International Journal of Automation and Computing 17(4), August 2020, 562-571
DOI: 10.1007/s11633-020-1226-3

Binocular Vision Object Positioning Method for Robots Based on Coarse-fine Stereo Matching

Wei-Ping Ma Wen-Xin Li Peng-Xia Cao

Lanzhou Institute of Physics, China Academy of Space Technology, Lanzhou 730000, China

Abstract: In order to improve the low positioning accuracy and execution efficiency of the robot binocular vision, a binocular vision positioning method based on coarse-fine stereo matching is proposed to achieve object positioning. The random fern is used in the coarse matching to identify objects in the left and right images, and the pixel coordinates of the object center points in the two images are calculated to complete the center matching. In the fine matching, the right center point is viewed as an estimated value to set the search range of the right image, in which the region matching is implemented to find the best matched point of the left center point. Then, the similar triangle principle of the binocular vision model is used to calculate the 3D coordinates of the center point, achieving fast and accurate object positioning. Finally, the proposed method is applied to the object scene images and the robotic arm grasping platform. The experimental results show that the average absolute positioning error and average relative positioning error of the proposed method are 8.22 mm and 1.96% respectively when the object's depth distance is within 600 mm, the time consumption is less than 1.029 s. The method can meet the needs of the robot grasping system, and has better accuracy and robustness.

Keywords: Object positioning, stereo matching, random fern, normalized cross correlation, binocular vision model.

1 Introduction

As an important part of intelligent robots[1], the robot vision positioning system can acquire image information through visual sensors, and use image processing technology to make the robot have the ability to perceive the space of environmental objects and realize specific tasks such as robot autonomous navigation, industrial sorting and gripping, etc.

At present, the vision positioning system is mainly divided into monocular vision positioning and binocular vision positioning according to the number of visual sensors used. The monocular vision system[2] uses only one camera to obtain the position information of object feature point by establishing a projection transformation relationship between the spatial point and corresponding image point through the camera mathematical model. This method is simple and flexible in structure, and easy in calibration, but its positioning accuracy is low. The binocular vision system[3] imitates the human visual structure, uses two cameras placed at different positions to acquire the scene images of the same object, and calculates the parallax of object feature points in the two images to achieve object positioning, which has a higher positioning accuracy. The key to the binocular vision system is stereo matching, which is needed to select matched object feature points with spatial position consistency in the left and right images. There are usually two solutions: the first solution is to extract local feature points in the left and right images to achieve object matching and positioning, namely feature matching[4]. The solution has high matching precision and robustness with small calculation amounts and fast matching speed. The second one is region matching[5], which can obtain a dense and uniform disparity map. It mainly finds the two points with the highest similarity of the neighborhood sub-windows in the two images to complete the matching, but its robustness is poor when rotation and illumination occur, and it has a high computational complexity. In the binocular vision positioning system for robots, if the feature matching is adopted, the final object positioning point will not be the center point, and the horizontal and vertical distances of the object in the camera coordinate system cannot be accurately obtained. If the region matching is adopted, the huge amount of calculation is a problem. Therefore, an improved matching method is proposed in the binocular vision positioning system. First, taking the object center point as a feature point, the center matching of objects in the left and right images is realized. Second, the matching result is regarded as an estimated value to set the search range of the region matching. Finally, after the above coarse-fine matching, the matched center points obtained in the two images have a better spatial positional consistency, and the obtained 3D coordinates of the object center point have higher precision.

Research Article
Manuscript received August 27, 2019; accepted February 3, 2020; published online April 1, 2020
Recommended by Associate Editor Zhi-Jie Xu

The premise of the center matching is to obtain the pixel coordinates sets of the object area in the two images, in other words, object recognition is required. Generally, the path of object recognition is mainly achieved by extracting the local features of object template and scene image and matching them. Among them, feature extraction and recognition are especially important. Scale-invariant feature transform (SIFT)[6] has good performance in the field of object recognition, but the algorithm complexity cannot meet the system with high real-time requirements. On the basis of ensuring the high specificity of SIFT, speeded up robust features (SURF) accelerates the extraction and matching of features, but it still cannot meet the high real-time requirements. Many research works in the later period have been continuously improved, greatly increasing the efficiency of the SIFT and SURF[7, 8]. The above algorithms are based on the framework of the following ideas: 1) local features extraction, 2) invariant description of features, 3) features matching, 4) calculating corresponding relationship between two images. To improve matching speed and recognition rate, Ozuysal et al.[9] show the random fern algorithm, treating feature matching problems as a simple classification problem. Compared with the traditional natural feature matching methods, the random fern algorithm has outstanding online matching speed, which is widely applied in target tracking[10], augmented reality[11] and face tracking[12]. In addition, some scholars have applied it to visual positioning, Luo et al.[13] show the monocular vision real-time positioning algorithm based on random ferns. Therefore, this paper applies it to the robot binocular vision positioning system to improve the object recognition speed in the left and right images, achieving the fast and accurate center matching.

Based on the center matching, the region matching is added. The object center points in the two images should be a pair of natural matched points, but the object areas identified in the two images are not completely identical, resulting in poor consistency of the left and right center points. So taking the right center point as an estimated value, the pixel search range of the region matching in the right image is set, in which the best matched point of the left center point is found, and the matched object center points with good consistency are extracted. Finally, the similar triangle principle of the binocular vision is utilized to achieve rapid object positioning.

2 Robot positioning system

2.1 Robot platform

The self-developed design of robot platform is adopted in the paper, which can realize the tasks of object recognition, positioning and grasping. The overall structure is shown in Fig. 1, it shows the main hardware components briefly, the visual sensor is the KS861 parallel binocular camera, the actuator uses the YiXueTong 6-DOF (degree of freedom) manipulator, image processing, operation interface display and various communication tasks are done by a PC with Intel Core i3-3217U.1.8 GHz.

