

Article

Line-Features-Based Pose Estimation Method for the Disc Cutter Holder of Shield Machine

Zhe Xie, Guoli Zhu *, Dailin Zhang , Dandan Peng, Jinlong Hu and Yueyu Sun 

School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China

* Correspondence: glzhu@mail.hust.edu.cn; Tel.: +86-27-8754-3555

Abstract: To achieve automatic disc cutter replacement of shield machines, measuring the accurate pose of the disc cutter holder by machine vision is crucial. However, under polluted and restricted illumination conditions, achieving pose estimation by vision is a great challenge. This paper proposes a line-features-based pose estimation method for the disc cutter holder of the shield machine by using a monocular camera. For the blurring effect of rounded corners on the image edge, a rounded edge model is established to obtain edge points that better match the 3D model of the workpiece. To obtain the edge search box corresponding to each edge, a contour separation method based on an adaptive threshold region growing method is proposed. By preprocesses on the edge points of each edge, the efficiency and the accuracy of RANSAC linear fitting are improved. The experimental result shows that the proposed pose estimation method is highly reliable and can meet the measurement accuracy requirements in practical engineering applications.

Keywords: visual positioning; rounded edge model; contour extraction; line detection



Citation: Xie, Z.; Zhu, G.; Zhang, D.; Peng, D.; Hu, J.; Sun, Y. Line-Features-Based Pose Estimation Method for the Disc Cutter Holder of Shield Machine. *Sensors* **2023**, *23*, 1536. <https://doi.org/10.3390/s23031536>

Academic Editor: Gregorij Kurillo

Received: 5 December 2022

Revised: 23 January 2023

Accepted: 27 January 2023

Published: 30 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

A shield machine is a kind of large construction machine commonly used in tunneling construction, such as subways, highways, and railroads [1]. During the shield tunneling process, the shield machine head rotates and drives the disc cutters to cut up the rock. However, the disc cutters are prone to wear and tear during tunneling, and replacing them is an expensive and time-consuming activity that affects the speed and efficiency of shield tunneling [2,3]. The existing disc cutter replacement method is manually done and needs assisted mechanical tools. The working conditions are hazardous, especially in some special construction environments [4]. In the case of a mud-water balanced shield, the safety risk of disc cutter replacement work is high as the construction workers have to enter the high-humidity and high-pressure work bin. Therefore, the automation of disc cutter replacement is very necessary and has become one of the important research directions in the field of shield construction [5,6]. During automatic replacement, visual positioning is usually selected as a high-precision positioning method to locate the disc cutter holder accurately.

In practice, due to the narrow replacement space, the camera has very limited permissible mounting space and must be mounted on the end of the disc cutter replacement robot. Multiple cameras can obtain a lot of information but have insufficient mounting space, and the accuracy of the depth camera that conforms to the mounting dimensions cannot meet the requirement. On the contrary, the monocular vision measurement mode provides a feasible method for the visual system in the disc cutter replacement robot.

The vision positioning algorithm is divided into two steps: key features acquisition and feature-based pose calculation. The key features on the object must be simple and easy to identify, such as points [7,8], circles [9], lines [10,11], etc. The disc cutter holder has typical line features but no obvious corner points, and in reality, there may be a certain amount of mud on the surface. Thus, how to extract linear features quickly and accurately from

the image under pollution is an important issue for the research. Existing line detection methods can be divided into two categories: one is to obtain the linear features directly from the gradient information in the grayscale image, such as LSD [12], Linelet [13], etc; the other is to obtain edge points through image preprocessing and then extract the straight lines on the target boundary by combining the methods of straight line recognition, such as Hough transformation line detector [14], etc. The former cannot handle relatively complex edge cases and is particularly sensitive to noise; in contrast, the second method is more effective in processing complex images. To obtain image edge, the commonly used methods are edge detectors [15,16] and image-segmentation methods to extract contours. For edges with uneven illumination and blurred boundaries, the traditional edge detection operators cannot extract edges well. The commonly used sub-pixel edge extraction methods to handle blurred edges rely more on the characteristics of the image edge itself and do not consider some specific factors that cause edge blurring, such as rounded corners. Among the image segmentation methods, the region-growing method [17] is the most commonly used, but it is less efficient in processing high-resolution images.

Since the edge points on each edge of the disc cutter are easily separated, fitting a straight line to each edge becomes a two-dimensional data fitting problem, which is faster than global detection methods such as probability Hough transform [18], EDlines [19], Cannylines [20], etc. The commonly used methods are the least square method and the RANSAC fitting method [21]. The RANSAC algorithm-based linear fitting method has higher robustness and accuracy, but the disadvantage is that the running time is more affected by the number of outlier points and the efficiency is lower.

For features-based pose calculation, there are many mature methods, including Perspective-n-Point (PnP) methods [22–25] and Perspective-n-Line (PnL) [26–28] methods. For the disc cutter holder positioning, Peng proposes a PnP method that aims to minimize the sum of the chamfering distances [29,30], which must intersect the corresponding feature lines to obtain the feature points. However, in practice, each extracted edge may not be complete, resulting in additional errors in the process of finding the intersection points.

The contributions of this paper are shown as follows: first, to address the blurring effect of rounded corners on image edges, we propose a rounded edge model to obtain edge points that better match the 3D model of the workpiece by the edge search box. Second, we propose a contour separation method based on an adaptive threshold region growing method to obtain the edge search box corresponding to each edge under uneven illumination. Third, by performing some preprocessing on the edge points of each edge, we improve the efficiency and accuracy of the RANSAC linear fitting method.

The rest of this paper is organized as follows. Section 2 states the hardware framework of the vision measurement system. Section 3 describes our proposed linear feature extraction algorithm. Section 4 shows the result of the simulation of the state-of-the-art PnP method and PnL method on this problem. In Section 5, experiments are implemented to test the feasibility of the practical engineering application of our method. In Section 6, the conclusions are given.

2. Introduction to Hardware Framework

Figure 1 shows the schematic diagram of the disc cutter replacement robot and the installation position of the vision system. For the convenience of the study, we use the hardware framework shown in Figure 2 in the laboratory. In the framework, the pose-estimation module, which consists of a calibrated camera and an LED array light source, is mounted at the end of the robotic arm. Due to the extremely limited mounting space allowed on the tool-changing robot, a smaller LED array light source is selected for lighting. Considering the completeness of the features, the inner contour of the disc cutter holder with line features was selected for positioning. In practice, each edge may have clay and defects, and the effects of these factors also need to be taken into account in the design phase. How to accurately extract the linear features in the image is a key step in pose estimation. The pose information is transmitted to the robot cabinet, which controls the

robot manipulator to complete disc cutter replacement. The requirement for position accuracy must be within 1mm, and the required accuracy for attitude must be within 1° .

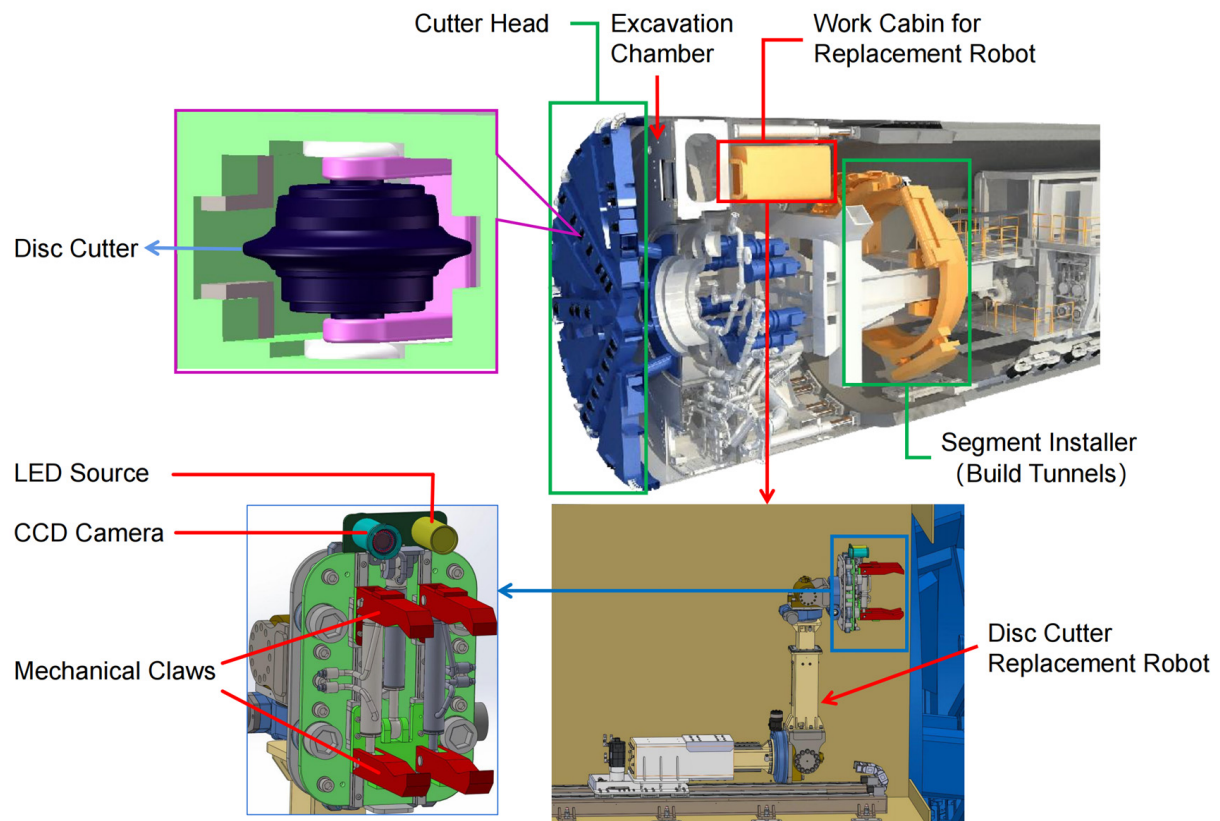


Figure 1. Schematic diagram of the disc cutter replacement robot.

