Article

Object Segmentation by Spraying Robot Based on Multi-Layer Perceptron

Mingxiang Zhu [1,2], Guangming Zhang [1,*], Lingxiu Zhang [1], Weisong Han [3], Zhihan Shi [1] and Xiaodong Lv [1]

[1] College of Electrical Engineering and Control Science, Nanjing Tech University, Nanjing 211899, China
[2] Taizhou College, Nanjing Normal University, Taizhou 225300, China
[3] College of Transportation Engineering, Nanjing Tech University, Nanjing 211899, China
[*] Correspondence: zgm@njtech.edu.cn

Abstract: The vision system provides an important way for construction robots to obtain the type and spatial location information of the object. The characteristics of the construction environment, construction object, and robot structure are jointly examined in this paper to propose an approach of object segmentation by spraying the robot based on multi-layer perceptron. Firstly, the hand-eye system experimental platform is built through establishing the mathematical model of the system and calibrating the parameters of the model. Secondly, effort is made to carry out research on image preprocessing algorithms and related experiments, and compare the effects of different binocular stereo-matching algorithms in the actual engineering environment. Finally, research and an experiment are conducted to identify the applicability and effect of the depth image object segmentation algorithm based on multi-layer perceptron. The experimental results prove that the application of multi-layer perceptron to object segmentation by spraying robots can meet the requirement on solution accuracy and is suitable for the object segmentation of complex projects in real life. This approach not only overcomes the shortcomings of the existing recognition methods that are poor in accuracy and difficult to be used widely, but also provides basic data for the subsequent three-dimensional reconstruction, thus making a significant contribution to the research of image processing by spraying robots.

Keywords: hand-eye calibration; stereo matching; object segmentation; multi-layer perceptron

check for updates

Citation: Zhu, M.; Zhang, G.; Zhang, L.; Han, W.; Shi, Z.; Lv, X. Object Segmentation by Spraying Robot Based on Multi-Layer Perceptron. *Energies* **2023**, *16*, 232. https://doi.org/10.3390/en16010232

Academic Editor: Luigi Fortuna

Received: 25 November 2022
Revised: 20 December 2022
Accepted: 21 December 2022
Published: 25 December 2022

1. Introduction

The construction of the load-bearing structure of steel structure is one of the main schemes in the field of engineering construction either at present or for a long time in the future [1]. However, the materials of steel structures feature poor fire resistance. It is pressing to explore the use of robots to replace the original manual construction work in the fire protection spraying of steel structures, which has strict requirements on image processing by spraying robot and is of great significance to the automation, intelligence, less manned operation and even unmanned operation of the whole construction process.

To facilitate the effective image object segmentation by spraying robots, scholars have proposed a variety of models, such as grey models [2,3], deep learning neural networks [4,5], machine learning [6,7], grey time-varying parameters model [8], grey-neural network model [9,10], etc. Nevertheless, these models require time series monitoring data to be stable, face difficulties in capturing the relationship between nonlinear data, need a large-size sample of data, and have defects such as slow convergence speed or hardship to accurately determine the large number of parameters. In practice, the image data available to spraying robots is usually characterized by a small sample size and nonlinearity. It should be noted that depth images belong to gray images. In the case that the background image is complex, it is impossible to reliably finish the task of object segmentation through the traditional gray image segmentation method based on threshold or dynamic threshold.

In recent years, intelligent optimization and machine learning algorithm, especially their combined models such as particle swarm optimization combined with support vector machine [11], integrated intelligent algorithm [12,13], genetic algorithm combined with support vector machine [14], neural network algorithm combined with machine learning model [15], have become the research focus of machine vision as a result of their excellent performance. Nonetheless, in the combined models above, intelligent optimization algorithm itself may be prone to issues such as low local optimal value and convergence accuracy. The combination of optimization algorithm and machine learning algorithm cannot reasonably determine the hyper-parameters of machine algorithm, thus leading to low prediction accuracy. Image object segmentation is now widely used in object detection [16]. Vision inspection by spraying robot is regarded as an image segmentation problem. For example, structural feature convolution neural network (SCNN) [17] uses an image segmentation model to segment lane lines, and adopts message passing and additional scene annotations to capture global context information to improve accuracy. It has stronger representation capability than traditional image processing methods, but intensive pixel-level communication requires a lot of computing resources, resulting in low processing efficiency of the algorithm. In Ref. [18], one pixel in each line of lane images is detected as lane line in the processing of these images. Compared with image segmentation algorithm, this method reduces the work of calculation and improves the reasoning speed, but it has low universality and cannot be used for lane line detection in multiple environments.

With the development of computer technology, computer hardware and software have been significantly improved in performance, creating an environment for the introduction and application of deep learning. As a method of machine learning, Artificial Neural Network (ANN) has been widely used in industrial fields such as intelligent buildings. Multi-Layer Perceptron (MLP) is a classical model of artificial neural network, which is developed from perceptron. Its main feature is that there are multiple neuron layers. The basic structure of MLP includes input layer, hidden layer and output layer. The number of hidden layers can be varied. The input layer to the hidden layer can be regarded as a fully connected layer, and the hidden layer to the output layer can be regarded as a classifier. Many recent studies on MLP [19] have shown that MLP can better extract global semantic information from images. Specifically, Cycle-MLP in Ref. [20] has achieved good results in the downstream tasks of computer vision such as image segmentation. In Ref. [21], the decoupling of training and reasoning is achieved through the technology of structural re-parameterization, where obvious improvement in accuracy is made without sacrificing the reasoning speed. For instance, the MLP model in Ref. [22] builds an internal group convolution layer while training to obtain local information, and combines with the technology of re-parameterization. This method has achieved better results in pattern recognition. MLP implementation form is to map multiple input data sets to a single output data set. The main advantage is that it solves the nonlinear problem that single-layer perceptron cannot solve on the basis of linear regression, which is favored by scholars in different fields. Zhang [23] used MLP model to predict the traffic flow of a certain road in Chongqing, and achieved good accuracy. Lyu et al. [24] focused on the application of multi-layer perceptron in the prediction of the growth of various microorganisms in aviation catering, and established a growth model for the microbial community. Ding Xuesong et al. [25] predicted protein denaturation temperature based on multi-layer perceptron model, which solved the shortcomings of traditional experimental methods. This data-driven method predicts by mining the implicit relationship between data and target state, which has low dependence on physical mechanism. It can directly use the original sampling data as input and extract important features for predicting response, which is more versatile.

Therefore, the region segmentation of the whole object is considered in this paper as the task of classifying each pixel point in the object. Multi-layer perceptron is used to classify each pixel point in the corrected, colored left-eye images, so as to complete the task of the object segmentation of steel structure columns. The segmentation results are employed to extract information from each depth image, and obtain the depth images containing only steel structure columns. This approach not only overcomes the shortcomings of the existing recognition methods that are poor in accuracy and difficult to be used widely, but also provides basic data for the subsequent three-dimensional reconstruction, thus making significant contribution to the research of image processing by spraying robot.

2. Structure and Modeling of the System

2.1. Construction of the System Experimental Platform

The main hardware of the system experimental platform is shown in Figure 1. In order to reduce the movement of the robot body in the spraying process, a six-axis mechanical arm is used to meet the flexibility requirement for the planning of spraying trajectory. The maximum arm span of the mechanical arm is 1.710 m, and the end pose accessibility is good within 1.5 m; the rated load of the wrist is 20 kg, and the repeated positioning accuracy at the end space is ±0.1 mm, meeting the load requirements of end devices such as the spray gun and the binocular camera. A monocular 1080P high-resolution CMOS sensor is used as the binocular camera. The lens field angle and the focal length are estimated according to the end reach range of the mechanical arm in the horizontal and vertical directions. The standard for fine adjustment of the binocular focal length is determined to be the clear imaging of the object at 1.4 m away from the camera.

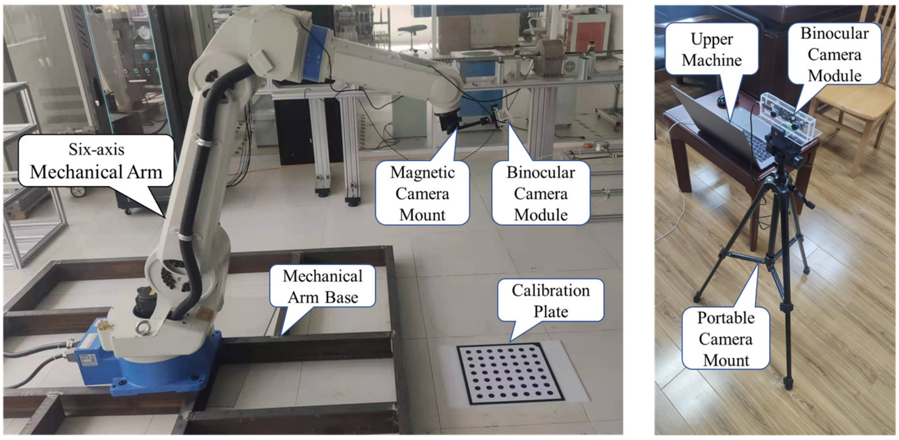


Figure 1. Hardware of hand-eye system experimental platform.

