located at the edge of the field of view. In fact, it is not possible to correctly determine the crosspoint type, so the actual confidence level will be lower, which can easily lead to misjudgment. However, due to only a few captured images being at the edge of the field of view, it did not have a significant impact on the recognition results.

On the basis of the above analysis, it can be concluded that the external conditions that have the greatest impact on the accuracy of rebar crosspoint recognition are weather conditions, while the internal conditions are due to the fact that the rebar crosspoint image is located at the edge of the camera's field of view. Due to the complex conditions of sunlight exposure on actual sites, it is recommended to choose uniform or shaded areas for identification as much as possible when conditions permit; The camera's field of view also has a significant impact on the

recognition of steel bar nodes. It is recommended to remove the recognition of rebar crosspoints along the edge of the field of view.

Although the generalization ability can be improved to a certain extent under the model in this article, there are still differences in the recognition accuracy of rebar crosspoints under different conditions. Therefore, to achieve better recognition results, it is recommended to obtain training sets in more binding scenarios and continuously expand them to effectively improve the generalization of the model.

5.2.2 | The analysis of rebar crosspoints localization

In this section, 178 images of unbound crosspoints with positive crosses 20 cm from the crosspoints are taken for localization test, 78 and 100 images for sunny and cloudy days, respectively, 84 and 94 images for redwood and cement, respectively, and the localization process is shown in Figure 21.

To verify the effectiveness of this unbound crosspoints localization method, the unbound crosspoints localization method proposed in this paper is compared with the localization method based on the

TABLE 6 Crosspoints recognition results under different weather conditions.

Weather condition	Crosspoints classification	T_p	F_p	F_N	P (%)	R (%)	F_1 (%)	$F_{1-macro}$ (%)
Sunny day	Bound	137	4	13	97.20	91.30	94.20	95.45
	Unbound	147	7	3	95.50	98.00	96.70	
Cloudy day	Bound	147	6	3	96.10	98.00	97.00	96.55
	Unbound	149	11	1	93.10	99.30	96.10	

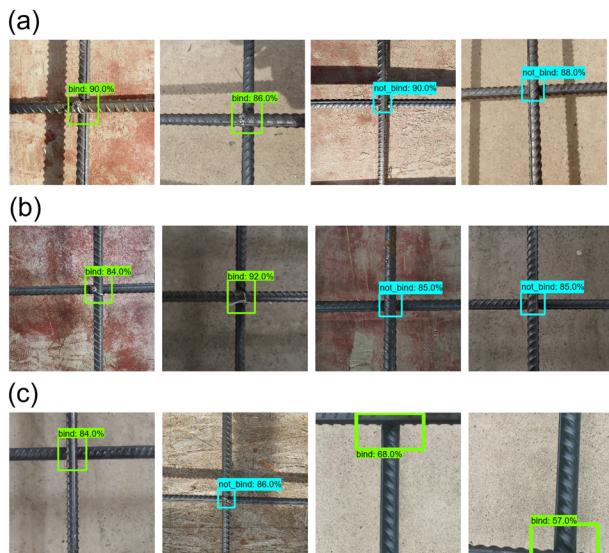


FIGURE 20 (a) Sunny day, (b) cloudy day, and (c) error recognition.

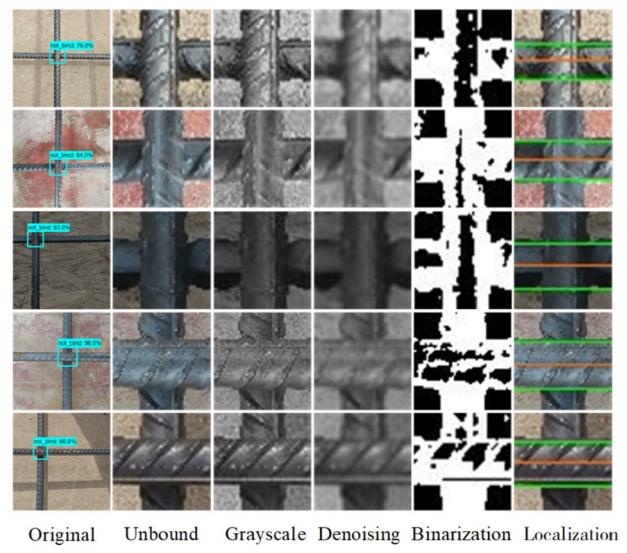


FIGURE 21 The localization process of unbound crosspoint.