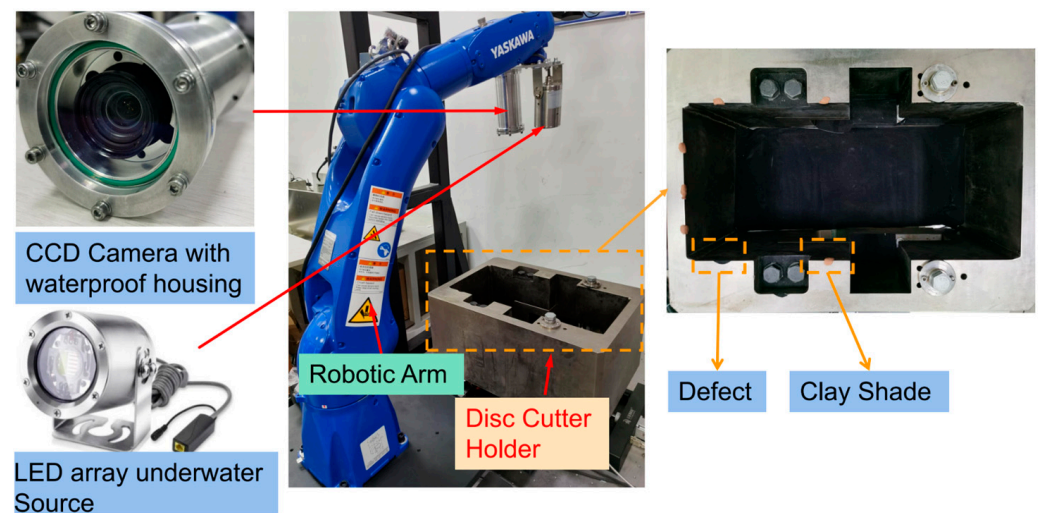


Figure 2. The hardware framework of the pose-estimation module in the lab.

3. Proposed Method

The flow chart of our proposed method is shown in Figure 3. First, we acquire the image of the disc cutter holder from the calibrated camera. After Gaussian filtering and image brightening, we downsample the image and use the region growing method with an adaptive threshold to obtain the inner contour and then separate the contour of each edge. Then, the corresponding edge search box is generated based on the contour of each edge. Among the edge search lines in the edge search box, the edge points that better match the

model are obtained after processing the one-dimensional grayscale curve on the search line based on the rounded edge model. For all edge points on an edge, a preprocessing-based RANSAC linear fitting algorithm is used to obtain the linear feature. Finally, we estimate the pose based on the 3D model of the disc cutter holder with linear features of the image by using the MinPnL algorithm. The rounded edge model, the edge search box generation method, and the preprocessing-based line fitting algorithm of our proposed method are detailed as follows.

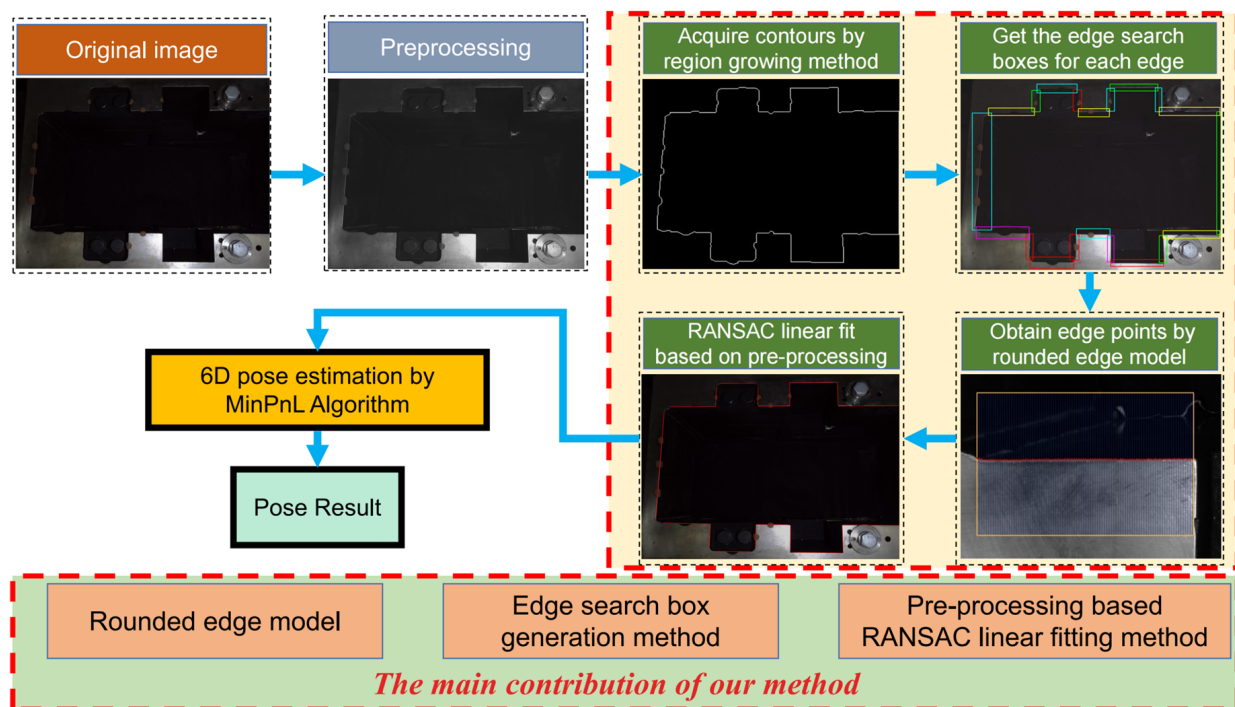


Figure 3. The flowchart of our proposed method.

3.1. Rounded Edge Model

To accurately obtain the line features of each edge of the inner contour of the disc cutter holder, it is necessary to extract the edge points. As shown in Figure 4, the traditional edge detector could not extract the edge of the disc cutter holder image well. There are two reasons, one is that the gradient of each part of the image varies greatly due to uneven illumination, which is not convenient for the separation of the edge points of the edges. The other is that the rounded corners on the edges make the gradient of the image maximum within a certain pixel width (about 15 pixels), and the location of the true edge points cannot be obtained directly from the image.

Due to the effect of the light source and rounded corners, the edges on the image of the disc cutter holder can be divided into two types: strongly illuminated edges and weakly illuminated edges, which present different edge characteristics, respectively. As is shown in Figure 5a, the grayscale change at the weakly illuminated edge shows an overall trend of a step-down, close to the ramp model in the edge model, but due to the existence of noise (such as scratches on the workpiece surface), where the fastest gradient change is not the real edge point. The one-dimensional grayscale curve of 400-pixel points near a strongly illuminated edge is shown in Figure 5b. The grayscale curve at the strongly illuminated edge shows an overall trend of abrupt change, close to the roof model in the edge model within a certain width (10 pixels), where the point with the largest grayscale value cannot be considered as the true edge point.

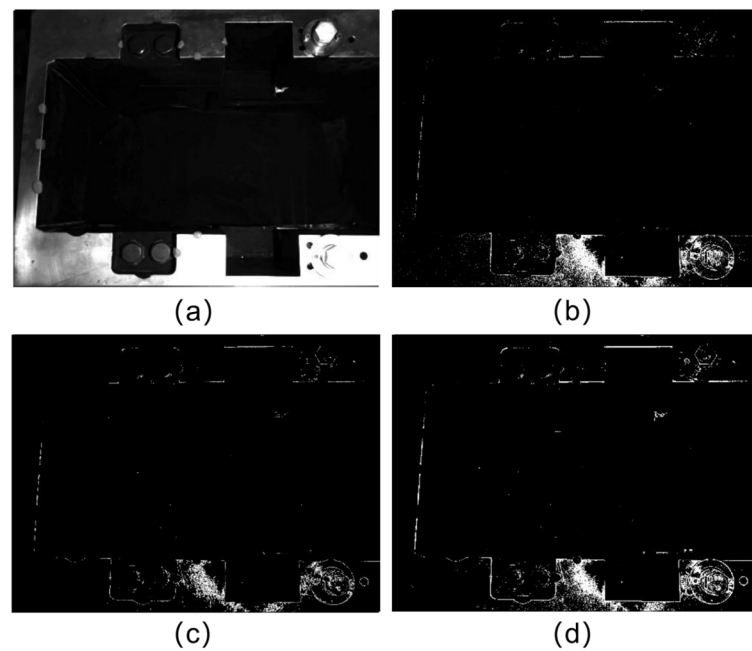


Figure 4. Results of edge detector: (a) Original image; (b) Laplacian's Outcome; (c) Canny's Outcome; (d) Sobel's Outcome.