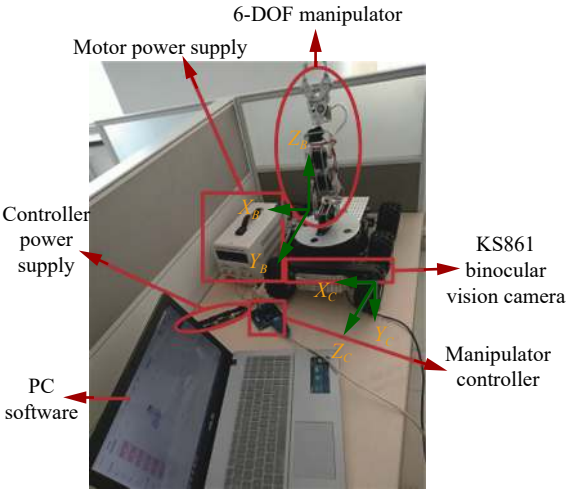


Fig. 1 Hardware structure diagram of the robot platform

The steps for the robot platform to perform grasping task are as follows.

Step 1. The internal and external parameters of the KS861 camera are calibrated to establish the correspondence between the image pixel points and the depth value of a certain spatial point. Thus, the spatial distance of a certain spatial point can be obtained if the image coordinates in the left and right images are known, then the 3D coordinates of the spatial point can be calculated through the conversion relationship between the image coordinate system and the camera coordinate system. The conversion relationships between pixel coordinate values and 3D coordinate values are shown in (13)–(18).

Step 2. Completing hand-eye calibration on the manipulator system in order to obtain the 3D coordinates of the spatial point in the manipulator base coordinate system. The manipulator base coordinate system is regarded as the reference coordinate system to control the manipulator and effector to perform the grasping operation, And the conversion relationship between the camera coordinate system and the manipulator base coordinate system can be described as

$$\begin{bmatrix} X_b \\ Y_b \\ Z_b \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \tag{1}$$

where $\boldsymbol{R}$ is a rotation matrix with size 3×3, $\boldsymbol{t}$ is translation vector with size 3×1.

Step 3. Taking the center of the object as the grasping point, a coarse-fine matching method combining the center matching based on random fern and the region

matching based on normalized cross-correction (NCC) is used to obtain the pixel coordinates of the object center point in the left and right images, then the two pixel points are substituted into the formulas of the Steps 1 and 2, and the 3D coordinates of the spatial point in the manipulator base coordinate system are calculated.

Step 4. The 3D coordinates of the spatial point obtained in the Step 3 is transformed into the sending instructions of the manipulator upper computer control software through inverse kinematics calculation and trajectory planning of the manipulator, and the manipulator is driven to complete the object grasping task.

The most critical technical issue throughout the grasping task is the object positioning, so Sections 3–5 focus on the object positioning of the binocular stereo vision based on the proposed coarse-fine matching method.

2.2 Binocular vision positioning method

The binocular vision system mainly uses the position difference generated by a certain spatial point on the left and right images to recover the depth information of the spatial point and realize the object positioning. The prerequisite for obtaining position difference is to achieve stereo matching. The paper adopts a coarse-fine stereo matching method, i.e., the region matching is performed based on the center matching. The center matching is the coarse matching, the objects in the left and right images are detected by the random fern, and the center coordinates of them are calculated to achieve matching. Considering that the object areas extracted in the left and right images are not completely identical, the two obtained center points will not be consistent, so the right center point is regarded as the estimated value, and the region matching based on NCC is used in the stage of fine matching to obtain a more consistent matching result. In this way, the advantages of fast speed of the center matching and the high consistency of the region matching are utilized. The specific implementation process of the positioning method is shown in Fig. 2. In Sections 3–5, the proposed coarse-fine stereo matching method including the center matching based on random fern, the region matching based on NCC and the binocular visual mathematical model for calculating the 3D coordinates of the object will be introduced in detail.

3 Center matching based on random fern

3.1 Random fern feature recognition and matching

The overall framework of the random fern algorithm is shown in Fig. 3. The random fern uses the Bayesian classification model[14] in the machine learning algorithm to deal with feature recognition and matching, and transfers the huge computational amount generated by the feature description and matching to the classifier. This section uses the naive Bayesian classification model to achieve the classification and matching of object features through classifiers offline training and feature recognition and matching, completing the object recognition of the left and right images.