Hough transform. As shown in Figure 22, due to the complicated texture features on the surface of the rebar, too many texture features led to the failure of the Hough algorithm, resulting in the wrong recognition of the horizontal and vertical ribs of the rebar as boundaries.

Table 7 shows the comparison of the localization accuracy between the proposed method and the Hough method. The results show that the localization accuracy of the proposed method for locating unbound crosspoints is 90.45%, and the localization accuracy of the Hough algorithm is 38.76%. The proposed method can locate the unbound crosspoints under the complicated surface characteristics of the rebar, and has high localization accuracy and speed.

While verifying the localization accuracy, we calculate the localization error of unbound crosspoints based on the real horizontal coordinates of the unbound crosspoints and the theoretical horizontal coordinates output by the localization method. In this section, the prediction box image coordinate system X-Y is used to find y_0 for the test, and 15 images are randomly selected for the test, and the results are shown in Figure 23. The average localization error between the actual coordinates and the algorithm coordinates is 1.43 pixels, and the average relative error is 6.15%, which can meet the binding requirements of the rebar binding robot.



FIGURE 22 Comparison of localization algorithms.

TABLE 7 The comparison of the localization accuracy between the proposed method and the Hough method.

Algorithms	Number of correctly located samples	Total samples	localization accuracy (%)	Time (ms)
Ours	161	178	90.45	6.94
Hough	69	178	38.76	50.00

5.2.3 | Recognition and localization experiments on robots

To test the speed and overall performance of the detection and localization models, we performed real-time tests on a simulated rebar binding site using a monocular camera placed on the robot. Much of the machine learning and AI research currently being conducted in the construction field uses large, expensive equipment to perform computation, detection, and classification. Yet the cost factor is a major consideration when deciding on the deployment of automation solutions at construction sites. Therefore, we are interested in minimizing costs while maximizing efficiency. CPU performance in performing machine learning tasks is typically lower, and large models based on R-CNN algorithms used in previous research result in significantly longer latencies. The model proposed in this paper is based on the lightweight MobileNetV3-SSD algorithm and is able to run on weaker devices (CPUs) with acceptable performance. We use a Raspberry Pi-based main controller and use a CPU to run on it. At the same time, a monocular camera is mounted on the robot, which runs over the rebar and records its real-time inspection results.

The robot detection and localization in different weather is shown in Table 8. A screenshot for a live video feed is shown in Figure 24. Under the actual environmental conditions, it can also achieve a high recognition rate, and can effectively

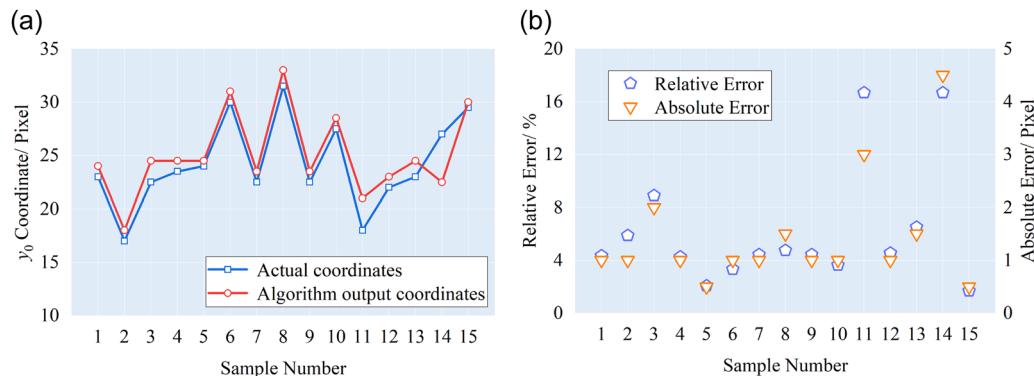


FIGURE 23 (a) Actual coordinates and algorithm output coordinates and (b) absolute error and relative error.

distinguish between the bound crosspoints and unbound crosspoints, which can meet the requirements of the specification for rebar binding.

The robot needs to recognize the bound crosspoints, and also needs to localize the crosspoints to carry out the binding. For this purpose, we localized the images on the basis of crosspoint recognition. We compared the actual position of the crosspoints with the theoretically derived position of the algorithm, and the statistical results are shown in Table 9. It can be seen that the localization algorithm in this paper is able to accurately locate the unbound crosspoints under different weather conditions, and the localization model has high robustness even under complex lighting conditions and different backgrounds.