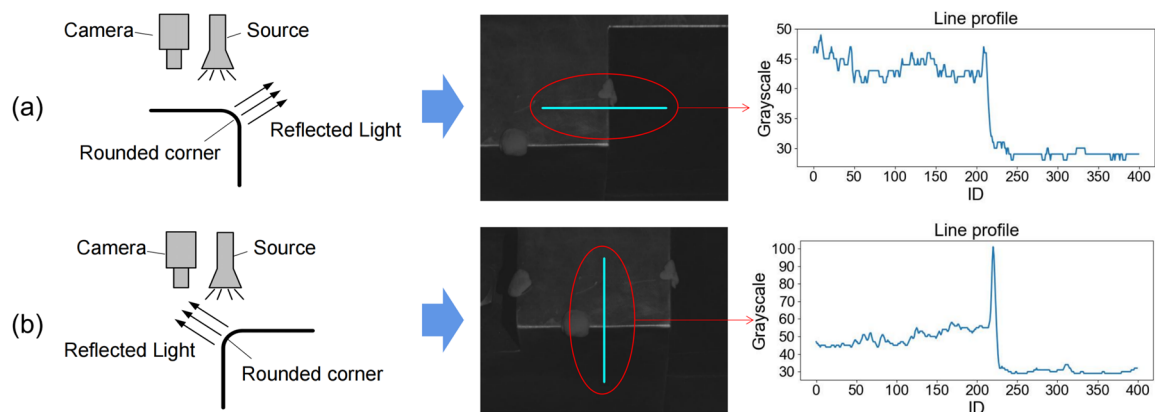


Figure 5. The line profiles of two edge types: (a) Weakly illuminated edge; (b) Strongly illuminated edge.

To obtain more accurate edge points from the image, we construct two edge models with rounded corners as follows.

3.1.1. Strongly Illuminated Edge Model

In Figure 6, the plane to be measured is Π . In the actual measurement, the imaging plane of the camera and Π is close to parallel, so each key point can be projected onto the imaging plane to reflect the situation. In this case, the camera picks up the light reflected by the rounded corners, which is reflected in the image as a higher luminance value than the adjacent area. Note that the brightest point on the corner of the circle is P_2 , the projection point of which on Π is P_2' . The center of the corner is O_r , the starting point is P_1 , the end point is P_3 , and the projection point of P_3 on Π is P_3' , and the radius of the corner is r . When there is no rounded corner, the theoretical edge point is noted as P_0 . The key points on Π correspond to the grayscale changes as shown in the figure. Where P_1 corresponds to the starting point of the grayscale increase, P_2' corresponds to the maximum grayscale value, P_3' corresponds to the endpoint of the grayscale decrease, and P_0 corresponds to somewhere in the grayscale decrease. O_c represents the center of the camera, and Q is the

intersection of the optical axis and Π . The distances from O_c to Q , and from P_0 to Q are b and a , respectively. In the established coordinate system $O_c - xy$ the coordinates of the following points can be obtained as:

$$P_1(a+r, b), P_3(a, b+r), P_3\left(\frac{ab}{b+r}, b\right), P_0(a, b) \quad (1)$$

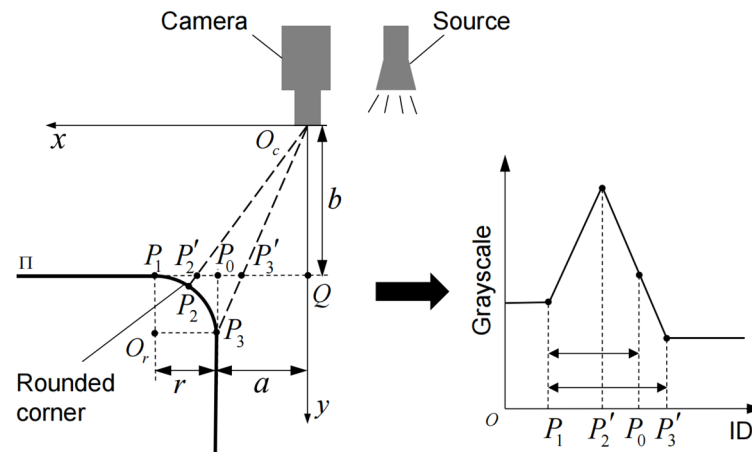


Figure 6. Schematic diagram of strongly illumination edge model.

With these coordinates, the length relationship is calculated as:

$$\frac{\overline{P_1P_0}}{\overline{P_0P_3'}} = \frac{b+r}{a} \quad (2)$$

Denote the coefficient α as the ratio of the distance between the real edge point and the edge start point to the length of the edge segment. Due to $b > a \gg r$, α can be written as:

$$A = \frac{\overline{P_1P_0}}{\overline{P_1P_3'}} \approx \frac{b}{a+b} \quad (3)$$

In the actual measurement, $a \approx 200\text{mm}$, $b \approx 600\text{mm}$, $\alpha \approx 0.75$. Therefore, the point in the one-dimensional grayscale change curve located at the edge of the strong illumination, whose distance from the edge start point is 0.75 times the width of the edge segment, is considered to be the actual edge point.

3.1.2. Weakly Illuminated Edge Model

Similarly, as shown in Figure 7, the plane to be measured at the disc cutter holder is noted as Π . When the reflection of the round corner is away from the camera, the camera will not receive the light from the light source reflected at the round corner, which is reflected in the image as a ramp edge. The center of the corner is O_r , the starting point is P_1 , the endpoint is P_3 , and its radius is r . The center of the camera is O_c , the line O_cP_2 is tangent to $\widehat{P_1P_3}$ at P_2 , and P_2' is the projection point of P_2 on Π . When there is no rounded corner, the theoretical edge point is identically noted P_0 . The corresponding grayscale changes of these key points on Π are shown in the figure, where P_1 corresponds to the starting point of the grayscale increase, P_2' corresponds to the endpoint of the grayscale decrease and P_0 corresponds somewhere in the grayscale decrease. O_c represent the center of the camera, and Q is the intersection of the optical axis and Π . The distance from O_c to Q is b , and the distance from P_0 to Q is a . A coordinate system $O_c - xy$ is established.

According to the geometric relationship, $\angle O_c P_2' Q = \alpha$, $\angle P_1 P_2' O_r = \angle P_2 P_2' O_r = \theta$, where $\alpha + 2\theta = \pi$. Let $P_1 P_2' = x$, we can reach:

$$\frac{b}{a - r + x} = \frac{2 \cdot \frac{r}{x}}{\left(\frac{r}{x}\right)^2 - 1} \quad (4)$$

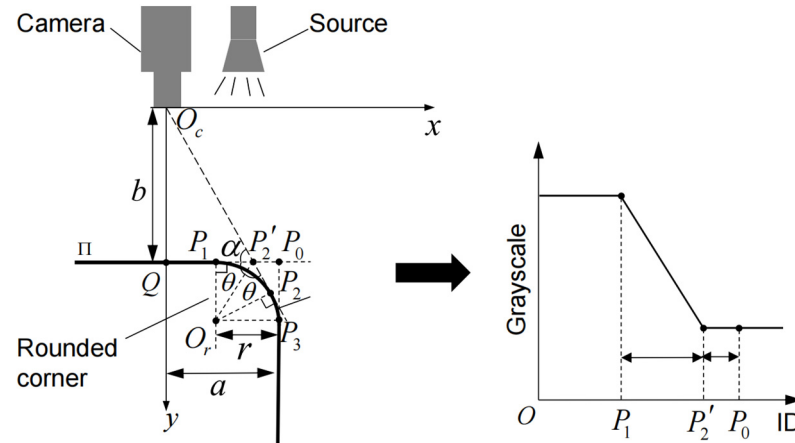


Figure 7. Schematic diagram of weakly illumination edge model.

Equation (3) can be expanded as:

$$(b + 2r)x^2 + 2r(a - r)x - br^2 = 0 \quad (5)$$

This is a quadratic equation and is easy to solve. After eliminating the negative solution, x is calculated as:

$$x = \frac{-r(a - r) + r\sqrt{(a - r)^2 + b(b + 2r)}}{(b + 2r)} \quad (6)$$

Due to $b > a \gg r$, x can be written as:

$$x \approx \frac{-a + \sqrt{a^2 + b^2}}{b} r \quad (7)$$

Denote the coefficient α as the ratio of the distance between the real edge point and the edge endpoint to the length of the edge segment. α is calculated as:

$$\alpha = \frac{P_2' P_0}{P_1 P_2'} = \frac{r - x}{x} = \frac{a + b - \sqrt{a^2 + b^2}}{\sqrt{a^2 + b^2} - a} \quad (8)$$

In the actual measurement, $a \approx 200$ mm, $b \approx 600$ mm, $\alpha \approx 0.387$. Therefore, the point in the one-dimensional grayscale change curve after the end of the grayscale descent section, whose distance from the edge endpoint is 0.387 times the width of the edge segment, is considered to be the actual edge point.

3.1.3. Edge Search Box

To summarize, the edge model with rounded corners consists of two types: one is the roof model when the camera can receive light reflected from rounded corners, and the actual edge point is in the grayscale descent section; the other is the ramp model when the camera cannot receive light reflected from rounded corners, and the actual edge point is at the lower level of grayscale value after the grayscale descent section. The position of

the actual edge point is related to the length of the grayscale edge segment, the working distance, and the distance between the edge and the camera's optical axis.

As shown in Figure 8, to obtain all edge points on an edge side, an edge search box is used to search the edge points by searching lines inside. For each search line, the one-dimensional grayscale variation curve is obtained from the starting position to the ending position, and the edge point on the search line is calculated according to our proposed rounded edge model after median filtering. By setting several parallel search lines in the search box, we can reach all the edge points on the edge side. Meanwhile, there is a certain search interval between two adjacent search lines to improve the search efficiency. The effect of edge point extraction on one of the edges is shown in Figure 9, and the result shows the rounded edge model can accurately extract the edge points on the image of the disc holder.