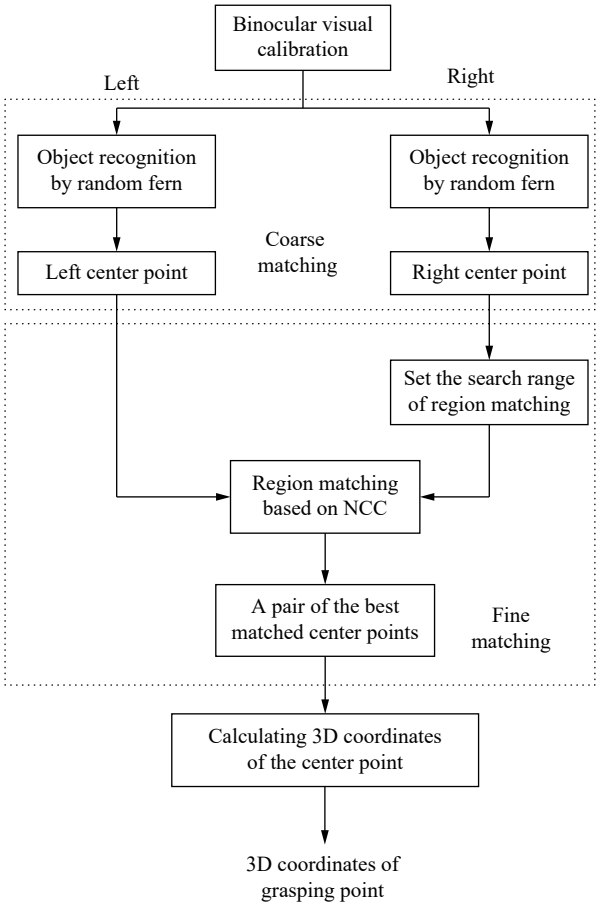


Fig. 2 Positioning diagram

3.1.1 Bayesian classification model

In the process of the random fern feature matching, the feature points[15] of the object image are first collected, and the image patches are generated as the basic unit of recognition and classification. The set of all possible appearances of the image patch surrounding a feature point is treated as a same class. Therefore, given the patch surrounding a feature point detected in an image, our task is putting it into the most likely class. Let $c_i, i = 1, \cdots, N$ be the set of classes and let $f_j, j = 1, \cdots, M$ be the set of binary features that will be calculated over the patch we are trying to classify. The classification problem is described as

$$\widehat{c}_i = \arg\max_{c_i} P(C = c_i \,|f_1\, , \cdots, f_M) \tag{2}$$

where C is a random variable that represents the class. According to the Bayesian formula, (2) can be

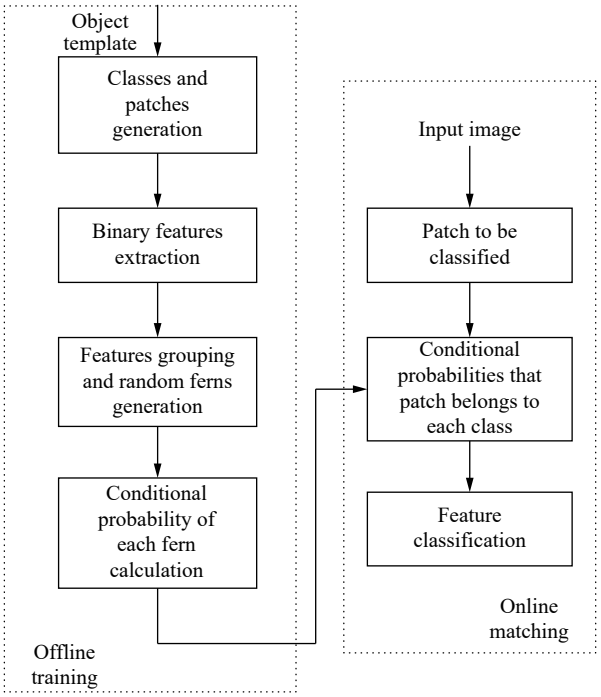


Fig. 3 Overall framework of random fern

decomposed into

$$P(C=c_i\,|f_1\,,\cdots,f_M)=\frac{P(f_1,\cdots,f_M\,|C=c_i)P(C=c_i)}{P(f_1,\cdots,f_M)}. \tag{3}$$

Assuming a uniform prior $P(C=c_i)$, since the denominator is simply a scaling factor that is independent from the class, the problem reduces to

$$\widehat{c}_i=\arg\max_{c_i} P(f_1,\cdots,f_M\,|C=c_i\,). \tag{4}$$

In implementation, the value of each binary feature f_j is calculated as

$$f_j=\begin{cases}1, & \text{if } I(d_{j1})<I(d_{j2})\\ 0, & \text{otherwise}\end{cases} \tag{5}$$

where d_{j1} and d_{j2} are two random selected pixel locations in the image patch, they are a test point pair, I represents the grayscale value.

Assuming that all features are independent with each other, an extreme version of (4) is to assume complete independence, i.e.,

$$\widehat{c}_i=\arg\max_{c_i}\prod_{j=1}^{M} P(f_j\,|C=c_i\,). \tag{6}$$

However, this assumption ignores the relationship between features. In order to ensure the correlation between features and reduce the amount of storage, features are divided into K groups with size $S=M/K$. These groups are defined as Ferns. The features in the fern are correlated with each other, fern and fern are independent with each other. The conditional probability becomes

$$\widehat{c}_i=\arg\max_{c_i}\prod_{k=1}^{K} P(F_k\,|C=c_i\,) \tag{7}$$

where $F_k=\{f_{\sigma(k,1)},f_{\sigma(k,2)},\cdots,f_{\sigma(k,S)}\},k=1,\cdots,K$ represents the k-th fern, and $\sigma(k,j)$ is a random permutation function with range 1–M.

3.1.2 Classifier offline training

In order to train a classifier with strong robustness to image projection deformation, illumination variation, image blur and noise, it is a prerequisite to select stable feature points detected on the object template and form a training sample set for each class.

Affine deformation is a key step in the classifier offline training of the random fern algorithm. It is mainly used to achieve the selection of the stable feature points and the generation of the training samples (the training sample is the image patch), which determines the performance of the entire classifier. Affine deformation is defined as

$$\begin{aligned}&\boldsymbol{A}=\boldsymbol{R}_\theta\boldsymbol{R}_\phi\mathrm{diag}(\lambda_1,\lambda_2)\boldsymbol{R}_{-\boldsymbol{\phi}}\\&\mathrm{diag}(\lambda_1,\lambda_2)=\begin{bmatrix}\lambda_1 & 0\\ 0 & \lambda_2\end{bmatrix}\\&\boldsymbol{R}_{\boldsymbol{\theta}}=\begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix}\\&\boldsymbol{R}_{-\boldsymbol{\phi}}=\begin{bmatrix}\cos\phi & \sin\phi\\ -\sin\phi & \cos\phi\end{bmatrix}\end{aligned} \tag{8}$$

where $\boldsymbol{R}_\theta$ and $\boldsymbol{R}_\phi$ are an object rotation matrix and scale transformation rotation matrix respectively. $\mathrm{diag}(\lambda_1,\lambda_2)$ is a scale transformation diagonal matrix. Affine deformation parameters are set: $\theta,\phi\in[0,2\pi)$, $\lambda_1,\lambda_2\in[0.6,1.5]$.