2.2. Mathematical Model of the Robot's Hand-Eye System

2.2.1. Theory of Coordinate System Transformation

Without losing generality, $O_\mathrm{A} - X_\mathrm{A}Y_\mathrm{A}Z_\mathrm{A}$ and $O_\mathrm{B} - X_\mathrm{B}Y_\mathrm{B}Z_\mathrm{B}$ are respectively set to be two Cartesian space rectangular coordinate systems. The coordinates of the point P in space in the coordinate system and {B} are $P_\mathrm{A}(x_\mathrm{A}, y_\mathrm{A}, z_\mathrm{A})$ and $P_\mathrm{B}(x_\mathrm{B}, y_\mathrm{B}, z_\mathrm{B})$ respectively. The coordinate system {B} can be obtained from the coordinate system {A} through the following transformation process: rotating in the positive direction around the axis Z_A by the angle γ, then rotating in the positive direction around the axis Y_A by the angle β and then rotating in the positive direction around the axis X_A by the angle α, finally with the origin O_B translating upwards in the positive direction at X_A, Y_A and Z_A respectively by t_X, t_Y and t_Z. According to the theory of spatial coordinate system transformation [25], it can be obtained that:

$$\begin{bmatrix} x_\mathrm{A} \\ y_\mathrm{A} \\ z_\mathrm{A} \end{bmatrix} = R_X(\alpha)R_Y(\beta)R_Z(\gamma)\begin{bmatrix} x_\mathrm{B} \\ y_\mathrm{B} \\ z_\mathrm{B} \end{bmatrix} + \begin{bmatrix} t_X \\ t_Y \\ t_Z \end{bmatrix} \triangleq {}^\mathrm{A}_\mathrm{B}R \begin{bmatrix} x_\mathrm{B} \\ y_\mathrm{B} \\ z_\mathrm{B} \end{bmatrix} + {}^\mathrm{A}_\mathrm{B}T \tag{1}$$

where, the matrix ${}^\mathrm{A}_\mathrm{B}R$ is the rotation matrix from the coordinate system {B} to the coordinate system {A}, and the matrix ${}^\mathrm{A}_\mathrm{B}T$ is the translation matrix from the coordinate system {B} to the coordinate system {A}.$R_X(\alpha)$, $R_Y(\beta)$, $R_Z(\gamma)$ and ${}^{\mathrm{A}}_{\mathrm{B}}R$ are all invertible matrices and are respectively expressed as:

$$R_X(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \tag{2}$$

$$R_Y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \tag{3}$$

$$R_Z(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$${}^{\mathrm{B}}_{\mathrm{A}}R = R_X(\alpha)R_Y(\beta)R_Z(\gamma) \triangleq \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{5}$$

The rotation angle α, β and γ can be obtained by the element r_{ij} in the matrix ${}^{\mathrm{B}}_{\mathrm{A}}R$ through calculation according to Equation (6):

$$\begin{cases} \alpha = \mathrm{atan2}(r_{32}, r_{33}) \\ \beta = \mathrm{atan2}\left(-r_{31}, \sqrt{{r_{31}}^2 + {r_{33}}^2}\right) \\ \gamma = \mathrm{atan2}(r_{21}, r_{11}) \end{cases} \tag{6}$$

To facilitate continuous coordinate transformation, Equation (1) is extended to homogeneous forms, including:

$$\begin{bmatrix} x_{\mathrm{A}} \\ y_{\mathrm{A}} \\ z_{\mathrm{A}} \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{\mathrm{A}}_{\mathrm{B}}R & {}^{\mathrm{A}}_{\mathrm{B}}T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{\mathrm{B}} \\ y_{\mathrm{B}} \\ z_{\mathrm{B}} \\ 1 \end{bmatrix} \triangleq {}^{\mathrm{A}}_{\mathrm{B}}M \begin{bmatrix} x_{\mathrm{B}} \\ y_{\mathrm{B}} \\ z_{\mathrm{B}} \\ 1 \end{bmatrix} \tag{7}$$

where, the matrix ${}^{\mathrm{A}}_{\mathrm{B}}M$ is the transformation matrix from the coordinate system {B} to coordinate system {A}, and there is:

$$\left({}^{\mathrm{A}}_{\mathrm{B}}M\right)^{-1} = \begin{bmatrix} {}^{\mathrm{A}}_{\mathrm{B}}R & {}^{\mathrm{A}}_{\mathrm{B}}T \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} {}^{\mathrm{A}}_{\mathrm{B}}R^{-1} & -{}^{\mathrm{A}}_{\mathrm{B}}T \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^{\mathrm{B}}_{\mathrm{A}}R & {}^{\mathrm{B}}_{\mathrm{A}}T \\ 0 & 1 \end{bmatrix} = {}^{\mathrm{B}}_{\mathrm{A}}M \tag{8}$$

2.2.2. Eye-in-Hand Model

In the hand-eye system, the binocular camera is installed at the end of the sixth axis of the mechanical arm through a fixing bracket. The Cartesian space rectangular coordinate system of the system is established in accordance with the way shown in Figure 2. Specifically, $O_{\mathrm{base}} - X_{\mathrm{base}}Y_{\mathrm{base}}Z_{\mathrm{base}}$ is the coordinate system of the base of the mechanical arm and also the world coordinate system, of which the origin is located in the center of the base of the mechanical arm; $O_{\mathrm{tool}} - X_{\mathrm{tool}}Y_{\mathrm{tool}}Z_{\mathrm{tool}}$ is the tool coordinate system, of which the origin is located in the center of the flange plate of the sixth axis of the mechanical arm; $O_{\mathrm{cam}} - X_{\mathrm{cam}}Y_{\mathrm{cam}}Z_{\mathrm{cam}}$ is the ideal binocular camera coordinate system, of which the origin is located at the optical center of the left-eye camera; $O_{\mathrm{cal}} - X_{\mathrm{cal}}Y_{\mathrm{cal}}Z_{\mathrm{cal}}$ is the calibration plate coordinate system, of which the origin is located in the center of the calibration plate.

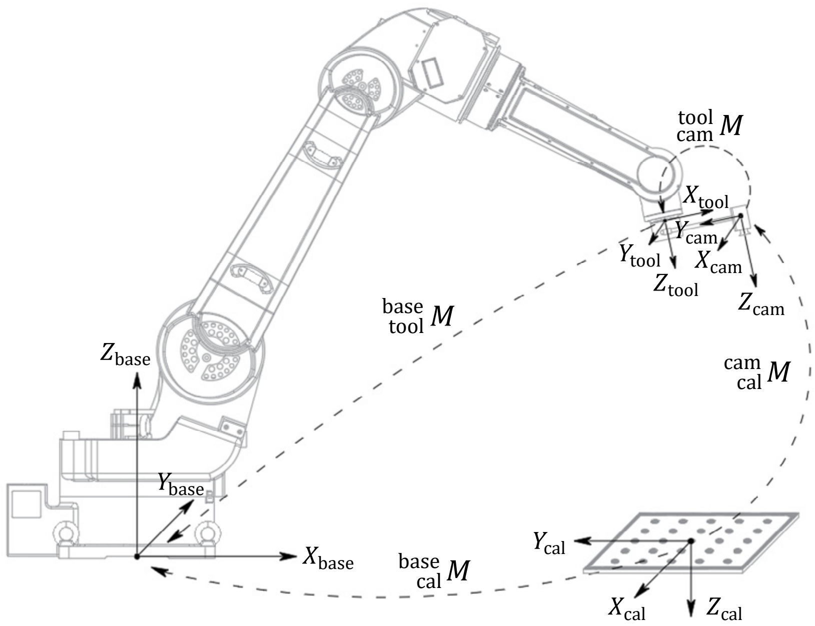


Figure 2. Eye in hand system model.

2.3. Mathematical Model of the Binocular Camera

The coordinate system is constructed according to the way shown in Figure 3, in order to analyze and establish the imaging mathematical model of the binocular camera used in this study.