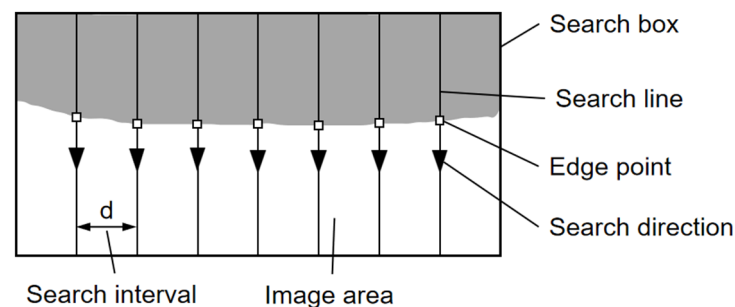


Figure 8. Schematic diagram of the edge search box.

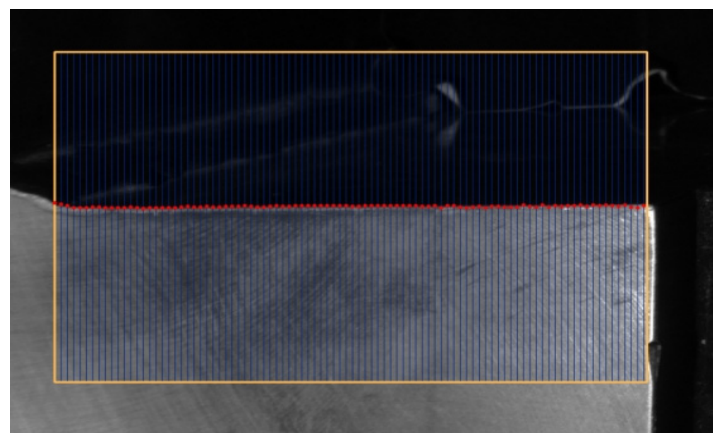


Figure 9. The effect of edge point extraction.

3.2. Edge Search Box Generation Method Based on Improved Region Growing

To obtain the position of the edge search box corresponding to each edge, it is necessary to make a rough estimate of the position of each edge of the disc cutter holder. There is a large grayscale difference between the surface to be measured and the background part. As a common method of image segmentation, the region-growing method is used to separate them. In the case of high image resolution, the region-growing method takes a longer time. Therefore, to improve the processing efficiency, we downsample the original image and reduced the image resolution from 4112×3008 to 411×300 pixel size (the region growth time was reduced from 300 s to 1 s). The contour is roughly at the center of the image, so we can directly set the image center point as the seed point. After the seed point is selected, the most critical thing is the selection of the growth threshold. In the image of the disc cutter holder, due to the large size of the holder and uneven illumination, the grayscale distribution of the disc cutter holder surface is different, so it is difficult to select a fixed threshold to separate the inner contour of the disc cutter holder, and the adaptive growth threshold method can better meet the image processing requirements.

We adopt an adaptive threshold region growth method based on the local grayscale average, which appropriately increases the threshold level when the local gray average near the point to be grown is large and decreases the threshold level when the local grayscale average is small to ensure the integrity of the extracted disc cutter holder internal contour. The specific steps are shown in Table 1.

Table 1. Steps of the contour extraction method based on improved region-growing.

Input:	The grayscale image of the disc cutter holder
Step1:	Set the image center point to the seed and add it to the sequence Q to be grown. Take a point S_b from Q and calculate the mean gray value G_m of a square pixel area centered on S_b . The growth threshold T at S_b is set as
	$T = \left(\frac{G_m}{30} + 1 \right) * T_{base} \quad (9)$
Step2:	Calculate the absolute value of the difference G_{diff} between the grayscale of each ungrown point in the eight-neighborhood of S_b . If $G_{diff} \leq T$, add the ungrown point to Q . After calculating all eight-neighborhood points, add S_b to the growth completion sequence E .
Step3:	Repeat Step (2) if there are remaining points in Q .
Step4:	After the growth is completed, mark all points in E as 1 and other points as 0 in a new binary image to reach a background binary image.
Step5:	Extracts the outer contour of the background binary image, which is the inner contour edge of the disc cutter holder
Output:	The inner contour edge of the disc cutter holder

As shown in Figure 10, there are a few areas that cannot be processed well by the fixed threshold. On the contrary, the inner contour is extracted completely well by the adaptive threshold based on the local grayscale average. After testing, the best extraction effect is achieved when T_{base} is set to 2.

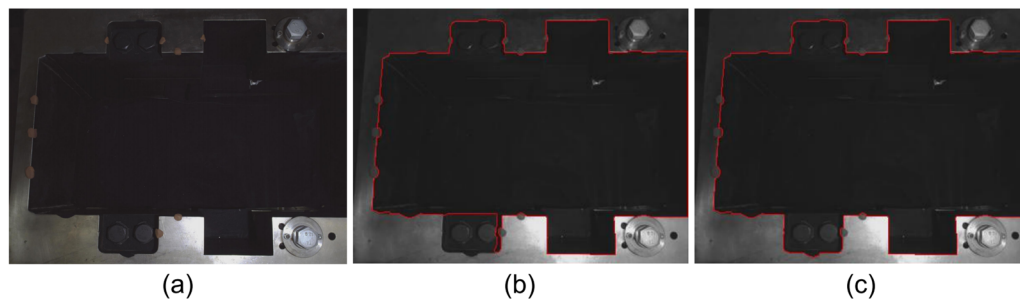


Figure 10. Results of region growing methods (a) Original image; (b) By fixed threshold; (c) By adaptive threshold based on the local grayscale average.

After extracting the inner contour of the disc cutter holder, we need to separate the contour of each edge. Since the shape of the inner contour of the disc cutter holder is relatively regular, we can first separate the horizontal edge points from the vertical edge points in the direction of the edge points to segment each edge. We determine whether the edge point belongs to the horizontal or vertical edge point based on the sum of the absolute value of the gradient difference between the X and Y directions of the adjacent points of each edge point. After separating the edge points by direction, however, due to the presence of interference, the contour of each edge does not consist of individual vertical or horizontal edge points but also requires the edge connection to form a complete edge. After finding each edge, it is easy to find the smallest rectangle that can contain all the edge points of this edge, and the search box corresponding to each edge is gotten after expanding the rectangle along the shorter edge. The specific steps are shown in Table 2.

Table 2. The steps of the rectangular search box generation method.

Input:	The inner contour edge of the disc cutter holder
	For each edge point $P_i(X_i, Y_i)$ in the inner contour, calculate the sum of the absolute values of the gradient difference in the X direction as: $D_x = \sum_{i=-2}^1 X_{i+1} - X_i \quad (10)$
Step1:	and also calculate the sum of the absolute values of the gradient difference in the Y direction as $D_y = \sum_{i=-2}^1 Y_{i+1} - Y_i \quad (11)$
	If $D_x > D_y$, the edge point is judged as a vertical edge point; otherwise, it is judged as a horizontal edge point.
Step2:	Set a distance threshold dis and a count threshold N . If the number of edge points between two edge segments in the same direction is less than dis , as the same edge segment is considered, all edge segments are retained containing more edge points than N .
Step3:	Take the leftmost vertical edge section as the starting mark, and mark all the edge sections as the corresponding edge of the disc cutter holder in order.
Step4:	For each edge of the disc cutter holder, calculate the rectangle that can completely wrap it, and then expand the shorter side of the rectangle three times to generate the edge search box.
Output:	The edge search box of each edge.

Figure 11 shows the result of our proposed edge search box generation method in a binary image of the disc cutter holder's inner contour. Figures 11b and 11c show the horizontal and vertical contour points for each edge, where every edge is well separated. The search boxes for each edge are shown in Figure 11d. The result indicates that this algorithm can automatically generate edge search boxes very well.

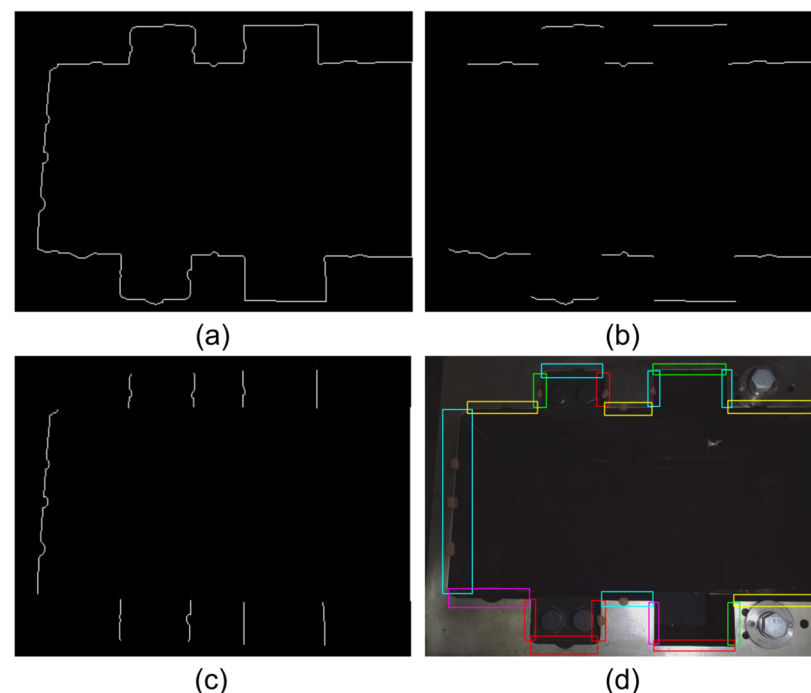


Figure 11. Results of contour separation and edge search box generation: (a) Original contour; (b) Contour points for horizontal edges; (c) Contour points for vertical edges; (d) Edge search boxes.