Next, an affine deformation is used to extract N stable feature points as a stable point set of the object template. Firstly, a certain number of feature points of the object template are extracted, then randomly select the affine parameters, and multiple affine deformations on the object template are performed, a certain number of feature points in each affine view are extracted. After completing all affine deformations, the number of occurrences of each feature point in all affine views is counted, and the feature points with the most occurrences are treated as the most stable points.

The training sample set for each class includes thousands of sample images at different visual angles and scales, which can be generated by randomly picked affine deformation. Specifically, the stable feature points in the object template are taken as the center, the pixel patches are intercepted, and multiple random affine deformations are performed to generate a plurality of pixel patches,

which are used as the elements of the training sample set. In the training process, the rotation factor is taken as the key point, each stable point is regarded as a class, and 10 800 affine deformations are taken for each class. Taking the rotation factor as a loop variable, affine parameters are randomly selected from $1°-360°$, 30 training samples are got in per degree. In addition, to improve the robustness of the classifier to noise and complex scenes, Gaussian noise is added to each sample image.

After selecting the stable feature points and forming the training sample sets, the class conditional probabilities $P(F_k\,|C=c_i\,)$ for each fern F_k and class c_i will be estimated by counting the frequency that ferns of each class occur in the training set. For each fern F_k, we write these terms as

$$P(F_k\,|C=c_i\,)=\frac{n_{k,i}+u}{\sum_k\,(n_{k,i}+u)} \tag{9}$$

where $n_{k,i}$ is the number of training samples of the k-th fern in the class c_i, the denominator represents the number of all training samples in the class c_i, u is a nonzero coefficient and $u=1$.

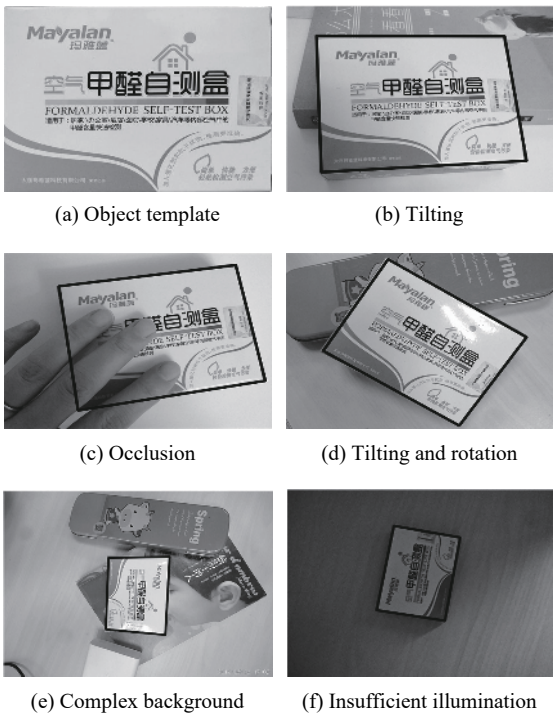
3.1.3 Online feature recognition and matching

During the online feature recognition stage, multiscale feature points of the input image are extracted, the patch surrounding a feature point as item to be classified, then its binary feature set is obtained by (5) for the calculation of conditional probability. Applying the patch to be classified to trained classifier, and the conditional probabilities that binary features belong to each class are counted. Finally, the class with the largest conditional probability is the one which the patch belongs to, and template feature points and input image feature points are identified and matched. Furthermore, the random fern feature matching algorithm can be used for object recognition, and the recognition results in the different conditions are shown in Fig. 4.

Fig. 5 shows the trend of the recognition rate under different parameters after the rough matching and random sample consensus (RANSAC).

According to the change trend of the recognition rate obtained in Fig. 5, it can be known that the parameters that affect the performance of the classifier are the number of classes, the number of ferns, and the number of randomly selected test point pairs. If the number of test points and ferns increases, the recognition rate will increase. If the number of classes increases, the recognition rate will decrease.

Fig. 6 shows the average online matching time of each corner. The online matching time is proportional to the number of ferns. To ensure recognition rate and correct rate, the number of ferns is controlled, whose range is $K\in(20,50)$. In order to achieve a stable recognition effect, the parameters selected in the training classifier are set: the number of classes is 100, the number of test point pairs is 7, and the number of ferns is 30.



(a) Object template (b) Tilting

(c) Occlusion (d) Tilting and rotation

(e) Complex background (f) Insufficient illumination

Fig. 4 Object recognition

3.2 The center matching

The center points of the standard object rectangle regions in the left and right images are a pair of natural matching points, but the two rectangle regions acquired in the object recognition stage are not exactly the same. Therefore, the center matching is just a coarse matching result, which is an estimated value for setting the search range of the region matching in the right image. Knowing the object rectangular regions in the left and right images, the left and right center points can be calculated as