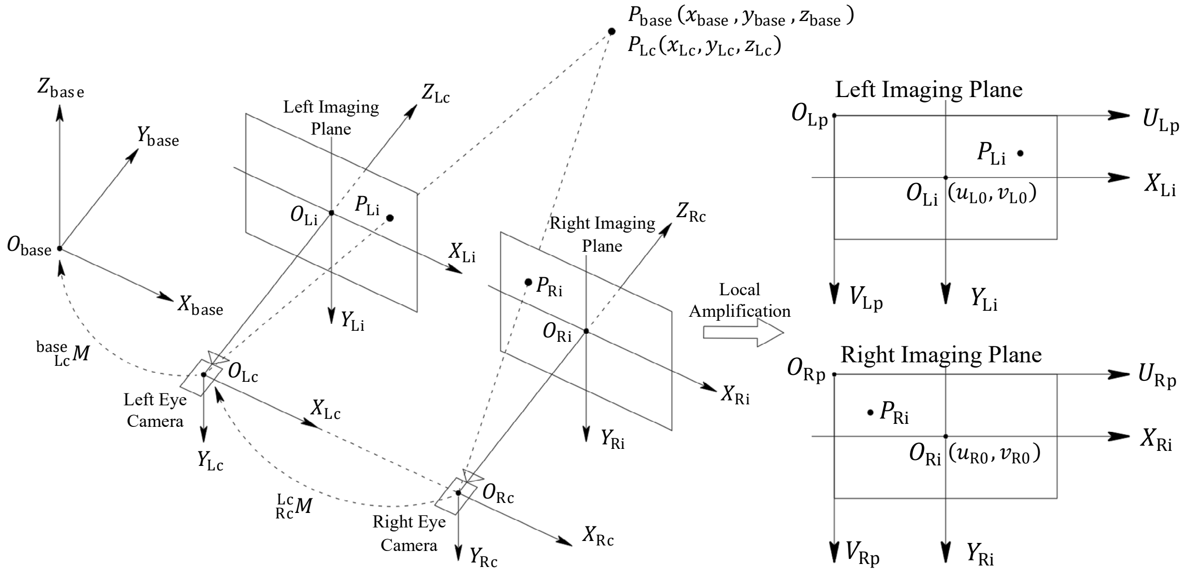


Figure 3. Binocular imaging system model.

Where, $O_{\text{Lc}} - X_{\text{Lc}}Y_{\text{Lc}}Z_{\text{Lc}}$ is the left-eye camera coordinate system, with O_{Lc} being located at the optical center of the left-eye camera and Z_{Lc} being the optical axis of the left-eye camera; $O_{\text{Rc}} - X_{\text{Rc}}Y_{\text{Rc}}Z_{\text{Rc}}$ is the right-eye camera coordinate system, with O_{Rc} being located at the optical center of the right-eye camera and Z_{Rc} being the optical axis of the right-eye camera; the distance between O_{Rc} and O_{Lc} is b, that is, the baseline length; the matrix ${}^{\text{Lc}}_{\text{Rc}}M$ is the transformation matrix from the coordinate system to the coordinate system {Lc}, and the matrix ${}^{\text{base}}_{\text{Lc}}M$ is the transformation matrix from the coordinate system {Lc} to the coordinate system {base}; $O_{\text{Lp}} - U_{\text{Lp}}V_{\text{Lp}}$ and $O_{\text{Rp}} - U_{\text{Rp}}V_{\text{Rp}}$ are the left- and right-eye pixel coordinate system respectively; $O_{\mathrm{Li}} - X_{\mathrm{Li}}Y_{\mathrm{Li}}$ and $O_{\mathrm{Ri}} - X_{\mathrm{Ri}}Y_{\mathrm{Ri}}$ are the left- and right-eye image coordinate system respectively; The coordinate of O_{Li} in the coordinate system {Lp} is $(u_{\mathrm{L0}}, v_{\mathrm{L0}})$ and its distance from O_{Lc} is the left-eye lens focal length f_{Lc}; The coordinate of O_{Ri} in the coordinate system is $(u_{\mathrm{R0}}, v_{\mathrm{R0}})$ and its distance from O_{Rc} is the right-eye lens focal length f_{Rc}; the images of the Point P_{base} in space in the left and right imaging planes are respectively P_{Li} and P_{Ri}.

3. Calibration of System Parameters

3.1. Calibration of Binocular Camera Parameters

The binocular calibration task is to determine the accurate values of relevant parameters in the camera model, covering the camera focal length f_c, the pixel sizes including S_U and S_V, the pixel coordinate of the image center point (u_0, v_0), the imaging distortion model parameters including k_1, k_2, k_3, p_1 and p_2, and the transformation matrix ${}^{\mathrm{Lc}}_{\mathrm{Rc}}M$ from the coordinate system to the coordinate system {Lc} including the rotation matrix ${}^{\mathrm{Lc}}_{\mathrm{Rc}}R$ and the translation matrix ${}^{\mathrm{Lc}}_{\mathrm{Rc}}T$.

The binocular calibration experiment in this paper adopts high-precision 7 × 7 center displaying calibration plate. The calibration pattern region is a square with a side length of 400 mm, in which the circular pattern has a diameter of 25 mm and a spacing of 50 mm. There is a pose identification mark of the calibration plate coordinate system {cal} in the upper left corner. The binocular camera calibration experimental equipment and layout are shown in Figure 4.

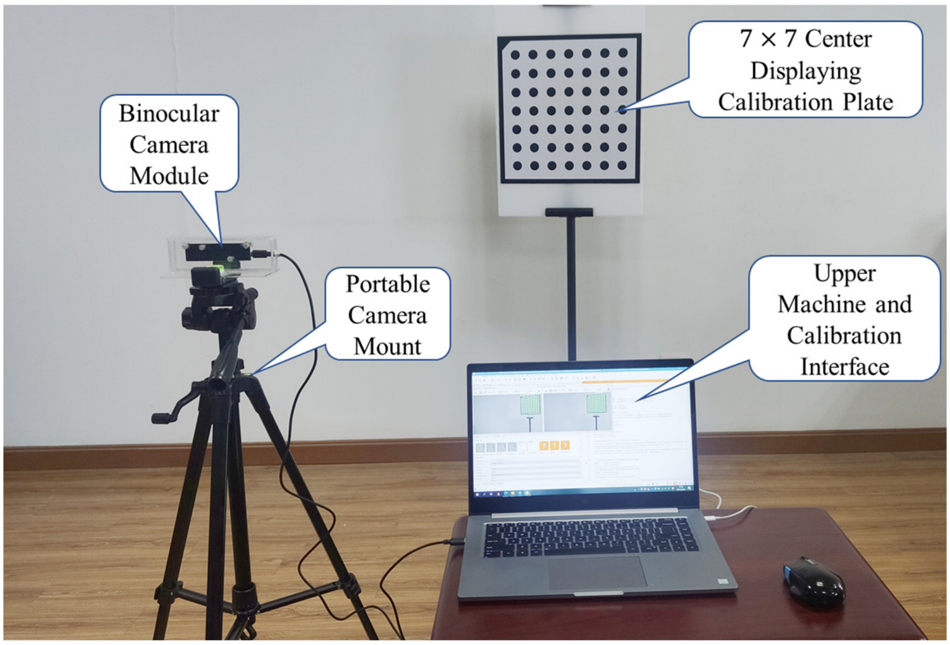

Binocular Camera Module
Portable Camera Mount
7 × 7 Center Displaying Calibration Plate
Upper Machine and Calibration Interface

Figure 4. Binocular camera calibration experimental equipment and layout.

The binocular camera calibration results obtained by the calibration plate through image acquisition and landmark extraction are shown in Tables 1–3.

Table 1. Calibration results of internal parameters of binocular camera.

Parameter	f_c/mm	S_U/µm	S_V/µm	u_0	v_0	Image Width	Image High
Left eye	4.767	3.394	3.380	967.551	529.788	1920	1080
Right eye	4.776	3.394	3.380	951.657	540.111	1920	1080

Table 2. Distortion parameter calibration results.

Parameter	$k_1/(\mathrm{m}^2)^{-1}$	$k_2/(\mathrm{m}^4)^{-1}$	$k_3/(\mathrm{m}^6)^{-1}$	$p_1/(\mathrm{m}^2)^{-1}$	$p_2/(\mathrm{m}^2)^{-1}$
Left eye	−519.880	-1.594×10^7	4.256×10^{12}	0.124104	−0.236724
Right eye	−482.356	-3.230×10^7	5.406×10^{12}	0.180746	−0.204752

Table 3. Calibration results of binocular position relationship parameters.

Parameter	Rotation Matrix ${}^{\mathrm{Lc}}_{\mathrm{Rc}}R$			Translation Vector ${}^{\mathrm{Lc}}_{\mathrm{Rc}}T$		
	α_{Lc}/deg	β_{Lc}/deg	γ_{Lc}/deg	X_{Lc}/mm	Y_{Lc}/mm	Z_{Lc}/mm
parameter value	0.0238	0.2601	359.9840	60.1305	0.0956	1.1106

The average double projection error of binocular calibration is 0.1104 pixels, and the overall calibration error is small. The values of k_1, k_2 and k_3 are relatively large, because r_{Li} and r_{Ri} in the distortion model are taken in meters, while the absolute values of their actual physical quantities are quite small. At the same time, the values of p_1 and p_2 are smaller, indicating that there is less tangential distortion in the imaging process and that the imaging distortion is dominated by radial distortion.