5.3 | Path-planning experiment of rebar crosspoints

To test the effectiveness of BFP planning and dead zone escape algorithm based on improved A*, this paper uses simulation experiments to verify the algorithm. The simulation environment consists of 15×15 and 20×20 environments, which contain obstacles such as columns, doors, and windows, with a number of free grids of 175 and 336, respectively.

Figure 25 shows the result of robot path planning using the path-planning algorithm, where the size of the environment map is

TABLE 8 Robot recognition and localization in different weather.

Weather condition	Detection Accuracy (%)	Classification Accuracy (%)	Time (ms)
Sunny day	85.70	93.77	292
Cloudy day	87.33	95.03	280
Average	86.52	94.40	286

20×20 . The black grid represents obstacles. The lower left corner $(0, 0)$ of the grid map is the starting point, and the lower right corner $(19, 0)$ is the endpoint. The robot traverses the grid map through the priority BFP algorithm. When the robot moves through $(3, 14)$, $(2, 14)$, and $(1, 14)$ in turn, the surrounding areas are obstacles, boundaries and traversed grids, and the robot falls into a dead zone. At this time, the robot uses the dead zone escape algorithm. The robot moves to the grid that has not been traversed recently $(5, 13)$ to escape the dead zone. After that, we continue to use the priority BFP algorithm to traverse until we complete the whole region traversal.

Tables 10 and 11 show the performance metrics of the complete coverage path-planning algorithm in 15×15 and 20×20 maps. It can be found from the table that although all three can guarantee 100% coverage rate, there are certain differences in the number of dead zone, number of turns, and track repetition rate. The simple BFP planning method is lower than the other two methods in each index. Although the priority BFP planning method is equal to this paper's algorithm in terms of trajectory repetition rate, this paper's algorithm reduces the number of turns by 2 and 4 compared with the 15×15 and 20×20 maps, which verifies the effectiveness of this paper's proposed priority BFP with the dead zone escape method based on improved A*.

Meanwhile, the robot performs path-planning experiments on a rebar mesh, which contains obstacles of a certain size. To avoid the robot colliding with the obstacles, we inflate the obstacles in the grid map. Its path planning is shown in Figure 26a. The robot encounters an obstacle and escapes from it, as shown in Figure 26b,c. The results show that the robot's trace matches the expectation and can completely complete the traversal of the rebar mesh.

TABLE 9 Localization results on the robot.

Weather condition	Localization accuracy (%)	Position error (%)
Sunny day	88.94	16.67
Cloudy day	92.03	8.89
Average	90.49	12.78

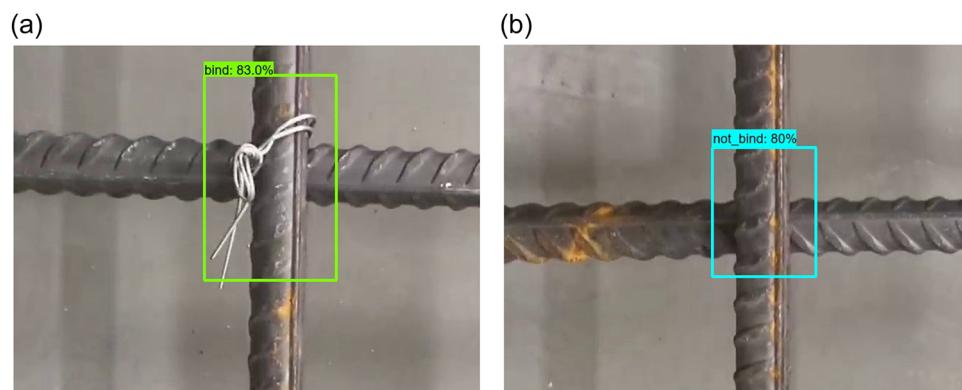


FIGURE 24 (a) Bound crosspoints recognized on the robot and (b) unbound crosspoints recognized on the robot.