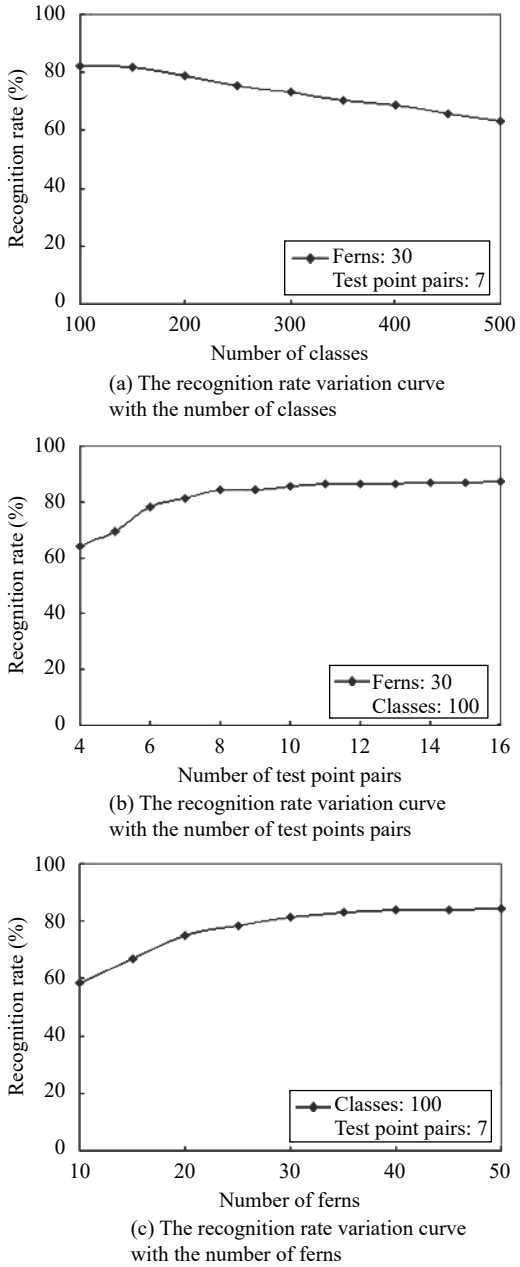
$$u=\frac{1}{4}\sum_{i=1}^{4}u_i \tag{10}$$

$$v=\frac{1}{4}\sum_{i=1}^{4}v_i \tag{11}$$

where (u_i,v_i) are four vertices of the object rectangle in the image. The pixel coordinates of the left and right center points are $C_l(u_l,v_l)$ and $C_r(u_r,v_r)$, respectively.

4 The region matching based on NCC

The region matching is based on the local gray information of the image, and the matching is performed by using the gray value of the image point. In order to find the best matched point of the left center point in the right image, the region matching is further adopted based on the center matching. The specific implementation idea



(a) The recognition rate variation curve with the number of classes

(b) The recognition rate variation curve with the number of test points pairs

(c) The recognition rate variation curve with the number of ferns

Fig. 5 Recognition rate under different parameters

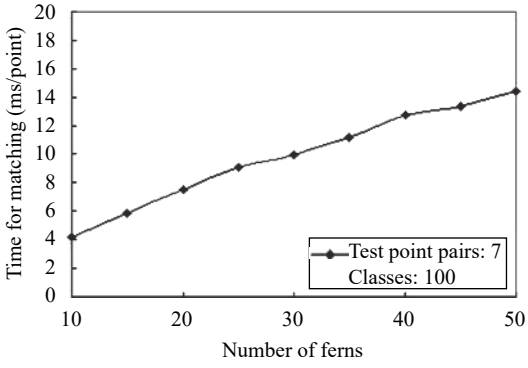


Fig. 6 Matching time of each corner

is to define a rectangular window centering on the left center point, and search for a window with the highest similarity in the right image, the center of the window is the best matched point of the left center point. Among them, the similarity measure method is the key, which directly affects the accuracy and time of the matching algorithm. The normalized cross-correlation algorithm[16] has strong anti-noise interference ability, and its value is not affected by the linear transformation of gray scale. It has good accuracy and adaptability in image matching, as is shown in (12)

$$R_{NCC} = \frac{\sum_{2n+1}\sum_{2n+1}(T(i,j)-\overline{T})(T'(i,j)-\overline{T'})}{\sqrt{\sum_{2n+1}\sum_{2n+1}(T(i,j)-\overline{T})^2}\sqrt{\sum_{2n+1}\sum_{2n+1}(T'(i,j)-\overline{T'})^2}} \tag{12}$$

where $T(i,j)$ is the pixel value of a point in the rectangular window centering on the left center point, $T'(i,j)$ is the pixel value of a point in the rectangular window centering on a candidate matching point in the right image. $\overline{T}$ and $\overline{T'}$ are the mean pixel values of their window, the length of window is $2n+1$.

The pixel search range is the parameter to be determined before the region matching. The right center point obtained in the center matching stage can be used as the estimated value to reflect the approximate range of the best matched point of the left center point. Taking the right center point as center, a narrower pixel range is set for the region matching. In this way, the matching calculation amount is reduced compared with the traditional region matching, and a large amount of time consumption is saved. At the same time, the matching accuracy is improved compared with the single center matching, and the probability of mismatching is reduced. In addition, the values in the *Y*-axis of the matching point pair are same according to the polar line constraint, but the ideal parallel binocular vision model can not be realized. After stereoscopic correction, two image points of a spatial point in the left and right images are on the same polar line as much as possible. In order to further improve the accuracy of matching, ε is set as a small error of the two values in the *Y*-axis between the left center point and the right matched point, the value in the *X*-axis of the right center point is considered, $\boldsymbol{R}$ is the final search range of the region matching, $\boldsymbol{R} = \{p(x,y) \,|x \in (u_r - m, u_r + m), y \in (v_l - \varepsilon, v_l + \varepsilon)\}$, where m is the maximum absolute difference of the values in the *X*-axis between the right center point and the matched point of the left center point. Traversing the pixel point of the $\boldsymbol{R}$, the pixel point having the largest NCC value with the left center point will be the matched point of the left center point.