3.2. Calibration of Hand-Eye Positional Relationship Parameters

The goal of hand-eye calibration is to obtain the positional relationship between the ideal binocular camera coordinate system {cam} and the tool coordinate system {tool} (or the base coordinate system {base} of the mechanical arm). There is no essential difference between the two hand-eye system calibration methods. With the assistance from HALCON algorithm library, the basic process of hand-eye calibration in this study is:

(1) Adjusting the end pose of the mechanical arm using a teach pendant or remote-control mode, and reading the pose of the end tool coordinate system {tool} under the coordinate system {base} from the information system of the mechanical arm;

(2) Using the left-eye camera to capture the images containing the complete calibration plate and adopting the operator map_image() to correct the correction mapping images (file of maps) obtained from calibration by the binocular camera;

(3) Using the operator find_calib_object() to find the object in the corrected calibration plate images and adopting the operator get_calib_data_observ_pose() to extract the pose of the calibration plate coordinate system {cal} under the camera coordinate system {cam};

(4) Repeating the process above to obtain 15 sets of pose relationships, using calibrate_hand_eye() to perform calibration calculation according to the hand-eye system model and using get_calib_data() to read the results.

4. Binocular Stereo Matching

4.1. Image Preprocessing

Because the actual binocular camera objectively has some problems such as imaging distortion, sensor noise and non-ideal parameters of the binocular model, it is necessary to correct, filter and gray the original-colored binocular images in preprocessing before the stereo matching of binocular images, so that they can meet the basic requirement for binocular stereo matching. In addition to some binocular images collected in the laboratory environment for verifying the algorithm's accuracy, the binocular images used in the actual effect verification of the algorithm are taken from the construction site. The measured object is large-scale square steel structure columns. During the acquisition of images, the binocular camera keeps the optical axis horizontal and is fixed at the position 1.565 m in front of the rectangular edge of steel structure columns (the horizontal distance between the optical center of the left-eye camera and the rectangular edge is measured by a laser rangefinder). The acquired binocular images are shown in Figure 5.

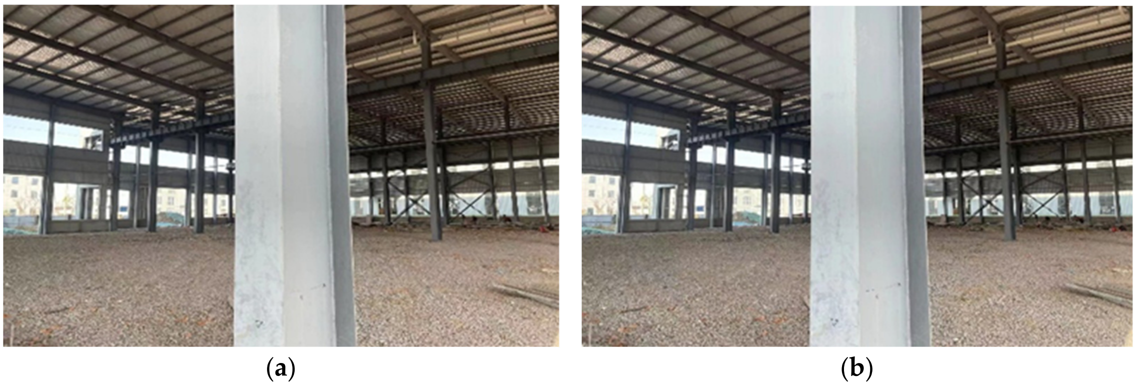


Figure 5. Original binocular images of steel structure column: (**a**) is left-eye image; (**b**) is right-eye image.

4.1.1. Smoothing Filtering

In the process of image acquisition and signal transmission by the camera, its electronic components would inevitably be subject to external electromagnetic interference, so there is often noise in the image obtained. In order to reduce the effect of imaging noise on stereo matching, it is necessary to eliminate noise through image smoothing filtering. In this study, Gaussian filtering is used as the image smoothing filtering method. For the Gaussian template H with a side length of $2k + 1$, if the center point coordinate of the template is set to (0,0), the formula for calculating each element in the template is:

$$H_{(i,j)} = \frac{1}{2\pi\sigma^2}\exp\left[-\frac{(i-k-1)^2+(j-k-1)^2}{2\sigma^2}\right], \quad i,j \in [-k,k] \tag{9}$$

where, k and σ are the parameters to be designed in the template. In practical application, the template size k and the parameters σ need to be selected and adjusted according to the actual image filtering effect.

4.1.2. Image Graying

Image graying is the process of transforming RGB three-channel (or multi-channel) images into single-channel images, to realize the compression of image information and increase the contrast ratio.

The weighted average method is a commonly used approach of graying. This method assigns different weights to different channels according to the different sensitivity of human eyes to different colors, so that the gray images obtained are closer to the subjective feelings of human beings. In this study, the weighted average method is used as the approach of image graying. The formula for calculating each pixel point in the gray images is as follows:

$$I(u,v) = 0.299 * R(u,v) + 0.587 * G(u,v) + 0.114 * B(u,v) \tag{10}$$

where, $R(u,v)$, $G(u,v)$ and $B(u,v)$ are respectively the pixel values of the image Red channel, Green channel and Blue channel at the pixel point (u, v).

4.1.3. Histogram Equalization

Due to the complex lighting conditions in the construction environment, especially the uncontrollable and uneven illumination of natural light on the object, the difference in the angle of view of the binocular camera in this environment would cause a certain brightness difference in the collected binocular images, which destroys the basic assumption that stereo matching algorithm depends on. In order to reduce the impact of the brightness difference in binocular images on subsequent stereo matching, this paper uses the method of histogram equalization to equalize the histograms of binocular gray images and thus to balance the dynamic brightness range of the binocular images.

The basic idea of histogram equalization is to transform the histogram of the original gray image into a form of uniform distribution in the whole gray range. If the total number of pixel points in the image is n and the gray value of pixel points is $k \in [0, 255]$, the corresponding histogram equalization process is:

(1) Traversing the whole image to count the number n_k of pixel points with the gray value $I(u, v) = k$;

(2) Calculating the probability of pixels with each gray value in the image:

$$P(I_k) = \frac{n_k}{n} \tag{11}$$

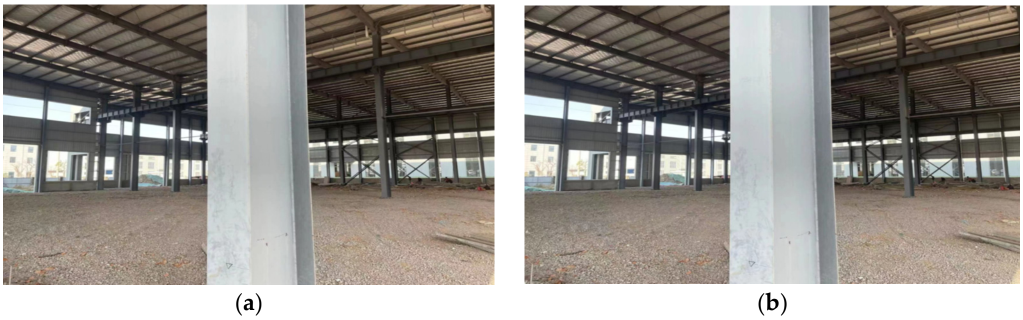
(3) For the pixel points with $I(u, v) = k$ in the original gray image, calculating the gray value of each pixel point after equalization:

$$I'(i, j) = \text{int}\left(255 * \sum_{t=0}^{k} P(I_t) + 0.5\right) = \text{int}\left(255 * \sum_{t=0}^{k} \frac{n_t}{n} + 0.5\right) \tag{12}$$

where, int() is takes the lower integer ceiling function. After the steps above are finished, the dynamic brightness range of the left and right images is balanced to the range of [0, 255]. In this way, the overall brightness difference in gray images is reduced to a certain extent.

4.1.4. Image Preprocessing Experiment

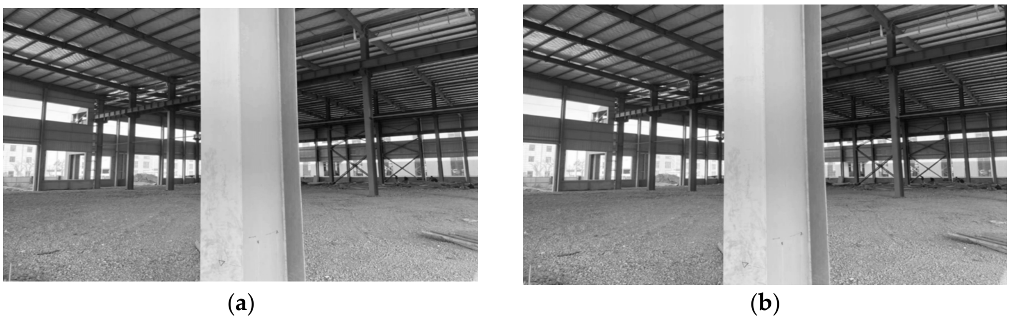
The image preprocessing experiment is completed under the HALCON development environment. For the binocular images obtained (Figure 5), gauss_image() is first used to smooth the binocular images. The side length of the Gaussian template is 3, and σ is set to 0.6. The filtered binocular images are shown in Figure 6.