3.3. RANSAC Linear Fitting Algorithm Based on Preprocessing

The RANSAC algorithm is a commonly used method for iterative robust estimation of model parameters from a data set, which can better reduce the influence of outliers on the results. However, the iteration number of the RANSAC algorithm is closely related to the approximate proportion of outliers in the data, so it is necessary to preprocess the data to eliminate the more obvious wrong edge points to improve the efficiency of the algorithm.

Unlike the general random number of data points to fit a straight line, in the previous step, we used the edge search box tool to obtain the edge points; the interval of each edge point is fixed and known, so the obtained edge point data can be treated as a linear sequence, and the value of each item is the distance from the starting line. Let the sequence of edge points as $EP = [X_1, X_2, \dots, X_n]$, and it is processed based on three types of processing: Continuity Processing, Linearity Processing, and Co-linear Processing.

A. Continuity Processing

For the edge points on each edge of the disc cutter seat, the type of contamination is the presence of defects and mud obscuration on the edge of the seat itself. Regardless of the type of contamination, it manifests itself as a continuous interval effect in the edge point sequence. The purpose of the continuous interval treatment is to remove abruptly varying discrete points from the edge points. The processing steps are shown in Table 3.

Table 3. Steps of the Continuity Processing.

Input:	The sequence of edge points $EP = [X_1, X_2, \dots, X_n]$.	
	Calculate the sequence value difference between each edge point and the previous edge point as follows:	
Step1:	$X_m' = X_{m+1} - X_m, m = 1, 2, \dots, n - 1$	(12)
	Obtain the difference sequence $EP_D = [X_1', X_2', \dots, X_{n-1}']$. For each point in EP_D , judge whether its absolute value exceeds the given difference threshold d_{max} . If $ X_i' > d_{max}$, determine it as a mutation point CP_i , and the edge points between two mutation points CP_i and CP_{i+1} form a continuous interval of edge points $CQ_i = [X_p, X_{p+1}, \dots, X_q]$. The set of all continuous intervals is denoted as $CQ_{ALL} = \{CQ_1, CQ_2, \dots, CQ_n\}$.	
Step2:	For each continuous interval CQ_i in CQ_{ALL} , calculate the interval length of CQ_i as $len(CQ_i) = q - p + 1$, and determine whether the number of edge points $len(CQ_i)$ included in CQ_i exceeds the given length threshold L_{min} . If $len(CQ_i) < L_{min}$, remove CQ_i from CQ_{ALL} . Finally, we can obtain a set of continuous intervals CQ_{ALL} that satisfy the length requirement.	
Output:	The set of continuous intervals CQ_{ALL} .	

B. Linearity Processing

Since the shape of the continuous interval formed by the wrong edge points is generally irregular and cannot be well fitted to a straight line, it can be used as a judgment criterion to determine the type of the continuous interval. The processing algorithm is shown in Table 4.

Compared to the exact linear fitting that follows, LD_{min} can be set a little larger here.

C. Co-linear Processing

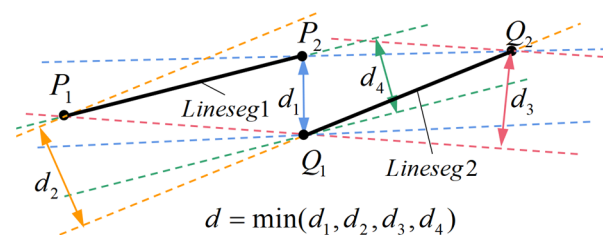
Since clay shade and edge, defects break the edge sequence into several linear intervals, erroneous linear intervals generally do not have linear intervals that are co-linear with them, only intervals that have co-linear features with others will be saved. The processing algorithm is shown in Table 5.

Table 4. Steps of the Linearity Processing.

Input:	The set of continuous intervals CQ_{ALL} .
Step1:	For each CQ_i in CQ_{ALL} , perform linear fitting $\hat{y} = ax + b$ by least squares with the sequence number of each edge point as x and the sequence value as y . Calculate the mean square error (MSE) of the residual of the fitted straight line from each point in CQ_i as
Step2:	$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y})^2 \quad (13)$
Step3:	Judge whether MSE exceeds the threshold LD_{min} , and if $MSE > LD_{min}$, remove CQ_i from CQ_{ALL} . The lastly retained CQ_{ALL} is denoted as the set of linear intervals LQ_{ALL} .
Output:	The set of continuous intervals LQ_{ALL} .

Table 5. Steps of the Co-linear Processing.

Input:	The set of continuous intervals LQ_{ALL} .
Step1:	Calculate the “Line Segment Distance” for every two linear intervals as shown in Figure 12. We define “Line Segment Distance” as the minimum distance between two parallel lines that completely contain two line segments. In practice, the linear interval is approximated as the line segment represented by the two endpoints. For example, note that P_1 and P_2 are the endpoints of a line segment LS_1 , and note that Q_1 and Q_2 are the endpoints of another line segment LS_2 . If P_1, P_2, Q_1, Q_2 form a quadrilateral $P_1P_2Q_2Q_1$, d_{11} is the distance from P_1 to the line P_2Q_2 , d_{12} is the distance from Q_1 to the line P_2Q_2 , the minimum distance d_1 of the line P_2Q_2 and its parallel line which can completely envelop two line segments can be written as $d_1 = \max(d_{11}, d_{12})$. Similarly, we can obtain the minimum distance d_2, d_3, d_4 between lines Q_1Q_2, P_1Q_1, P_1P_2 and there corresponding parallel lines that can completely envelop the line segment LS_1 and LS_2 . Thus, we can obtain the line segment distance LS_{Dis} between LS_1 and LS_2 as $LS_{Dis} = \min(d_1, d_2, d_3, d_4)$. If the line segment distance LS_{Dis} between two linear intervals is less than the threshold value CL_t , the two linear intervals are considered to have a common line relationship and are retained, and the linear interval that does not have a co-linear relationship is deleted from LQ_{ALL} . Retain the longest interval if the line segment of all linear interval pairs is greater than the threshold value. Save all edge points corresponding to linear intervals in LQ_{ALL} as a good set of edge points GEP_{ALL} .
Output:	The good edge point set GEP_{ALL} .

**Figure 12.** Diagram of Line Segment Distance.

The processing effects of each step are shown in Figure 13. After continuity processing, most of the mutation noise is removed. Multiple edge segments caused by clay and defect are eliminated by linearity processing. Finally, after co-linear processing, we obtain the correct edge point of the inner contour of the disc cutter holder.

Figure 14 shows the flow chart of the RANSAC linear fitting method based on pre-processing. The iteration time and fitting error results are shown in Table 6, which indicate that our proposed preprocessing method can effectively reduce the iteration time and improve the accuracy of the RANSAC algorithm on this problem.

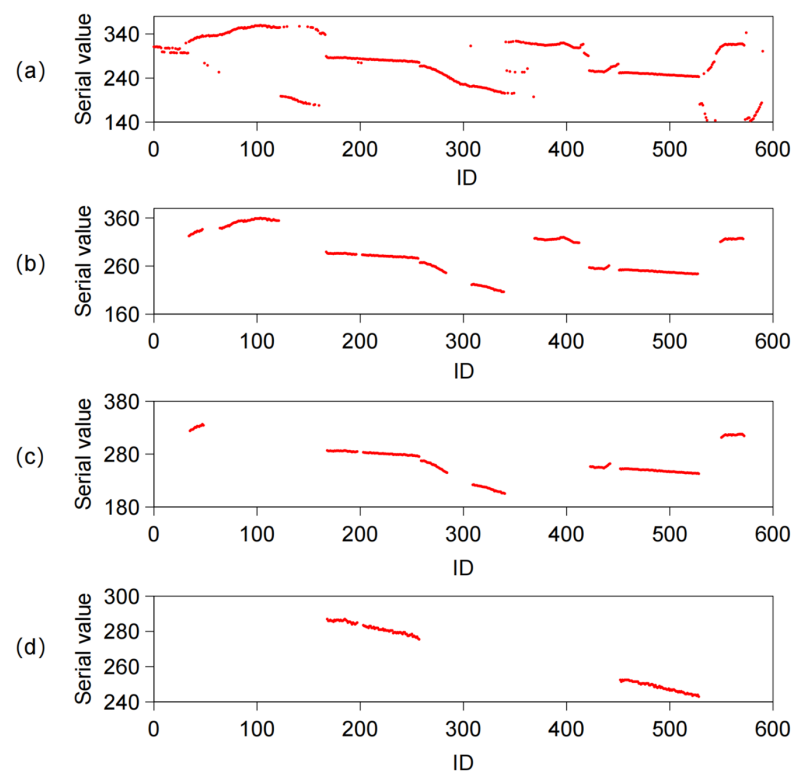


Figure 13. Results of line fitting preprocessing: (a) Original sequence; (b) Continuity Processing's Outcome; (c) Linearity Processing's Outcome; (d) Co-linear Processing's Outcome.