5 The mathematical model of the binocular stereo vision

The mathematical model of the parallel binocular ste-

reo vision is shown in Fig. 7. In the camera model, four coordinate systems are involved, which are the world coordinate system, the camera coordinate system, the pixel coordinate system and the image coordinate system. The world coordinate system is the three-dimensional coordinate system of scene space, in which the object is located. It is a hypothetical fixed coordinate system, generally selecting a three-dimensional rectangular coordinate system. The camera coordinate system is a space three-dimensional coordinate system with the camera plane as the X-Y plane and the camera optical axis as the Z-axis. The pixel coordinate system is the coordinate system of the camera's photosensitive plane, pixel usually is the basic unit. The image coordinate system is a two-dimensional coordinate system, which is fixed on the digital image, its origin is in the optical center.

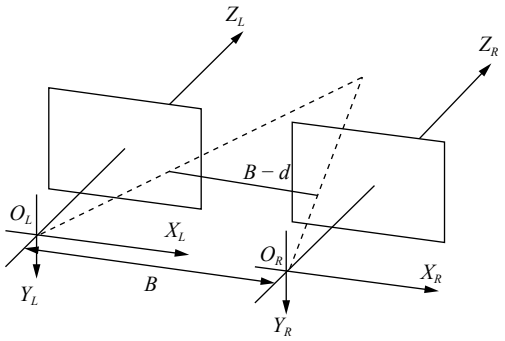


Fig. 7 Parallel binocular stereo vision model

Let $P(X, Y, Z)$ be a spatial point, its corresponding points in the left and right image coordinate systems are $p_l(x_l, y_l)$ and $p_r(x_r, y_r)$, respectively. According to the similar triangle principle[17], the correspondence between image points and depth value of a certain spatial point is established, i.e.,

$$Z_C = \frac{Bf}{x_i - x_r}. \tag{13}$$

At the same time, the conversion relationship between the pixel coordinate system and the image coordinate system is

$$u = u_0 + \frac{x}{\mathrm{d}x} \tag{14}$$

$$v = v_0 + \frac{y}{\mathrm{d}y}. \tag{15}$$

Then, equation (13) can be converted to

$$Z_C = \frac{Bf}{(u_l - u_r)dx} = \frac{Bf_x}{u_l - u_r}. \tag{16}$$

Finally, according to the conversion relationship between the image coordinate system and the world coordinate system (the left camera coordinate system), there is

$$X_C = \frac{x_l}{f} Z_C = \frac{(u_l - u_0)}{f_x} Z_C \tag{17}$$

$$Y_C = \frac{y_l}{f} Z_C = \frac{(v_l - v_0)}{f_y} Z_C \tag{18}$$

where (f_x, f_y) is calibrated camera focal length, (u_l, v_l) and (u_r, v_r) are corresponding points of the spatial point P in the left and right pixel coordinate systems, (u_0, v_0) is the pixel coordinate of the left camera center, B is the baseline length between the left and right cameras.

6 Experiment and analysis

6.1 Camera calibration

After knowing the coordinates of a point in the left and right image coordinate systems, according to the pinhole imaging model and the conversion relationship between the image coordinate system and the world coordinate system, it is necessary to calibrate the camera's internal parameter and the external parameter in order to convert the point of image coordinate system to the point in the camera coordinate system. The paper uses the KS861 parallel binocular camera to capture images with a focal length of 3.6 mm, a resolution of 640×480, and a baseline length of 170 mm between two cameras. Running the calibration program in VS2013 to get the parameters of the binocular camera, the calibration results are shown in Table 1.

Table 1 Camera calibration

	Left camera	Right camera
Internal parameter matrix	$\begin{bmatrix} 462 & 0 & 319 \\ 0 & 464 & 241 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 463 & 0 & 320 \\ 0 & 464 & 242 \\ 0 & 0 & 1 \end{bmatrix}$
Rotation matrix	$\begin{bmatrix} 1.0000 & -0.0036 & 0.0021 \\ 0.0036 & 0.9999 & 0.0032 \\ -0.0020 & -0.0031 & 1.0000 \end{bmatrix}$	
Translation matrix	$\begin{bmatrix} -169.63 & 0.9453 & -1.8956 \end{bmatrix}^{\mathrm{T}}$	

After obtaining the parameters of the stereo calibration, the stereo correction of the parallel binocular stereo vision is performed by using the Bouguet algorithm. The elements in the obtained re-projection matrix including: $(327, 248)$ is the pixel coordinates of camera center, camera focal length is 468 pixels, baseline is 169.61 mm. After stereo correction, corresponding points of a spatial point in the two images are basically on the same polar line.

6.2 Object positioning based on coarse-fine matching

The object is placed at six different positions, and the proposed matching method is used to perform the stereo matching and the positioning of object center point.

Fig. 8 shows the results of object recognition and center points matching in the left and right images when object is placed at the first position, Figs. 8 (a) and 8(b) are the left and right images after stereo correction. Figs. 8 (c) and 8 (d) are object recognition results of the left and right images. The centers of the circles shown in Figs. 8 (e)–8 (g) are the left center point, the right center point and the matched point of the left center point, respectively. In order to show the matching result more clearly, only part of the image in Figs. 8 (e) and 8 (g) is taken. The positioning results of the object center point, that object is placed at six different positions are shown in Table 2, including the pixel coordinates of the center point in two images, the calculated 3D coordinates of the center point and the measured 3D coordinates of the center point. In the experiment, the window length of the region matching is 35, ε is equal to 5, both m and n have a value of 10.