(**a**) (**b**)

Figure 6. Smooth filtered binocular images: (**a**) is left-eye image; (**b**) is right-eye image.

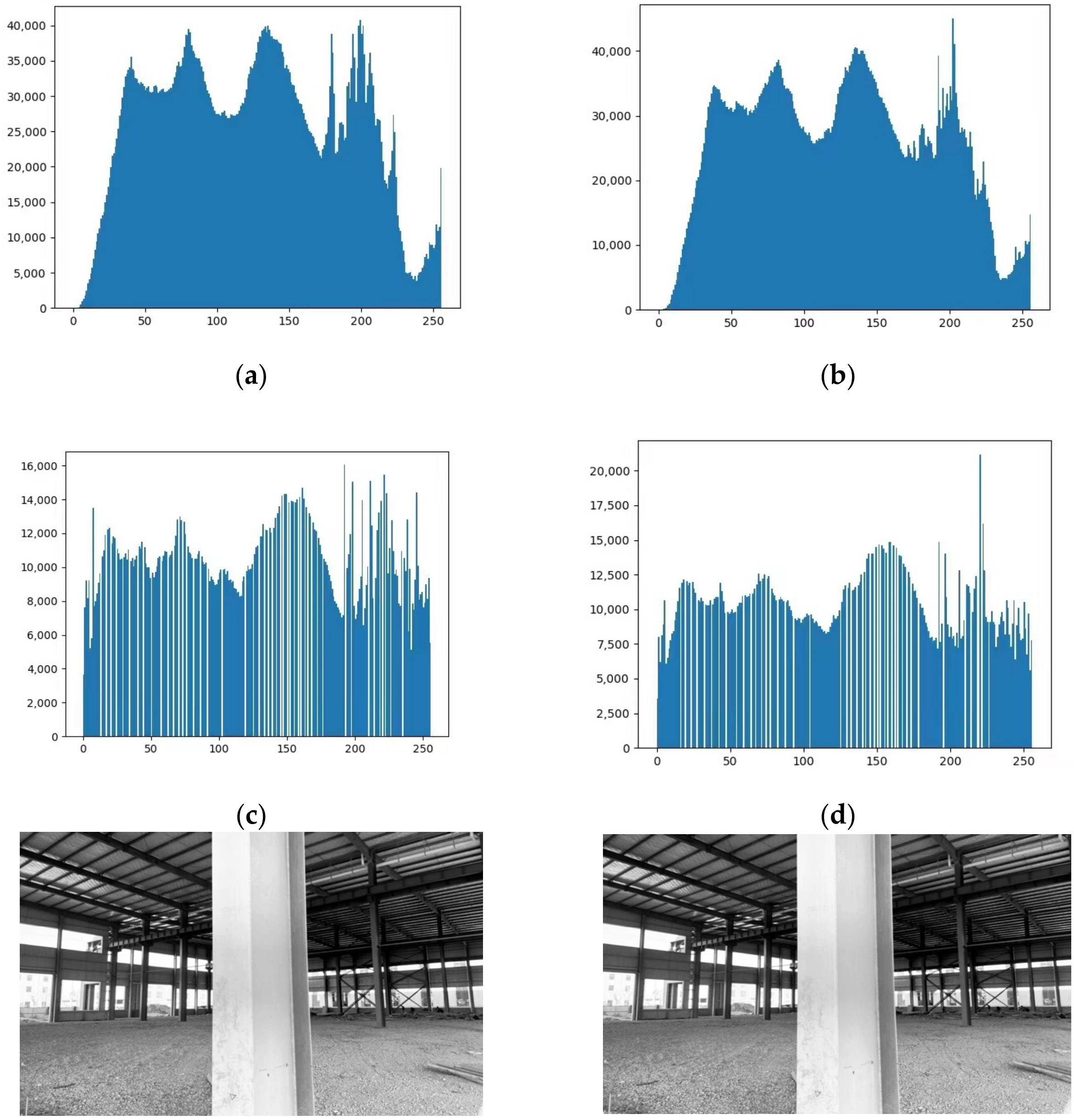
Afterwards, the operator rbg1_to_gray() is used to convert the filtered colored binocular images into the corresponding gray images, as shown in Figure 7a,b.



(**a**) (**b**)

Figure 7. Binocular images after gray processing: (**a**) is left-eye image; (**b**) is right-eye image.

Finally, the operator equ_histo_image() is used to perform histogram equalization. The gray level histograms of the left-eye and right-eye images before equalization are shown in Figure 8a,b, with the abscissa being k and the ordinate being n_k; the equalized gray histograms are shown in Figure 8c,d, and the equalized gray images are shown in Figure 8e,f.



(**a**) (**b**)

(**c**) (**d**)

(**e**) (**f**)

Figure 8. Histogram equalization of gray images: (**a**,**b**) are gray level histograms of the left-eye and right-eye images before equalization; (**c**,**d**) are equalized gray histograms of the left-eye and right-eye images; (**e**,**f**) are equalized gray images of the left-eye and right-eye images.

It can be seen from the histogram equalization results that compared with the gray images before equalization, the gray images after equalization, on the one hand, have a wider dynamic range, more even distribution of pixel gray values and a larger gray gap between pixel points with similar gray values in the original images, which is conducive to subsequent binocular stereo matching. On the other hand, histogram equalization also causes discontinuous changes in the gray values of the images and reduction of gray levels.

The number of gray levels in Figure 8e is 28.17% less than that in Figure 7a, and the number of gray levels in Figure 8f is 26.98% less than that in Figure 7b, which is unfavorable to binocular stereo matching to some extent. Nevertheless, for gray images with unbalanced distribution of the brightness dynamic range and binocular brightness difference, histogram equalization has a positive effect on subsequent binocular stereo matching.

4.2. Binocular Stereo Matching

4.2.1. Parallax and Depth

According to the equivalent ideal binocular camera parameters of the binocular imaging model, the point $P_{\mathrm{Lc}}(x_{\mathrm{Lc}}, y_{\mathrm{Lc}}, z_{\mathrm{Lc}})$ under the left-eye camera coordinate system forms the projection points $P_{\mathrm{Lp}}(u_{\mathrm{Lp}}, v_{\mathrm{Lp}})$ and $P_{\mathrm{Rp}}(u_{\mathrm{Rp}}, v_{\mathrm{Rp}})$ respectively on the left and right imaging planes. The simplified model is shown in Figure 9.

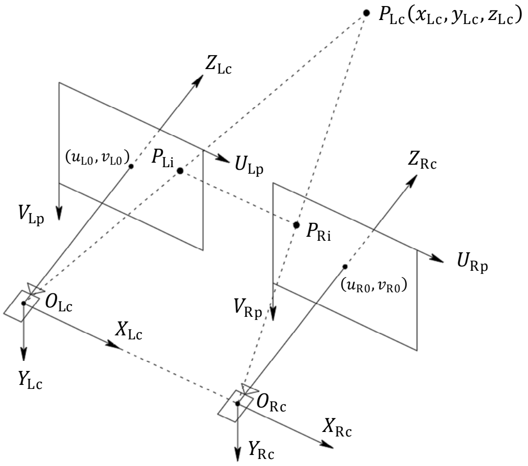


Figure 9. Parallax and depth in parallel binocular imaging.