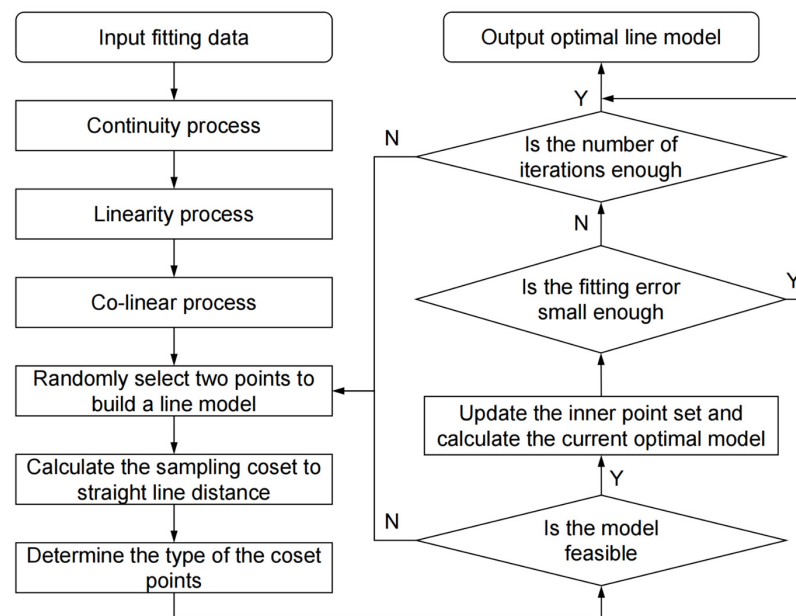


Figure 14. Flow chart of RANSAC linear fitting based on pre-processing.

Table 6. The iteration time and fitting error results.

Method	Iteration Time (ms)	Fitting Error (pixel)
RANSAC line fitting	38.251	0.482
RANSAC line fitting with pre-processing	8.810	0.408

4. Simulation for Pose Calculation Algorithms

After obtaining the linear features of each edge of the tool holder, to calculate the pose of the disc cutter holder, there are two options: one is to use the linear features directly to calculate the pose by the PnL algorithm; the other is to intersect the lines to reach the corresponding feature points and then calculate the pose by the PnP algorithm. To compare the solution accuracy of the PnP algorithm and the PnL algorithm, we simulate the state-of-the-art algorithms ASPnP and MinPnL by adding random Gaussian noise and masking a part of the straight line.

In the simulation experiment, a virtual camera with a resolution of 4112×3008 pixels is synthesized with a focal length of 8 mm and a pixel size of $1.85 \mu\text{m}$. The motion range of the disc holder in space is $[-10^\circ, 10^\circ]$ in all three rotation angles, $[0 \text{ mm}, 50 \text{ mm}]$ in the X direction, $[-50 \text{ mm}, 50 \text{ mm}]$ in the Y direction, and $[500 \text{ mm}, 700 \text{ mm}]$ in the Z direction. First, we randomly generate a pose in the above pose space and obtain the image coordinates of the 20 straight lines of the internal contour of the tool holder. Considering the actual situation, all lines in the image are randomly shortened (to ensure that the total length of the retained lines is not less than 0.6 times the length of the original complete line), and random noise is added at the end points of the lines under the normal distribution, where the variance of the noise increases from 0 to 5 pixels. After obtaining all intersection points, we calculate the absolute error in six degrees of freedom. The test is repeated 500 times. The final error is expressed as the standard deviation of all results.

The results are shown in Figure 15. In all six degrees of freedom, the error of the ASPnP algorithm is larger than the MinPnL algorithm. It is worth noting that in some cases, the ASPnP algorithm has larger solution errors due to intersection calculation errors, whereas the MinPnL algorithm has higher stability in comparison. Therefore, we choose the MinPnL algorithm for calculating the pose.

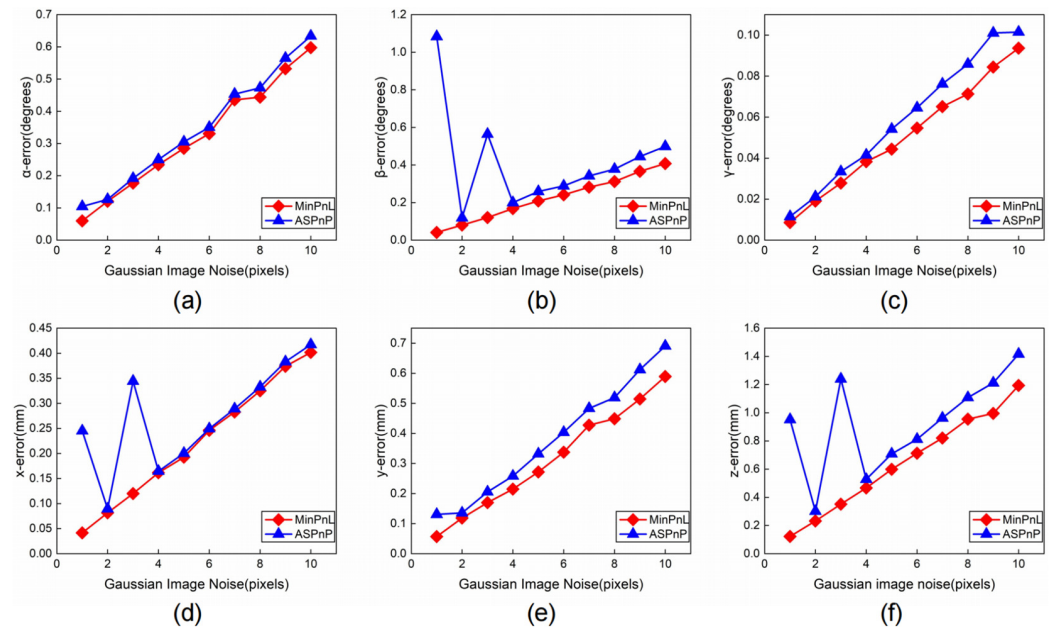


Figure 15. Accuracies of the MinPnL algorithm and the ASPnP algorithm when noise level changes: (a) The rotation error in roll (α) direction; (b) The rotation error in pitch (β) direction; (c) The rotation error in pitch (γ) direction; (d) The translation error in X-axis direction; (e) The translation error in Y-axis direction; (f) The translation error in Z-axis direction.

5. Experimental Results

5.1. Line Detection Experiment

During construction, the high-pressure water gun will be used to wash the disc cutter and its holder which will be replaced. However, the disc cutter holder cannot be completely

rinsed off, and there might still be clay left on it to obscure the line features. In order to simulate the state of clay obscuring the disc cutter holder, we use soil to randomly block the internal contour of the knife holder in our experiments. This method is easier and can simulate worse real conditions. Figure 16 shows a comparison of the line detection results between the classical CannyLines and EDLines methods and our method. The first column is the original image, the second column is the result of CannyLines, the third column is the processing result of EDLines, and the fourth column is the result of our method. The second and third columns show that the traditional CannyLines and EDLines methods cannot extract the complete straight lines in our problem, and bring a large number of unrelated straight lines, which will greatly affect the accuracy and efficiency of the detection. In contrast, our method extracts all complete straight lines under no pollution and light pollution conditions, and even under heavy pollution, our method still accurately extracts a larger number of complete straight lines. The last three rows show that our method can also achieve good extraction results in various poses.

5.2. Accuracy Verification Experiment

To verify the accuracy of the pose measurement method we proposed, we use a laser tracker to calibrate the accuracy. Due to the heavy weight of the disc cutter holder, it is impossible to change its position and attitude. Therefore, we installed the vision measurement system at the end of the Yaskawa robot to change the three-dimensional position and attitude of the camera. The laser tracker we use is Leica AT901-MR (Unterentfelden, Switzerland), the uncertainty of which in the range of 1.5 m is 0.024 mm ($0.015 + 0.006$ mm/m). The robot we use is Motoman-gp7. Our camera is MER-1220-9GM/C (Beijing, China), the resolution of which is 4024×3036 , and the pixel size is $1.85 \mu\text{m} \times 1.85 \mu\text{m}$. The focal length of the lens is 8 mm. Before the experiment, the intrinsic matrix of the camera has been calibrated by the checkerboard.

As shown in Figure 17, the relationship between the camera coordinate system O_C and the laser tracker coordinate system O_L can be expressed by a homogeneous transformation matrix ${}^{laser}_{cam}M$ as:

$${}^{laser}_{cam}M = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \quad (14)$$

Then, we have:

$$P_i^{laser} = R \cdot P_i^{cam} + T \quad (15)$$

where P_i^{cam} is the three-dimensional coordinate of the spherical prism center in the camera coordinate system, and P_i^{laser} is the three-dimensional coordinate of the spherical prism center in the laser tracker coordinate system. Since ${}^{laser}_{cam}M$ has twelve entries, at least four points are required to fully determine the system.

As shown in Figure 18, due to the large and bulky size of the disc cutter holder, it is impossible to perform a six-degree-of-freedom transformation, so a camera-assisted motion of the robot arm is used to achieve the relative positional transformation. The transformation matrix ${}^{cam}_{object}M_i$ of the coordinate system of the disc cutter holder to the camera coordinate system can be easily measured by the vision measurement method described in the previous section. Since the positions of the disc cutter holder and the laser tracker are not changed during the experiment, the transformation matrix of the disc cutter holder to the laser tracker coordinate system is the same for different camera positions. Due to the fact that there is no suitable feature on the surface of the disc cutter holder to directly measure the conversion matrix, it is impossible to obtain the absolute pose measurement error, so we have to measure the error of the relative pose change of the disc cutter holder as the measurement result by using the invariance of the conversion matrix ${}^{laser}_{object}M$ of the camera in two positions. We can express ${}^{laser}_{object}M$ as:

$${}^{laser}_{object}M = {}^{cam}_{object}M_1 \cdot {}^{laser}_{cam}M_1 = {}^{cam}_{object}M_2 \cdot {}^{laser}_{cam}M_2 \quad (16)$$