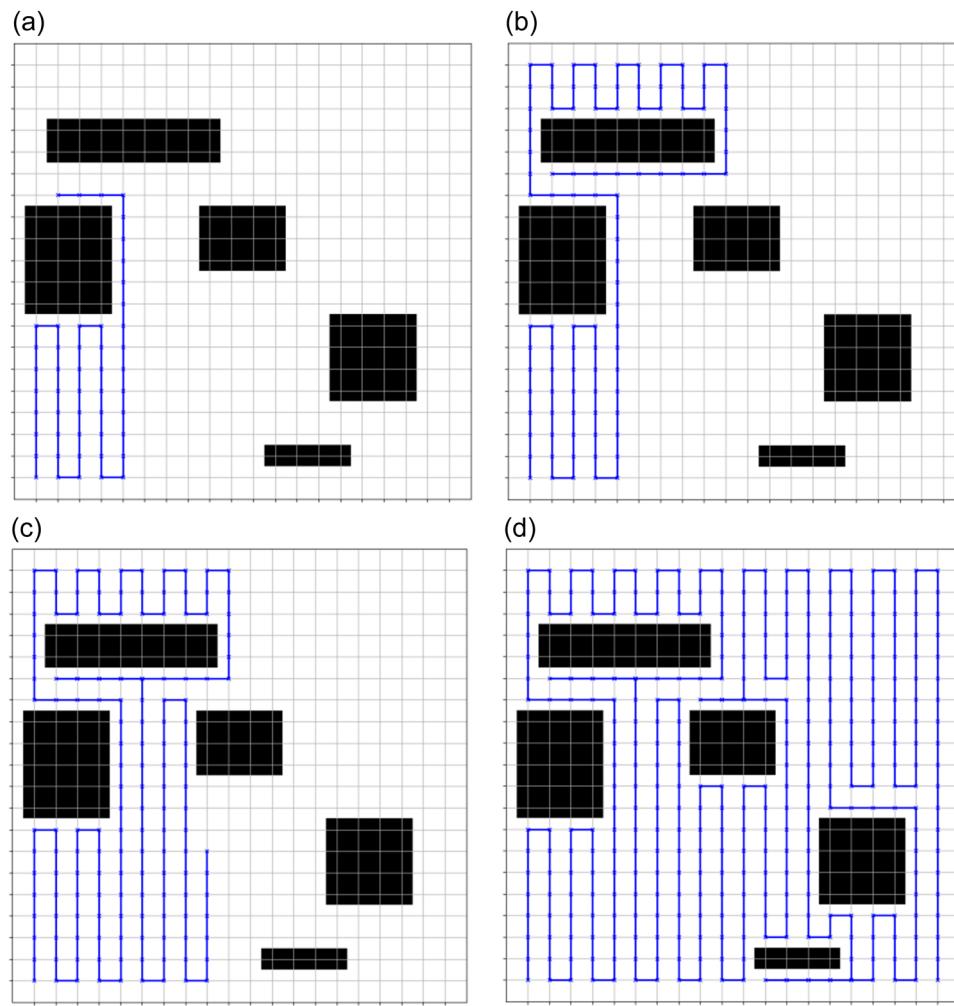


FIGURE 25 (a) Start complete coverage path planning, (b) fall into the dead zone, (c) escape the dead zone, and (d) complete the whole region traversal.

TABLE 10 The performance metrics of the complete coverage path-planning algorithm.

Grid map	The number of dead zone			The number of turns		
	Priority		Proposed	Priority		Proposed
	BFP	BFP		BFP	BFP	
15 × 15	2	1	1	58	46	44
20 × 20	4	3	3	80	78	74

Abbreviation: BFP, back-and-forth path.

5.4 | Field test

To verify the feasibility of the proposed rebar crosspoints detection method, this section deploys the recognition and classification method on the rebar binding robot prototype for application.

The model proposed in this article is the lightweight network MobileNetV3, which can run on devices with weaker performance (CPUs). We use a Raspberry Pi-based main controller and use a CPU

to run the robot. At the same time, a monocular camera is installed on the robot, which runs on the steel bars and records real-time detection results. Robot recognition is shown in Figure 27.

The experimental scenario is selected as a laboratory site, and the experimental platform mainly includes a prototype, a rebar mesh composed of 20 mm diameter steel bars (with a spacing of 200 mm), and obstacles. The experimental site is shown in Figure 28.

The real-time recognition situation on site is shown in Figure 29. The confidence score of the unbound crosspoint in the figure is 80%, and it can accurately locate the crosspoint's position and output its y-direction coordinates. The camera of the rebar binding robot was used to collect and recognize images, and a total of 150 images of rebar crosspoints were tested. Among them, 131 images of rebar crosspoints were accurately identified, with a recognition accuracy of 87.33%. The average time for a single image was 280ms. Under simulated environmental conditions, deploying the detection model on robots can achieve high accuracy and effectively distinguish between bound and unbound crosspoints, meeting the requirements of steel bar binding regulations.