The corresponding coordinate transformation relationship in the imaging process is:

$$z_{\mathrm{Lc}}\begin{bmatrix} u_{\mathrm{Lp}} \\ v_{\mathrm{Lp}} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f_{\mathrm{c}}}{S_U} & 0 & u_{\mathrm{L0}} & 0 \\ 0 & \frac{f_{\mathrm{c}}}{S_V} & v_{\mathrm{L0}} & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} x_{\mathrm{Lc}} \\ y_{\mathrm{Lc}} \\ z_{\mathrm{Lc}} \\ 1 \end{bmatrix} \tag{13}$$

$$z_{\mathrm{Lc}}\begin{bmatrix} u_{\mathrm{Rp}} \\ v_{\mathrm{Rp}} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f_{\mathrm{c}}}{S_U} & 0 & u_{\mathrm{R0}} & 0 \\ 0 & \frac{f_{\mathrm{c}}}{S_V} & v_{\mathrm{R0}} & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} x_{\mathrm{Lc}} \\ y_{\mathrm{Lc}} \\ z_{\mathrm{Lc}} \\ 1 \end{bmatrix} \tag{14}$$

As a result of binocular correction, $\Delta P_{\mathrm{Lc}} P_{\mathrm{Li}} P_{\mathrm{Ri}} \sim \Delta P_{\mathrm{Lc}} O_{\mathrm{Lc}} O_{\mathrm{Rc}}$ is obtained. In combination with Equations (4) and (5) and Equations (4)–(6) and based on the coordinate $(u_{\mathrm{Lp}}, v_{\mathrm{Lp}})$ of P_{Lp} in the left-eye images, it can be obtained that:

$$z_{\mathrm{Lc}} = \frac{b f_c}{S_U(u_{\mathrm{Lp}} - u_{\mathrm{Rp}} - u_{\mathrm{L0}} + u_{\mathrm{R0}})} \triangleq \frac{b f_c}{S_U(d - d')} \tag{15}$$

$$x_{\mathrm{Lc}} = \frac{S_U(u_{\mathrm{Lp}} - u_{\mathrm{L0}})}{f_{\mathrm{c}}}(z_{\mathrm{Lc}} - f_{\mathrm{c}}) \tag{16}$$

$$y_{\mathrm{Lc}} = \frac{S_V(v_{\mathrm{Lp}} - v_{\mathrm{L0}})}{f_{\mathrm{c}}}(z_{\mathrm{Lc}} - f_{\mathrm{c}}) \tag{17}$$

where, z_{Lc} is also called the depth value (*Depth*) of the space point P under the camera coordinate system {Lc}; b is the baseline distance of the binocular camera after binocular correction; $d = u_{\mathrm{Lp}} - u_{\mathrm{Rp}}$ is the parallax of the binocular images; $d' = u_{\mathrm{L0}} - u_{\mathrm{R0}}$ is the compensation item of parallax, which is a constant related to the principal point coordinate of the corrected binocular camera.

4.2.2. Binocular Stereo Matching Experiment

Binocular stereo matching is the antecedent link of restoring the depth information of space points. Stereo matching algorithm is used to find the matching relationship between the corresponding projection points of space points on the left and right imaging planes of the binocular camera, and the parallax of the corresponding points is calculated based on the left-eye image coordinate system. Then, the corresponding depth images are calculated from the parallax images.

Normalized Cross Correlation (NCC) algorithm uses the average gray level of pixels in the matching window to normalize the gray value of each point in the matching window, thus realizing the compensation for the brightness difference in the matching window. Multi-grid method takes the global energy function as the core, and the components of the global energy function take into account the gray level, the gray gradient and the impact of too large parallax changes. The information of images is used more fully, and the parallax images obtained are more continuous and smoother. This paper compares the local stereo matching algorithm based on NCC with the global stereo matching algorithm based on multi-grid method.

In the construction site environment, steel structure columns are taken as the foreground object, and the camera is facing the rectangular edge of steel structure columns. The horizontal distance between the optical center of the left-eye camera and the rectangular edge of steel structure columns measured by the laser rangefinder is 1.565 m. The algorithm parameters set in this experiment are consistent with those set in the in-laboratory book ranging experiment. The original images and the parallax images obtained through calculation are shown in Figure 10.

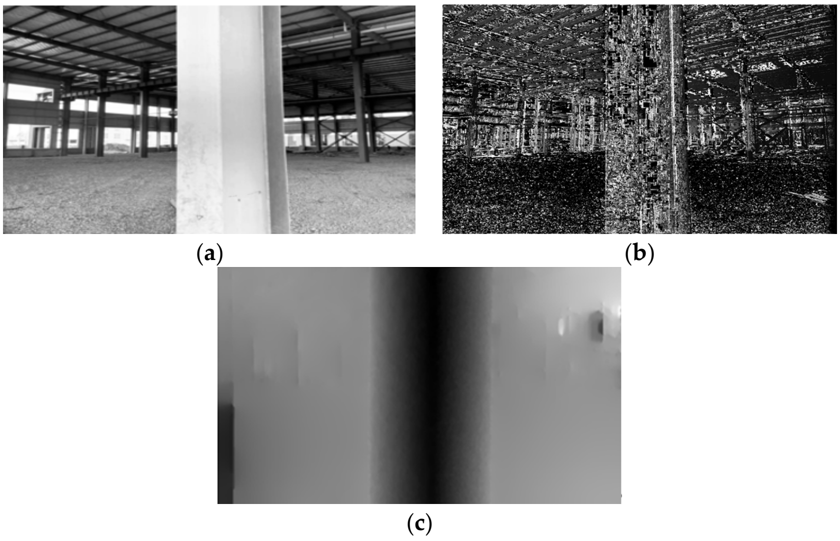


Figure 10. Comparison of algorithm effect in construction site environment: (**a**) is Original Drawing; (**b**) is NCC Algorithm; (**c**) is Multigrid Method.

In regions of steel structure columns shown in Figure 10, because of mismatches by NCC algorithm in the repeated texture regions, there are many parallax holes in the parallax images after the elimination of mismatches. In contrast, the parallax obtained by multi-grid method is uniform and continuous, which is consistent with the actual change in the depth of steel structure columns. The middle of the rectangular edge of steel structure columns is selected as the point for depth measurement. The depth obtained by multi-grid method through visual measurement is 1.548 m, compared with the actual depth of 1.565 m, with the relative error being −1.09% and the absolute error being 0.017 m, meeting the requirement of spraying operation on visual measurement accuracy.

5. Depth Image Object Segmentation

5.1. Workflow and Structure of Multi-Layer Perceptron

Multi-layer perceptron (MLP) is a kind of artificial neural network that has several advantages including a simple structure, intuitive model parameters, easy adjustment and fast model training speed. MLP can achieve better results for the task of pixel classification simple and fixed scenes. However, due to the complexity of the actual construction environment, there may be great changes in the colored images taken at different time periods and different operation positions. It is difficult for MLP to collect training data covering all working conditions. Besides, the pixel classification ability of a single MLP cannot cover all possible construction conditions. To solve the problems above, the method adopted in this paper is to use the color difference between the foreground object and the background image. According to the real-time object image, the assistant operator of the robot selects the local regions of the object in the box on the operation screen as the training set, followed by model training and object segmentation under this data set. The workflow is shown in Figure 11.

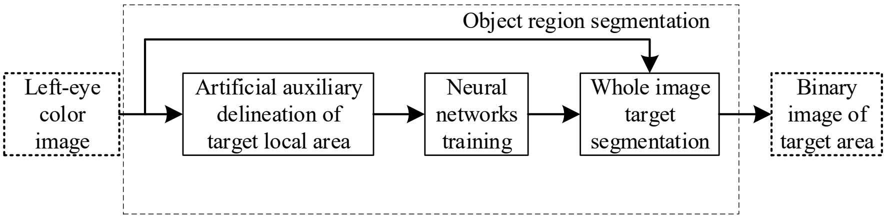

Object region segmentation
Left-eye color image
Artificial auxiliary delineation of target local area
Neural networks training
Whole image target segmentation
Binary image of target area

Figure 11. MLP training and target segmentation workflow.

The MLP network is trained using BP algorithm. Its network structure and the rationale of object segmentation are shown in Figure 12: First, the left-eye colored images are spilt into image channels to obtain R, G and B channel images respectively; Then, for a pixel point in an colored image to be classified, the gray value of the pixel point at the corresponding position in R, G and B channels is taken as the input of MLP, and the network output after classification judgment and mapping is the category the pixel belongs to; the steps above are repeated until the classification of all pixel points in the colored images is completed.

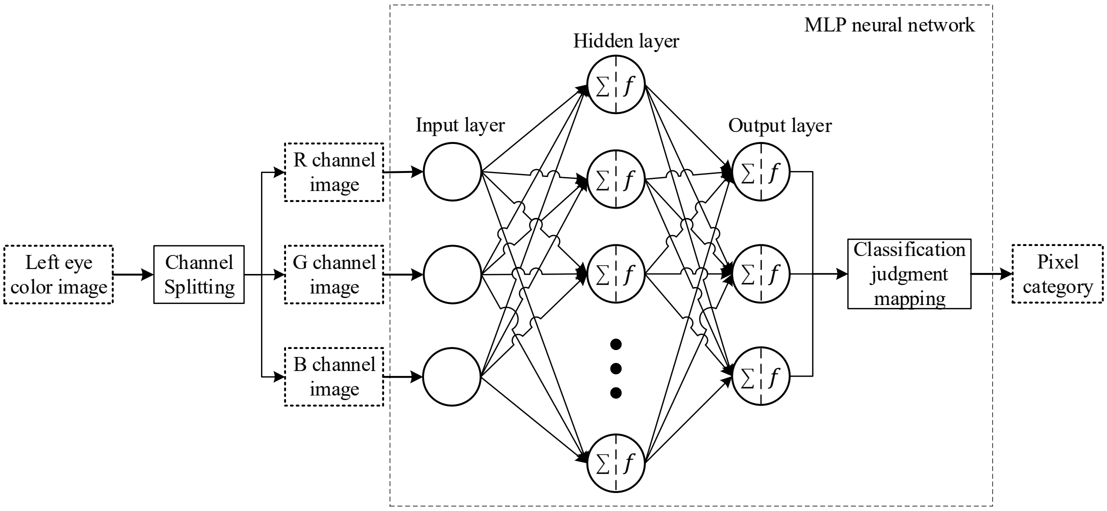


Figure 12. This is a figure. Schemes follow the same formatting.