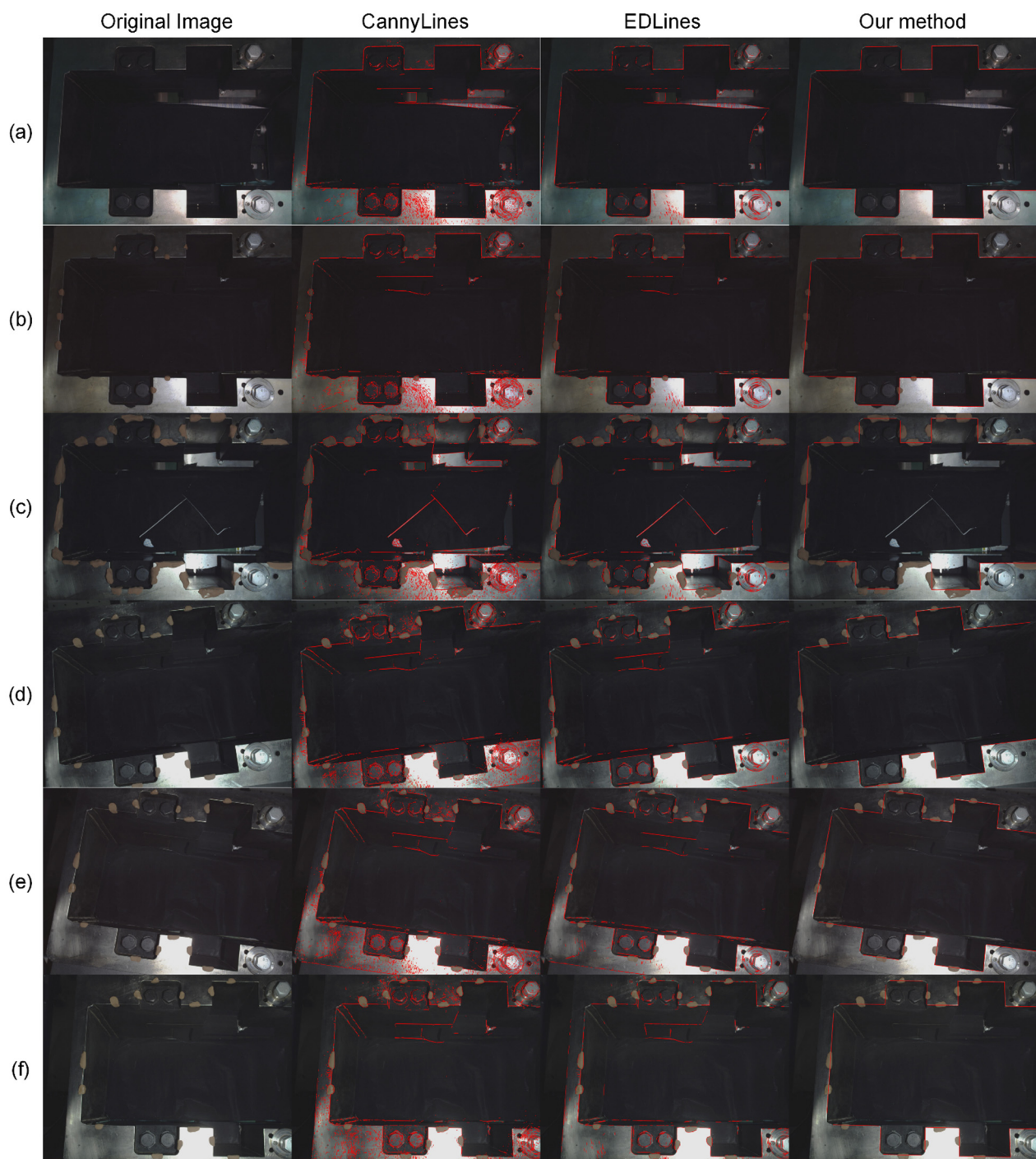


Figure 16. Results of line detection experiment under different conditions: (a) No pollution; (b) Light pollution; (c) Heavy pollution; (d–f) Different poses.

Then, the theoretical transformation matrix ${}^{cam}_{object}M_2'$ in the second position of the camera can be calculated based on Equation (16) as:

$${}^{cam}_{object}M_2' = {}^{cam}_{object}M_1 \cdot {}^{laser}_{cam}M_1 \cdot {}^{laser}_{cam}M_2^{-1} \quad (17)$$

The difference in the corresponding pose related to ${}^{cam}_{object}M_2$ and ${}^{cam}_{object}M_2'$ is the relative pose error.

Table 7 summarizes the entire process of the accuracy verification experiment.

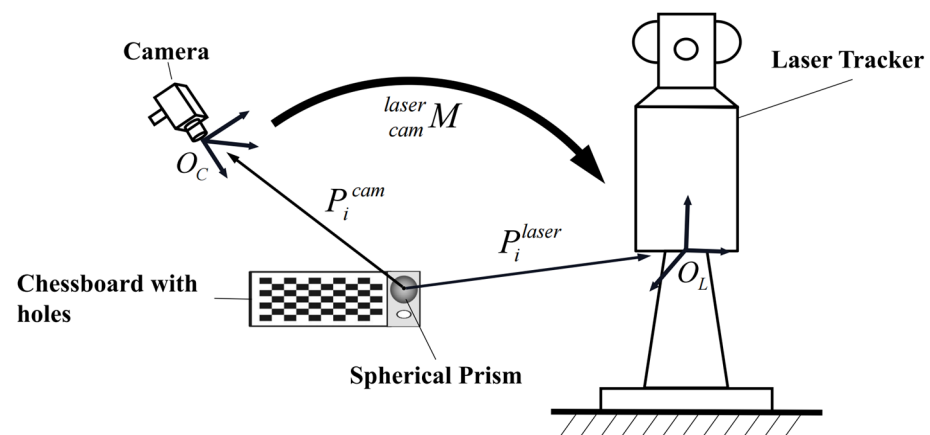


Figure 17. The schematic diagram of coordinate system conversion.

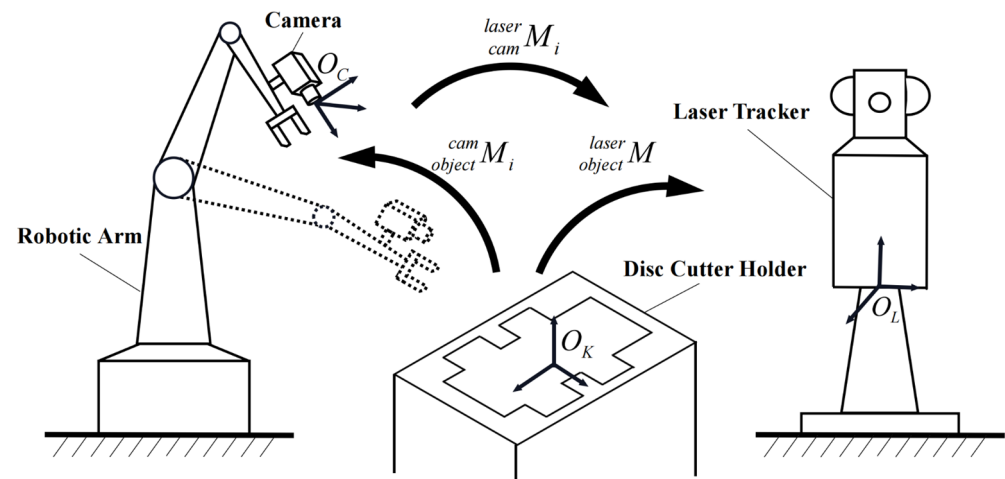


Figure 18. The schematic diagram of relative pose measurement.

Table 7. Steps of the accuracy verification experiment.

Step1:	Complete the calibration of the intrinsic matrix of the camera before the experiment.
Step2:	Randomly move the robot equipped with the vision measurement system to a certain pose within the pose range to be measured.
Step3:	Place the target ball on the hole of the calibration plate at six different positions within the field of view. Measure the coordinates of the target ball center in the laser tracker coordinate system and the coordinates of the calibration plate in the camera coordinate system, respectively.
Step4:	Calculate the conversion matrix ${}^{laser}_{cam}M_1$ between the camera coordinate system and the laser tracker coordinate system in the measurement system. Calculate the transformation matrix ${}^{cam}_{object}M_1$ between the disc cutter holder coordinate system and the camera coordinate system.
Step5:	Repeat the step1—step4 to obtain the coordinate system transformation matrix ${}^{laser}_{cam}M_2$ and ${}^{cam}_{object}M_2$ under the second pose of the robot.
Step6:	Calculate the theoretical transformation matrix ${}^{cam}_{object}M_2'$ between the disc cutter holder coordinate system and the camera coordinate system under the second pose of the robot, and calculate the difference between the poses $pose2$ and $pose2'$ corresponding to ${}^{cam}_{object}M_2$ and ${}^{cam}_{object}M_2'$ as the measurement error.

Figure 19 shows the self-built setup. The estimation error of the rotation matrix R is shown in Figure 20a. The rotation matrix R can be defined by three angles: α (roll), β (pitch), and γ (yaw), which are the rotation angles around the Z, X, and Y axes, respectively.

The α -error is between -0.6° and 0.6° , the β -error is between -0.3° and 0.3° , and the γ -error is between -0.1° and 0.1° . The translation error is shown in Figure 20b, the x-error is between -0.6 mm~ 0.9 mm, the y-error is between -0.7 mm~ 0.6 mm, and the z-error is between -0.8 mm~ 0.7 mm. Overall our measurement method has an attitude measurement accuracy of 0.6° and a displacement measurement accuracy of 0.9 mm, which meets the accuracy requirement.