TABLE 11 The performance metrics of the complete coverage path-planning algorithm.

Grid map	Coverage rate			Track repetition rate		
	BFP (%)	Priority BFP (%)	Proposed (%)	BFP (%)	Priority BFP (%)	Proposed (%)
15 × 15	100	100	100	3.43	2.29	2.29
20 × 20	100	100	100	3.62	2.41	2.41

Abbreviation: BFP, back-and-forth path.

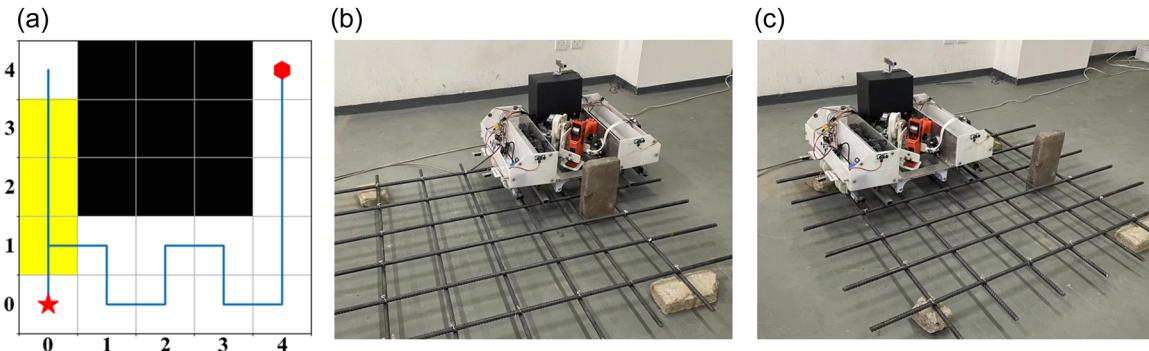


FIGURE 26 (a) Coverage path planning for robots, (b) robot encounters an obstacle, and (c) robot escape from obstacles.

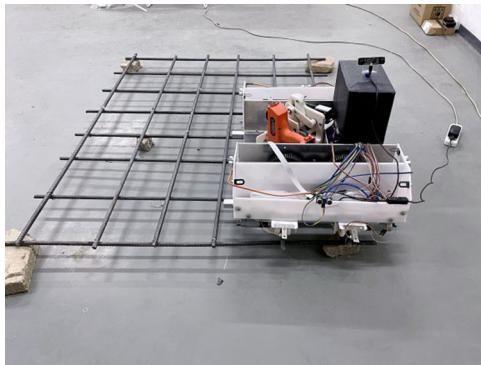


FIGURE 27 Robot prototype and crosspoints recognition.



FIGURE 28 Experimental site.

6 | CONCLUSIONS

To solve the problems of low efficiency in the process of manual rebar binding, a mobile robot vision detection and path-planning method for rebar crosspoints is proposed by combining industrial robot, machine vision and path-planning technologies, and the main contributions of this study are the following:

- (1) A rebar binding robot with independent recognition and binding ability is developed and designed by integrating multiple types of sensors. The robot includes the robot base, the visual recognition system, and the path-planning system, and realizes unmanned rebar binding under the rebar engineering site.

- (2) The rebar crosspoints recognition model and feature projection curve localization method are proposed to realize rapid recognition and localization of rebar crosspoints in multiple environmental backgrounds. The experimental results show that the P_{macro} and R_{macro} of the rebar crosspoints model are 95.7% and 97.0%. For the crosspoints localization method, compared with the traditional Hough detection method, its localization accuracy reaches 90.45%, the average relative error is 6.15%, and the localization speed reaches 6.94 ms, which ensures the timeliness of the algorithm while satisfying the localization accuracy. For recognition and localization, vision-based methods can show better performance because rich visual information can be utilized for feature matching.