The neuron node model is shown in Figure 13:

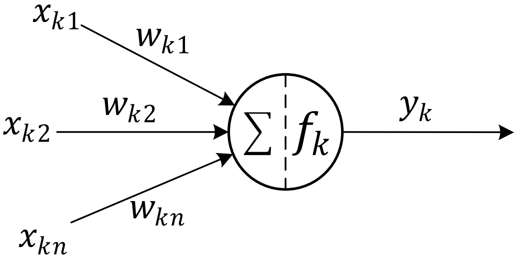


Figure 13. This is a figure. Schemes follow the same formatting.

The input-output relationship of the k-th neuron node is:

$$net_k = \sum_{i=1}^{n} w_{ki} x_{ki} - \theta_k \tag{18}$$

$$y_k = f_k(net_k) \tag{19}$$

where n is the number of the input items of the neuron node; x_{ki} is each input item; w_{ki} is the weight value corresponding to the input item; θ_k is the activation threshold of the neuron node; f_k is the activation function; y_k is the output value of the neuron node.

For neuron nodes in the hidden layer, f_k takes the tanh function as the activation function:

$$f_k = \tanh(net_k) = \frac{\exp(net_k) - \exp(-net_k)}{\exp(net_k) + \exp(-net_k)} \tag{20}$$

For neural nodes in the output layer, in order to carry out the normalization operation to obtain classification results in the form of probability, f_k takes the softmax function as the activation function:

$$f_k = \text{softmax}[\exp(net_k)] = \frac{\exp(net_k)}{\sum_{i=1}^{j} \exp(net_i)} \tag{21}$$

where j is the number of neuron nodes in the output layer, that is, the number of categories. The output value of nodes in the form of probability contributes to more intuitive classification judgment, that is, the classification corresponding to the node with the highest probability value is taken as the category of the pixel.

In this paper, the cross entropy loss function is used as the loss function of the neural network. Compared with the classification error rate loss function and the mean square error loss function, the network error value given by the cross entropy loss function is more stable. The cross entropy loss function is defined as follows:

$$Loss = -\frac{1}{N} \sum_{i} \sum_{c=1}^{j} \hat{p}_{ic} \log(p_{ic}) \tag{22}$$

where $\hat{p}_{ic}$ is the sign function, which takes 1 when the real category of the sample i is consistent with the given category c and takes 0 otherwise; p_{ic} is the predicted probability that the observation sample i belongs to the category c.

In the MLP network structure above, the number of nodes in the input layer is fixed to 3, and the number of nodes in the output layer is set according to the type of the object to be classified. The number of nodes in the hidden layer has great impact on the accuracy of pixel classification, the generalization performance of the neural network, and the computational resources consumed by neural network training. This parameter needs to be adjusted according to the actual pixel classification effect.

5.2. Object Segmentation Experiment

(1) Creation of the Training Set and Neural Network

The training set is generated by manually aided frame selection and label setting, as shown in Figure 14.

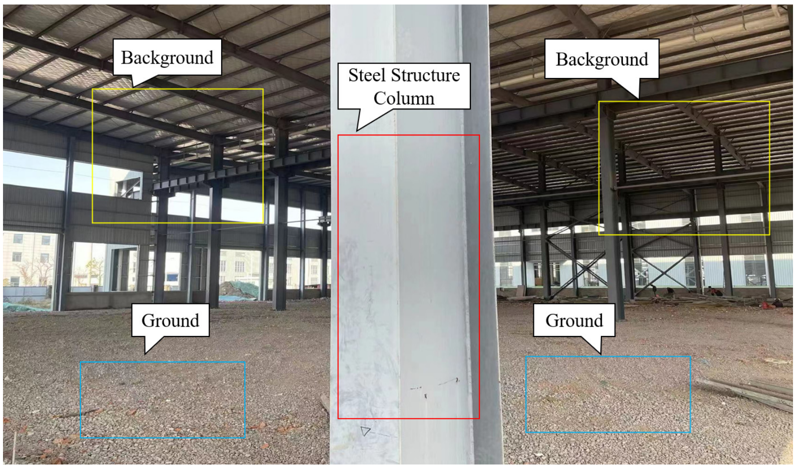


Figure 14. MLP network structure and working principle.

During the creation of the network training set, the regions shown in the label of "Steel Structure Columns" label are assigned a separate category, and the regions shown in the labels "Ground" and "Background" are both classified as "Other". After setting the regions and labels, the operator add_samples_image_class_mlp() is used to generate the training set of the MLP network.

The MLP network is created through the operator create_class_mlp(), of which the main parameters are:

(1) The number of nodes in the input layer is 3; the number of nodes in the hidden layer is 6; the number of nodes in the output layer is 2;
(2) The tanh function is the activation function of the hidden layer;
(3) The softmax function is the activation function of the output layer.

The MLP network is trained through the operator train_class_mlp(). The main parameters of the operator are:

(1) The number of training iterations is 400;
(2) The threshold of weight change is 1;
(3) The threshold of iteration error is 0.1.

The change in the number of misclassified points during training is shown in Figure 15.

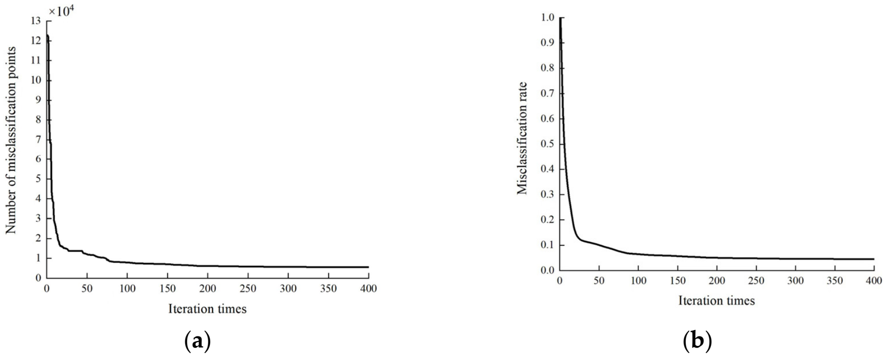


Figure 15. Error variation during MLP training. (**a**) Numbers of misclassification points. (**b**) Misclassification rate.

In Figure 15, the training sample includes a total number of 122,791 pixel points. The number of misclassified points in the sample is reduced to 6176 after the 200th iteration, and is 5564 after the 400th iteration. In the first 25 iterations, the number of classified pixel points decreases rapidly and converges. When the iteration reaches about 200 times, the changing trend of the number of misclassified points is basically stable. The number of misclassified points after 400 iterations accounts for only 4.53% of the training sample.

(2) Object Segmentation

After the MLP network training is completed, the operator classify_image_class_mlp() is used to perform object segmentation on the original images. The segmentation results of the original object are shown in the yellow region in Figure 16.

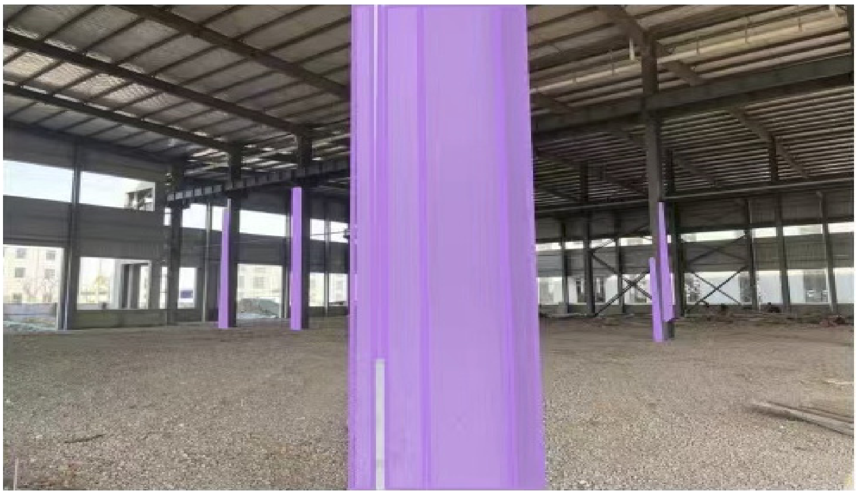


Figure 16. Preliminary segmentation results of steel structure columns.