In order to characterize the measurement accuracy of the system and quantitatively analyze the error, the average absolute error[18] is introduced. At the same time, considering the difference of the positioning error caused by the different distances of object from the camera, the average relative error is introduced to eliminate the influence of distance on the positioning results, they are defined as

$$E_a = \frac{\sum_{i=1}^{N} \sqrt{(X_i - X)^2 + (Y_i - Y)^2 + (Z_i - Z)^2}}{N} \tag{19}$$

$$E_r = \frac{\sum_{i=1}^{N} \sqrt{\dfrac{(X_i - X)^2 + (Y_i - Y)^2 + (Z_i - Z)^2}{X^2 + Y^2 + Z^2}}}{N}. \tag{20}$$

According to the data in Table 2, the average absolute positioning error is 8.22 mm, the average relative pos-

(a) Left image after stereo correction (b) Right image after stereo correction

(c) Object recognition of the left image (d) Object recognition of the right image

(e) Left center point (f) Right center point (g) Matched point of the left center point

Fig. 8 Objects recognition and center points matching in the left and right images

Table 2 Object positioning results

Left center point (pixel)	Matched point (pixel)	Calculated 3D coordinates of the center point (mm)	Measured 3D coordinates of the center point (mm)
(530, 311)	(132, 312)	(86.5, 26.9, 195.2)	(84, 25, 190)
(523, 274)	(273, 275)	(133, 17.6, 310.7)	(130, 15, 308)
(483, 269)	(285, 269)	(133.6, 18, 392.3)	(130, 15, 388)
(462, 163)	(293, 165)	(135.5, –85.3, 459.7)	(130, –81, 453)
(440, 177)	(297, 178)	(134, –84.2, 543.2)	(130, –81, 536)
(413, 213)	(286, 215)	(114.9, –46.7, 611.7)	(110, –42, 600)

itioning error is 1.96%, and the positioning error increases with the object distance increasing. Analyzing the error of each coordinate axis, it can be found that the error mainly comes from the Z-axis, because of the object depth information error collected by the vision system.

To further verify the positioning accuracy of the proposed method, three methods including the proposed method, the coarse matching and the fine matching are used for the object positioning respectively, and the positioning results are shown in Table 3. Since the error is derived from the object depth information, only the Z-axis positioning values of each method are compared.

Table 3 Object positioning results of three methods

Measured distances (mm)	Coarse matching (mm)	Proposed matching (mm)	Fine matching (mm)
190	195.7	195.2	195.2
308	314.5	310.7	310.7
388	398.4	392.3	392.3
453	462.4	459.7	–
536	558.9	543.2	543.2
600	611.7	611.7	–

According to the object distance measurement results in Table 3, the proposed method has obvious advantages in positioning accuracy compared with the other two methods. Because the identified left and right object areas of the coarse matching method are not identical, the extracted left and right center points may not match or even differ greatly, so the positioning accuracy depends entirely on the degree of coincidence of the left and right object center points. In the case of no mismatching, the positioning result of the fine matching is the same as the positioning result of the proposed method. However, due to the lack of the coarse matching, the pixel search range of the fine matching is larger, and the probability of mismatching increases ('–' indicates mismatch in Table 3), resulting in the occurrence of incorrect positioning results. The causes of binocular visual positioning errors mainly include: 1) Camera calibration has errors; 2) Stereo matching has matching error; 3) Camera pixel resolution is limited, and the acquired image quality is not good, which will result in positioning error; 4) The measurement values of object center point are inaccurate. In the specific robot grasping task, the value of each coordinate axis is compensated according to the positioning error of the proposed method, which can make the grasping success rate of the robot arm system higher.

6.3 Real-time analysis of the coarse-fine matching

In order to verify the real-time performance of the proposed method, the average positioning time of three methods in Table 3 is counted. The time consumption of the coarse matching is the smallest with 0.746 s, because it only uses the efficient random fern algorithm to identify the object, and the obtained left and right object center points are regarded as a matching result. The proposed method performs the region matching in a small range on the basis of the coarse matching, so the time consumption increases, which is about 1.029 s. The time consumption of the fine matching is about 1.984 s, which is larger than the coarse matching and the proposed method. The main reason is that the number of matching pixels of the fine matching is $640 \times (2\varepsilon + 1)$, the number of matching pixels of the proposed method is $(m + n + 1) \times (2\varepsilon + 1)$, the difference of candidate matching points between two methods is 6 809 according to the experimental parameters setting. It is clear that the matching pixel number of the fine matching is much more than that of the proposed method, so the calculation amount is greatly increased, and its running time is lengthened accordingly.

7 Conclusions

Starting from the object positioning problem of the robot binocular vision system, a binocular vision object positioning method based on coarse-fine stereo matching is proposed. Firstly, the method adopts the random fern, which can quickly and accurately identify the object area in complex object scenes, and obtain the pixel coordinates of the object center points in the left and right images. On this basis, the region matching based on NCC is used to obtain the best matched point of the left center point, and then the 3D coordinates of the object center point are calculated, the positioning result can be applied to the grasping task in the robot platform. The matched center points obtained by the coarse-fine matching method are highly consistent in position, and the proposed matching method has short time consumption and small positioning error when it is used in the binocular vision system, and can meet the real-time and accuracy requirements of the binocular vision positioning system.

Acknowledgements

This work was supported by National Natural Science Foundation of China (No. 61125101)

References

[1] B. Moore, E. Oztop. Robotic grasping and manipulation through human visuomotor learning. *Robotics and Autonomous Systems*, vol. 60, no. 3, pp. 441–451, 2012. DOI: 10.1016/j.robot.2011.09.002.

[2] L. Y. Xu, Z. Q. Cao, P. Zhao, C. Zhou. A new monocular vision measurement method to estimate 3D positions of objects on floor. *International Journal of Automation and Computing*, vol. 14, no. 2, pp. 159–168, 2017. DOI: 10.1007/s11633-016-1047-6.

[3] H. Li, Y. L. Chen, T. H. Chang, X. Y. Wu, Y. S. Ou, Y. S. Xu. Binocular vision positioning for robot grasping. In *Proceedings of IEEE International Conference on Robotics and Biomimetics*, IEEE, Karon Beach, Thailand, pp. 1522–1527, 2011. DOI: 10.1109/ROBIO.2011.6181505.