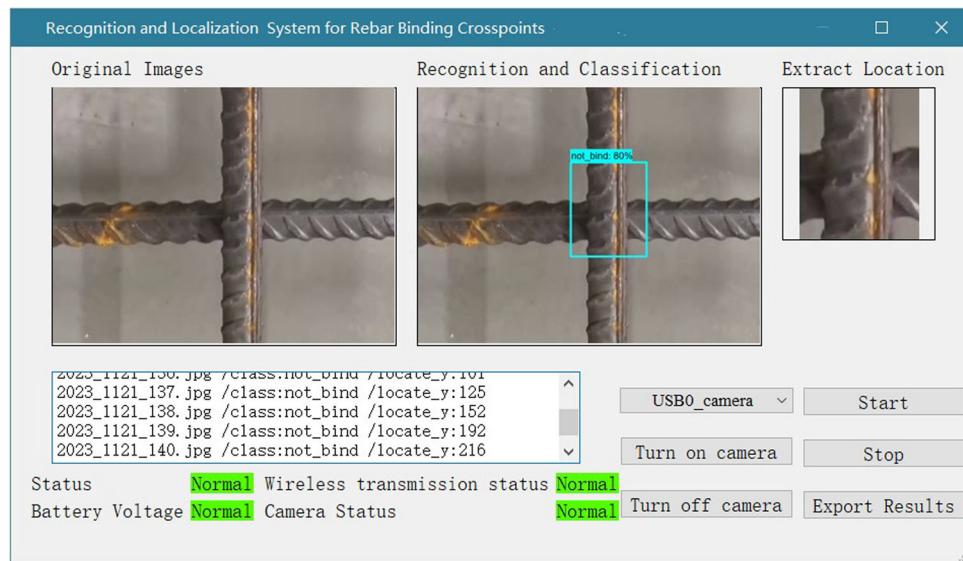


FIGURE 29 Recognition situation on the field test.

- (3) For the existence of complex environments such as obstacles at the rebar engineering work site, this paper proposes a priority-based BFP planning method combined with the A* algorithm. On the one hand, the priority direction restriction is added, on the other hand, the penalty function is introduced into the evaluation function to optimize the evaluation function. Through these two kinds of optimization, the mobile robot can achieve complete coverage of the entire environment. Simulation results show that the proposed path-planning algorithm can achieve complete traversal coverage of the crosspoint area and effectively reduce the path repetition rate and the number of turns.
- (4) The prototype of the rebar binding robot is constructed, and experiments on the motion of the prototype, recognition, and localization of rebar binding crosspoints, and complete coverage path-planning experiments are carried out, respectively, to verify that the robot can complete the upper and lower bar walking and binding actuators, the recognition and localization function, and the complete coverage path-planning function, which proves that the proposed rebar binding robot has a certain degree of feasibility.

Nonetheless, the proposed robot system still has limitations and further research will be conducted to resolve these limitations, as detailed below:

- (1) The overall structure of the rebar binding robot can be designed to be more lightweight and enhance its flexibility in the process of movement to adapt to a variety of different application scenarios.
- (2) This study conducted detection of rebar binding crosspoints, which can be well applied in rebar binding robots. However, further solutions are needed to address the issue of low discrimination between rebar surface reflection and longitudinal

and transverse texture, which affects the accuracy of recognition. Therefore, it is necessary to further improve the recognition framework and enhance recognition accuracy based on rebar features.

- (3) In terms of possible development directions and improvement space, the method proposed in this article can be further optimized and improved, which can be applied to the formation of steel rebar skeletons for upright large wall panels and T-shaped box beams, as well as the recognition and binding of rebar crosspoints.

ACKNOWLEDGMENTS

The authors would like to acknowledge the financial support provided by Natural Science Basic Research Project of Shaanxi Province, under Project No. 2021JM-360.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

- An, M. & Kang, D.S. (2022) The distance measurement based on corner detection for rebar spacing in engineering images. *The Journal of Supercomputing*, 78(10), 12380–12393.
- An, V., Qu, Z., Crosby, F., Roberts, R. & An, V. (2020) A triangulation-based coverage path planning. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 50, 2157–2169.
- Bisht, R.S., Pathak, P.M. & Panigrahi, S.K. (2022) Design and development of a glass façade cleaning robot. *Mechanism and Machine Theory*, 168, 104585. Available from: <https://doi.org/10.1016/j.mechmachtheory.2021.104585>
- Bosse, M., Nourani-Vatani, N. & Roberts, J. (2007) Coverage algorithms for an under-actuated car-like vehicle in an uncertain environment. In: *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*. Rome, Italy: IEEE, pp. 698–703.