It can be seen from Figure 16 that the MLP network has a good segmentation effect on the object overall. The main region of steel structure columns has clear edges and accurate segmentation, and there are only sporadic small holes in the region; There are some misclassified pixel points outside the main region, such as in the upper left corner region and the right-side region of the images. Because of the small number of mismatched points, it is possible to process the mismatched points according to regional morphology.

(3) Regional Morphologic Processing

The steps of regional morphologic processing adopted in this paper are as follows:

(1) the operator fill_up() is used to fill the holes in the regions;
(2) the operator opening_rectangle1() is used with a rectangular template to perform an opening operation on the regions and eliminate isolated points, burrs on regional edges and narrow bridges connecting the regions;
(3) the operator select_shape() is used to select regions with the pixel area of the region as the filtering element and retain the regions with a large pixel area;
(4) the operator closing_rectangle1() is used with a rectangular template to perform a closed operation on the regions, fill in notches on region edges, and make region edges flat.

The regions in Figure 16 are processed according to regional morphology, and the results are shown in Figure 17.

It can be seen from Figure 17 that the method of regional morphologic processing adopted in this paper eliminates a small number of original mismatched points, and fill the internal holes in the main region of steel structure columns. The amount of data is greatly reduced while the necessary information for the 3D reconstruction of the object is retained, providing basic data for subsequent surface 3D reconstruction of steel structure columns.

Figure 17. This is a figure. Schemes follow the same formatting.

6. Conclusions

This paper examines the method of object segmentation based on MLP, and gives the specific workflow that can realize object segmentation in practical projects. This approach not only overcomes the shortcomings of the existing recognition methods that are poor in accuracy and difficult to be used widely, but also provides basic data for the subsequent three-dimensional reconstruction, thus making a significant contribution to the research of image processing by spraying robots. The experimental results show that the MLP network, in combination with simple manual assistance to delimit the training sample and necessary regional morphological processing, has a better object segmentation effect in local fixed scenes. Meanwhile, MLP is used to segment the position of the target region in the left-eye-colored images, realizing the task of object segmentation of the depth images.

However, in the actual recognition and characterization process, it is necessary to select the artificial intelligence algorithm which is compatible with the data distribution characteristics in order to achieve the best recognition effect. Subsequent work can be considered to further improve the algorithm so that it can be applied to steel structures, and can shorten the training time.

Author Contributions: Conceptualization, M.Z.; methodology, M.Z.; software, L.Z.; validation, W.H., L.Z., Z.S. and X.L.; formal analysis, M.Z. and L.Z.; data curation, M.Z.; writing—original draft preparation, M.Z.; writing—review and editing, L.Z., G.Z. and Z.S.; visualization, W.H.; supervision, X.L.; project administration, G.Z.; funding acquisition, G.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data sharing not applicable.

Acknowledgments: The authors would like to express their gratitude to all those who helped them during the writing of this paper.

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

References

1. Chen, K.Z.; Yun-Zhang, L.I. A study on construction technology of load-bearing structure of traditional architecture in Taiping district. *Archit. Technol.* **2017**.
2. He, G.; Ahmad, K.M.; Yu, W.; Xu, X.; Kumar, J. A comparative analysis of machine learning and grey models. *Arxiv e-prints* **2021**, arXiv:2104.00871.

3. Wei, B.; Xie, N. On unified framework for continuous-time grey models: An integral matching perspective. *Appl. Math. Model.* **2022**, *101*, 432–452. [CrossRef]
4. Hu, M.; Mao, J.; Li, J.; Wang, Q.; Zhang, Y. A Novel lidar signal denoising method based on convolutional autoencoding deep learning neural network. *Atmosphere* **2021**, *12*, 1403. [CrossRef]
5. Yul, C.J.; Keun, Y.T.; Gi, S.J.; Jiyong, K.; Taewoong, U.T.; Hyungtaek, R.T.; Liu, B. Multi-categorical deep learning neural network to classify retinal images: A pilot study employing small database. *PLoS ONE* **2017**, *12*, e0187336.
6. Zhu, D.; Cai, C.; Yang, T.; Zhou, X. A Machine learning approach for air quality prediction: Model regularization and optimization. *Big Data Cogn. Comput.* **2018**, *2*, 5. [CrossRef]
7. Sun, Y.; Babu, P.; Palomar, D.P. Majorization-minimization algorithms in signal processing, communications, and machine learning. *IEEE Trans. Signal Process. A Publ. IEEE Signal Process. Soc.* **2016**, *65*, 794–816. [CrossRef]
8. Jin, M. Further promotion of quadratic time-varying parameters discrete grey model. *Am. J. Inf. Sci. Technol.* **2018**, *2*, 74. [CrossRef]
9. Shengwu, H.U. Study on deformation of foundation pit based on grey neural network model of genetic algorithm. *Sci. Surv. Mapp.* **2019**, *18*, 365–378.
10. Yu, H.; Xiao, M.; University, N.A. Grey neural network model for prediction of carbon emissions. *Comput. Meas. Control* **2017**, *6*, 562166.
11. Yang, M.; Bian, Y.; Zhang, H.; Liu, G.; Zhang, S. Fire image detection based on support vector machine with improved particle swarm optimization. In Proceedings of the 2020 Chinese Automation Congress (CAC), Shanghai, China, 6–8 November 2020.
12. Liu, C.; Luosang, R.; Yao, X.; Su, L. An integrated intelligent manufacturing model based on scheduling and reinforced learning algorithms. *Comput. Ind. Eng.* **2021**, *155*, 107193. [CrossRef]
13. Wei, J.; Wang, R.; Jin, Y.; Zhang, J. An integrated protection algorithm of intelligent substation. *J. Chang. Univ. Sci. Technol. (Nat. Sci. Ed.)* **2017**, *158*, 108569.
14. Alsghaier, H.; Akour, M. Software fault prediction using particle swarm algorithm with genetic algorithm and support vector machine classifier. *Softw. Pract. Exp.* **2020**, *50*, 407–427. [CrossRef]
15. Liang, Q. Application of Convolution Neural Network (CNN) model combined with pyramid algorithm in aerobics action recognition. *Comput. Intell. Neurosci.* **2021**, *2021*, 6170070. [CrossRef] [PubMed]
16. Xie, C.; Wang, J.; Zhang, Z.; Zhou, Y.; Xie, L.; Yuille, A. Adversarial examples for semantic segmentation and object detection. *IEEE Comput. Soc.* **2017**, *12*, 1378–1387.
17. Parashar, A.; Rhu, M.; Mukkara, A.; Puglielli, A.; Venkatesan, R.; Khailany, B.; Emer, J.; Keckler, S.W.; Dally, W.J. SCNN: An accelerator for compressed-sparse convolutional neural networks. *IEEE Comput. Soc.* **2017**, *24*, 1124–1137.
18. Aziz, M.; Ewees, A.A.; Hassanien, A.E. Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation. *Expert Syst. Appl.* **2017**, *83*, 242–256. [CrossRef]
19. Yassin, I.M.; Jailani, R.; Ali, M.; Baharom, R.; Rizman, Z.I. Comparison between cascade forward and multi-layer perceptron neural networks for NARX Functional Electrical Stimulation (FES)-based muscle model. *Int. J. Adv. Sci. Eng. Inf. Technol.* **2017**, *7*, 215. [CrossRef]
20. Gong, N.; Zhang, C.; Zhou, H.; Zhang, K.; Wu, Z.; Zhang, X. Classification of hyperspectral images via improved cycle-MLP. *IET Comput. Vis.* **2022**, *16*, 468–478. [CrossRef]
21. Htike, K.K. Hidden-layer ensemble fusion of MLP neural networks for pedestrian detection. *Inform. Int. J. Comput. Inform.* **2017**, *41*, 104265.
22. Wang, L.; Qin, Y.; Tao, T. Data modeling of calibration parameter measurement based on MLP model. In Proceedings of the 2019 14th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Changsha, China, 1–3 November 2019.
23. Chi, Z.; Xiangjun, F.; Xianying, Z. Traffic flow forecasting model of correlated roads based on MLP. *J. Chongqing Univ. Technol. (Nat. Sci.)* **2021**, *35*, 129–135. (In Chinese)
24. Yongsheng, L.; Jisheng, Q.; Jianping, H.; Hailei, W.; Shaoting, Q.; Yuan, Q.; Yanping, L. Establishment of microbial growth model for a viation catering by artificial neural network. *Food Technol.* **2010**, *35*, 104–109. (In Chinese)
25. XueSong, D.; Liqun, H.; Buzhong, Z.; Yang, Y.; Qiang, L. Prediction of protein denaturation temperature based on multilayer perceptron. *Comput. Appl. Res.* **2019**, *36*, 2421–2422. (In Chinese)

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