[4] H. Shi, H. Zhu. Stereo matching based on adaptive matching windows and multi-feature fusion. *Pattern Recognition and Artificial Intelligence*, vol. 29, no. 3, pp. 193–202, 2016. DOI: 10.16451/j.cnki.issn1003-6059.201603001. (in Chinese)

[5] J. Liu, J. X. Zhang, Y. Dai. Dense stereo matching based on region growing. *Robot*, vol. 39, no. 2, pp. 182–188, 2017. DOI: 10.13973/j.cnki.robot.2017.0182. (in Chinese)

[6] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004. DOI: 10.1023/B:VISI.0000029664.99615.94.

[7] L. R. Zhao, W. Zhu, Y. G. Cao, Y. H. Liu, J. X. Sun. Application of improved SURF algorithm to feature matching. *Optics and Precision Engineering*, vol. 21, no. 12, pp. 3263–3271, 2013. DOI: 10.3788/OPE.20132112.3263. (in Chinese)

[8] R. J. Zhu, Y. H. Zhu, L. Wang, W. Lu, H. Luo, Z. C. Zhang. Cotton positioning technique based on binocular vision with implementation of scale-invariant feature transform algorithm. *Transactions of the Chinese Society of Agricultural Engineering*, vol. 32, no. 6, pp. 182–188, 2016. DOI: 10.11975/j.issn.1002-6819.2016.06.025. (in Chinese)

[9] M. Ozuysal, M. Calonder, V. Lepetit, P. Fua. FAST keypoint recognition using random ferns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 448–461, 2010. DOI: 10.1109/TPAMI.2009.23.

[10] Y. Luo, J. Xiang, M. J. Yan, J. H. Hou. Online target tracking based on mulitiple instance learning and random ferns detection. *Journal of Electronics & Information Technology*, vol. 36, no. 7, pp. 1605–1611, 2014. DOI: 10.3724/SP.J.1146.2013.01358. (in Chinese)

[11] Y. Zhao, J. J. Li, H. P. Li, D. Yang. Real-time tracking and registration algorithm of scenarios of augmented reality based on improved random fern. *Journal of Northeastern University (Natural Science)*, vol. 37, no. 5, pp. 614–618, 2016. DOI: 10.3969/j.issn.1005-3026.2016.05.002. (in Chinese)

[12] J. M. Liu, Y. Liang, H. X. Sun, Y. Duan, X. Liu. Real-time face tracking based on detecting and tracking. *Journal of Image and Graphics*, vol. 20, no. 11, pp. 1473–1481, 2015. DOI: 10.11834/jig.20151106. (in Chinese)

[13] Y. Luo, Y. Fu, Y. Zhang. A monocular-vision real-time matching algorithm based on FAST corners and affine-improved random ferns. *Robot*, vol. 36, no. 3, pp. 271–278, 2014. DOI: 10.3724/SP.J.1218.2014.00271. (in Chinese)

[14] F. Zheng, G. I. Webb. A comparative study of semi-naive Bayes methods in classification learning. In *Proceedings of the Fourth Australasian Data Mining Conference*, University of Technology, Sydney, Australia, pp. 141–156, 2005.

[15] E. Rosten, T. Drummond. Machine learning for high-speed corner detection. In *Proceedings of the 9th European Conference on Computer Vision*, Springer, Graz, Australia, pp. 430–443, 2006. DOI: 10.1007/11744023_34.

[16] M. Hu, X. J. He, X. H. Wang. Fast image matching algorithm with area centroid. *Journal of Electronic Measurement and Instrument*, vol. 25, no. 5, pp. 455–462, 2011. DOI: 10.3724/SP.J.1187.2011.00455. (in Chinese)

[17] Y. Yang, F. Qiu, H. Li, L. Zhang, M. L. Wang, M. Y. Fu. Large-scale 3D semantic mapping using stereo vision. *International Journal of Automation and Computing*, vol. 15, no. 2, pp. 194–206, 2018. DOI: 10.1007/s11633-018-1118-y.

[18] H. B. Zhang, S. R. Liu, B. T. Zhang. Binocular vision position algorithm using hue-saturation histogram backproject combined with feature point extraction. *Control Theory & Applications*, vol. 31, no. 5, pp. 614–623, 2014. DOI: 10.7641/CTA.2014.30679. (in Chinese)

Wei-Ping Ma received the B. Eng. degree in electronic information science and technology from Xi'an University of Science and technology, China in 2011, and M. Eng. degree in communication and information system from Xi'an University of Science and technology, China in 2015. Currently, she is a Ph. D. degree candidate in space electronics at Lanzhou Institute of Physics, China Academy of Space Technology (CAST).

Her research interests include space electronic technology, computer vision and intelligent robotics.

E-mail: 498938802@qq.com (Corresponding author)

ORCID iD: 0000-0002-2317-253X

Wen-Xin Li received the M. Eng. degree in applied mathematics from Northwestern Polytechnical University, China in 1993, and Ph. D. degree in automatic control from Northwestern Polytechnical University, China in 2011. Currently, he is a researcher at Lanzhou Institute of Physics, CAST.

His research interests include space electronic technology, software reuse technology, system simulation and reconstruction technology.

E-mail: lwxcast@21cn.com

Peng-Xia Cao received the B. Eng. degree in communication engineering from Hunan International Economics University, China in 2011, and M. Eng. degree in circuits and systems from Hunan Normal University, China in 2015. Currently, she is a Ph. D. degree candidate in space electronics at Lanzhou Institute of Physics, CAST.

Her research interests include space electronic technology, computer vision and augmented reality.

E-mail: 316657294@qq.com