- Chen, Q., Zhang, Q., Niu, X. & Wang, Y. (2019) Positioning accuracy of a pipeline surveying system based on MEMS IMU and odometer: case study. *IEEE Access*, 7, 104453–104461.
- Choset, H. (2000) Coverage of known spaces: the boustrophedon cellular decomposition. *Autonomous Robots*, 9(3), 247–253.
- Fang, W., Ding, L., Zhong, B., Love, P.E.D. & Luo, H. (2018) Automated detection of workers and heavy equipment on construction sites: a convolutional neural network approach. *Advanced Engineering Informatics*, 37, 139–149.
- Galceran, E. & Carreras, M. (2013) A survey on coverage path planning for robotics. *Robotics and Autonomous Systems*, 61(12), 1258–1276.
- Gharbia, M., Chang-Richards, A., Lu, Y., Zhong, R.Y. & Li, H. (2020) Robotic technologies for on-site building construction: a systematic review. *Journal of Building Engineering*, 32, 101584. Available from: <https://doi.org/10.1016/j.jobe.2020.101584>
- Girshick, R., Donahue, J., Darrell, T. & Malik, J. (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. Columbus, USA: IEEE, pp. 580–587.
- Gurjali, A.K., Agarwal, H. & Parsadiya, D.K. (2016) Time-efficient A* algorithm for robot path planning. *Procedia Technology*, 23, 144–149.
- Hassan, M., Mustafic, D. & Liu, D. (2020) Dec-PPCPP: a decentralized predator-prey-based approach to adaptive coverage path planning amid moving obstacles. In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Las Vegas, USA: IEEE, pp. 11732–11739.
- Howard, A., Sandler, M., Chen, B., Wang, W., Chen, L.C., Tan, M. et al. (2019) Searching for MobileNetV3. In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. Seoul, Korea: IEEE, pp. 1314–1324.
- Joshi, D., Singh, T.P. & Sharma, G. (2022) Automatic surface crack detection using segmentation-based deep-learning approach. *Engineering Fracture Mechanics*, 268, 108467. Available from: <https://doi.org/10.1016/j.engfracmech.2022.108467>
- Kim, M.K., Thedja, J.P.P., Chi, H.L. & Lee, D.E. (2021) Automated rebar diameter classification using point cloud data based machine learning. *Automation in Construction*, 122, 103476. Available from: <https://doi.org/10.1016/j.autcon.2020.103476>
- Lee, J.H. & Park, S.O. (2019) Machine learning-based automatic reinforcing bar image analysis system in the internet of things. *Multimedia Tools and Applications*, 78(3), 3171–3180.
- Li, C., Huang, X., Ding, J., Song, K. & Lu, S. (2022) Global path planning based on a bidirectional alternating search A* algorithm for mobile robots. *Computers & Industrial Engineering*, 168, 108123. Available from: <https://doi.org/10.1016/j.cie.2022.108123>
- Li, D., Wang, L., Cai, J., Wang, A., Tan, T. & Gui, J. (2023) Research on path planning of mobile robot based on improved genetic algorithm. *International Journal of Modeling, Simulation, and Scientific Computing*, 14, 2341030. Available from: <https://doi.org/10.1142/s1793962323410301>
- Li, J., Su, Z., Geng, J. & Yin, Y. (2018) Real-time detection of steel strip surface defects based on improved YOLO detection network. *IFAC-PapersOnLine*, 51(21), 76–81.
- Li, Y., Lu, Y. & Chen, J. (2021) A deep learning approach for real-time rebar counting on the construction site based on YOLOv3 detector. *Automation in Construction*, 124, 103602. Available from: <https://doi.org/10.1016/j.autcon.2021.103602>
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y. et al. (2016) SSD: single shot multiBox detector. In: *Computer Vision—ECCV 2016*. Amsterdam, Holland: Springer, pp. 21–37.
- Ma, L., Zhu, J., Zhu, L., Du, S. & Cui, J. (2015) Merging grid maps of different resolutions by scaling registration. *Robotica*, 34, 2516–2531.
- Melenbrink, N., Werfel, J. & Menges, A. (2020) On-site autonomous construction robots: towards unsupervised building. *Automation in Construction*, 119, 103312. Available from: <https://doi.org/10.1016/j.autcon.2020.103312>
- Muthugala, M.A.V.J., Samarakoon, S.M.B.P. & Elara, M.R. (2022) Toward energy-efficient online complete coverage path planning of a ship hull maintenance robot based on Gladius bio-inspired neural network. *Expert Systems with Applications*, 187, 115940. Available from: <https://doi.org/10.1016/j.eswa.2021.115940>
- Pan, M., Linner, T., Pan, W., Cheng, H. & Bock, T. (2020) Influencing factors of the future utilisation of construction robots for buildings: a Hong Kong perspective. *Journal of Building Engineering*, 30, 101220. Available from: <https://doi.org/10.1016/j.jobe.2020.101220>
- Prabakaran, V., Le, A.V., Kyaw, P.T., Kandasamy, P., Paing, A. & Mohan, R.E. (2023) sTetro-D: a deep learning based autonomous descending-stair cleaning robot. *Engineering Applications of Artificial Intelligence*, 120, 105844. Available from: <https://www.sciencedirect.com/science/article/pii/S0952197623000283>
- Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. (2016) You only look once: unified, real-time object detection. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, USA: IEEE, pp. 779–788.
- Shi, Q., Wang, Z., Ke, X., Wang, Z., Gao, Q., Fan, Y. et al. (2023) Multiobjective trajectory optimization of the wall-building robot based on RBF-NSGA-II in an uncertain viscoelastic contact environment. *Journal of Field Robotics*, 40, 1964–1995. Available from: <https://doi.org/10.1002/rob.22235>
- Simonyan, K. & Zisserman, A. (2014) Very deep convolutional networks for large-scale image recognition. In: *Proceedings of the 2015 International Conference on Learning Representations*. San Diego, USA: ICLR, pp. 1–14.
- Spencer, B.F., Hoskere, V. & Narazaki, Y. (2019) Advances in computer vision-based civil infrastructure inspection and monitoring. *Engineering*, 5(2), 199–222.
- Spong, M., Hutchinson, S. & Vidyasagar, M. (2006) *Robot modeling and control*. John Wiley and Sons.
- Ueno, T.M.J., Yoshida, T. & Suzuki, S. (1986) CAD and robotics in architecture and construction. In: *Proceedings of the Joint International Conference*. Berlin, Germany: Springer, pp. 259–268.
- Wang, H., Polden, J., Jirgens, J., Yu, Z. & Pan, Z. (2019) Automatic rebar counting using image processing and machine learning. In: *2019 IEEE 9th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*. Suzhou China: IEEE, pp. 900–904.
- Wang, Y., Xie, L., Wang, H., Zeng, W., Ding, Y., Hu, T. et al. (2022) Intelligent spraying robot for building walls with mobility and perception. *Automation in Construction*, 139, 104270. Available from: <https://doi.org/10.1016/j.autcon.2022.104270>
- Watfa, M., Bykovski, A. & Jafar, K. (2022) Testing automation adoption influencers in construction using light deep learning. *Automation in Construction*, 141, 104448. Available from: <https://doi.org/10.1016/j.autcon.2022.104448>
- Wei, P., Yu, X., Di, Z., Dai, X., Wang, B. & Zeng, Y. (2022) Design of robot automatic navigation under computer intelligent algorithm and machine vision. *Journal of Industrial Information Integration*, 28, 100366. Available from: <https://doi.org/10.1016/j.jiii.2022.100366>
- Wu, Y., Zhou, X. & Zhang, Y. (2015) Steel bars counting and splitting method based on machine vision. In: *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*. Shenyang, China: IEEE, pp. 420–425.
- Yang, L., Liu, Y., Peng, J. & Liang, Z. (2020) A novel system for off-line 3D seam extraction and path planning based on point cloud segmentation for arc welding robot. *Robotics and Computer-Integrated Manufacturing*, 64(3), 101929. Available from: <https://doi.org/10.1016/j.rcim.2019.101929>
- Ying, X., Wei, X., Yuan, P.X., Han, Q.D. & Chen, C.H. (2010) Research on an automatic counting method for steel bars' image. In: *2010 International Conference on Electrical and Control Engineering*. Wuhan, China: IEEE, pp. 1644–1647.

- Zhang, Q., Hu, X., Yue, Y., Gu, Y. & Sun, Y. (2022) Multi-object detection at night for traffic investigations based on improved SSD framework. *Helijon*, 8(11), e11570.
- Zhao, J., Xia, X., Wang, H., & Kong, S. (2016) Design of real-time steel bars recognition system based on machine vision. In: 2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC). Hangzhou, China: IEEE, pp. 505–509.

How to cite this article: Cheng, B. & Deng, L. (2024) Vision detection and path planning of mobile robots for rebar binding. *Journal of Field Robotics*, 41, 1864–1886.

<https://doi.org/10.1002/rob.22356>