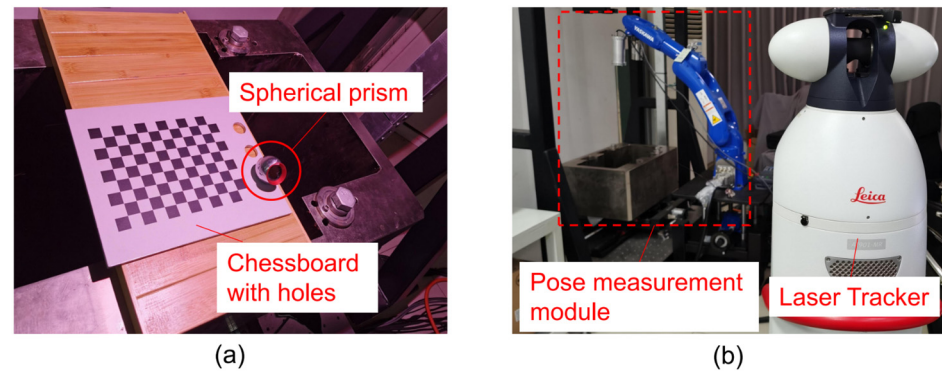


Figure 19. The setup of the accuracy verification experiment: (a) Spherical prism and checkerboard; (b) Laser tracker and pose measurement module.

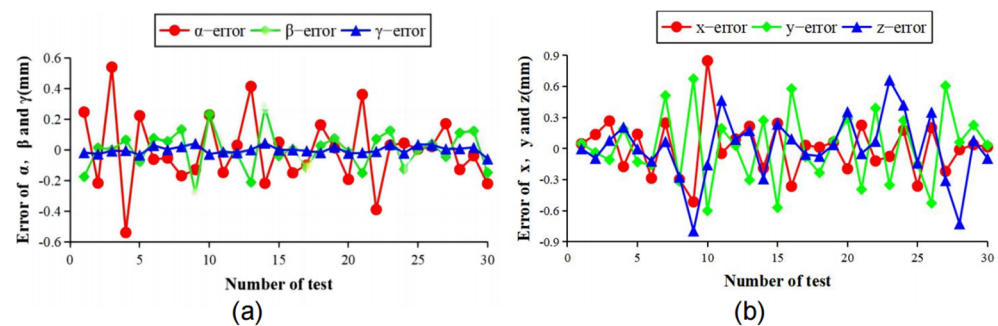


Figure 20. Accuracy verification experiment results: (a) Error of α , β , and γ ; (b) Error of x, y, and z.

In general, for the roof edge, the highest gray value is generally taken as the edge point, while for the ramp edge, the midpoint of the two endpoints of the ramp is generally taken as the edge point. For the results collected from the accuracy verification experiments, we conduct the pose calculation by conventional edge processing and by our proposed rounded edge model, respectively. As shown in Table 8, the result indicates that the absolute value of the maximum error of the results with conventional edge processing is greater than that with rounded edge model processing in all six directions of freedom, and the runtime of our method is shorter. Furthermore, the results of conventional edge processing cannot meet our accuracy requirements, which proves that our rounded edge processing is necessary.

Table 8. Maximum of the absolute errors of each direction of freedom and runtime.

Method	α -Error ($^\circ$)	β -Error ($^\circ$)	γ -Error ($^\circ$)	x-Error (mm)	y-Error (mm)	z-Error (mm)	Runtime (s)
Our proposed method	0.336	0.291	0.032	0.711	0.704	0.882	1.309
Conventional processing	0.925	0.848	0.230	1.735	1.328	0.976	1.858

6. Conclusions

In this paper, we propose a monocular vision method based on line features to estimate the pose of the disc cutter holder. After taking pictures of the disc cutter holder with a CCD

camera, the region growing method with adaptive growth threshold is used to extract the contour under uneven illumination, and then through edge point classification and edge connection, the corresponding edge search box is generated. For the blurring of borders caused by rounded corners, the accurate edge points on each edge are obtained by the rounded edge model and the edge search box. After obtaining all edge points of each edge, the RANSAC algorithm with preprocessing is used for line fitting, where the results show that the preprocessing can significantly improve the accuracy and efficiency. For the pose calculation method, the simulation results show that the MinPnL algorithm is more suitable for our problem. Accuracy verification experiments based on the laser tracker with relative positional errors show that the pose estimation method proposed in this paper can meet the required positioning accuracy and our rounded edge model can better extract the edge points that match the physical model.

The results indicate that our proposed method has good stability and reliability, and can solve the problem of visual positioning of disc cutter holders under contaminated conditions, which helps to realize industrial automation. Our method can also be used in other cases of positioning workpieces containing rectangular contours. For accurate extraction of linear features with rounded edges, our proposed rounded edge model combined with edge search boxes will also be helpful. The measurement accuracy can be further improved in the positional calculation algorithm, including the improvement of the vanishing point-based positional calculation method and the improvement of the PnL algorithm. Developing positional measurement methods for multi-rectangular combinations of targets is our future research focus.

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

Funding: This work was funded by the National Key Research and Development Plan of China, grant number 2018YFB1702504.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

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

References

1. Li, L.P.; Sun, S.Q.; Wang, J.; Song, S.G.; Fang, Z.D.; Zhang, M.G. Development of compound EPB shield model test system for studying the water intrushes in karst regions. *Tunn. Undergr. Space Technol.* **2020**, *101*, 103404. [\[CrossRef\]](#)
2. Shen, X.; Chen, X.; Fu, Y.; Cao, C.; Yuan, D.; Li, X.; Xiao, Y. Prediction and analysis of slurry shield TBM disc cutter wear and its application in cutter change time. *Wear* **2022**, *498*, 204314. [\[CrossRef\]](#)
3. Wang, Z.; Liu, C.; Jiang, Y.; Dong, L.; Wang, S. Study on Wear Prediction of Shield Disc Cutter in Hard Rock and Its Application. *KSCE J. Civ. Eng.* **2022**, *26*, 1439–1450. [\[CrossRef\]](#)
4. Copur, H.; Cinar, M.; Okten, G.; Bilgin, N. A case study on the methane explosion in the excavation chamber of an EPB-TBM and lessons learnt including some recent accidents. *Tunn. Undergr. Space Technol.* **2012**, *27*, 159–167. [\[CrossRef\]](#)
5. Du, L.; Yuan, J.; Bao, S.; Guan, R.; Wan, W. Robotic replacement for disc cutters in tunnel boring machines. *Autom. Constr.* **2022**, *140*, 104369. [\[CrossRef\]](#)
6. Yuan, J.; Guan, R.; Guo, D.; Lai, J.; Du, L. Discussion on the Robotic Approach of Disc Cutter Replacement for Shield Machine. In Proceedings of the IEEE International Conference on Real-time Computing and Robotics (IEEE-RCAR), Electr Network, Hokkaido, Japan, 28–29 September 2020; pp. 204–209.
7. Jiang, J.; Wu, F.; Zhang, P.; Wang, F.; Yang, Y. Pose Estimation of Automatic Battery-Replacement System Based on ORB and Improved Keypoints Matching Method. *Appl. Sci.* **2019**, *9*, 237. [\[CrossRef\]](#)
8. Zhang, Z.M.; Zhang, S.H.; Li, Q. Robust and Accurate Vision-Based Pose Estimation Algorithm Based on Four Coplanar Feature Points. *Sensors* **2016**, *16*, 2173. [\[CrossRef\]](#)
9. Zhou, K.; Huang, X.; Li, S.G.; Li, H.Y.; Kong, S.J. 6-D pose estimation method for large gear structure assembly using monocular vision. *Measurement* **2021**, *183*, 109854. [\[CrossRef\]](#)

10. Teng, X.; Yu, Q.; Luo, J.; Wang, G.; Zhang, X. Aircraft Pose Estimation Based on Geometry Structure Features and Line Correspondences. *Sensors* **2019**, *19*, 2165. [\[CrossRef\]](#)
11. Zhang, Y.Q.; Li, X.; Liu, H.B.; Shang, Y.; Yu, Q.F. Pose optimization based on integral of the distance between line segments. *Sci. China-Tech. Sci.* **2016**, *59*, 135–148. [\[CrossRef\]](#)
12. Von Gioi, R.G.; Jakubowicz, J.; Morel, J.M.; Randall, G. LSD: A Fast Line Segment Detector with a False Detection Control. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 722–732. [\[CrossRef\]](#)
13. Cho, N.G.; Yuille, A.; Lee, S.W. A Novel Linelet-Based Representation for Line Segment Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 1195–1208. [\[CrossRef\]](#)
14. Duda, R.O.; Hart, P.E. Use of Hough Transformation to Detect Lines and Curves in Pictures. *Commun. ACM* **1972**, *15*, 11–15. [\[CrossRef\]](#)
15. Canny, J. A computational approach to edge detection. In *Readings in Computer Vision*; Fischler, M.A., Firschein, O., Eds.; Morgan Kaufmann: San Francisco, CA, USA, 1987; pp. 184–203.
16. Qu, Z.G.; Wang, P.; Gao, Y.H.; Wang, P.; Shen, Z.K. Fast SUSAN edge detector by adapting step-size. *Optik* **2013**, *124*, 747–750. [\[CrossRef\]](#)
17. Wan, S.Y.; Higgins, W.E. Symmetric region growing. *IEEE Trans. Image Process.* **2003**, *12*, 1007–1015.
18. Jma, B.; Cg, C.; Jk, C.J.C.V.; Understanding, I. Robust Detection of Lines Using the Progressive Probabilistic Hough Transform—ScienceDirect. *Comput. Vis.* **2000**, *78*, 119–137.
19. Akinlar, C.; Topal, C. EDLines: A real-time line segment detector with a false detection control. *Pattern Recognit. Lett.* **2011**, *32*, 1633–1642. [\[CrossRef\]](#)
20. Lu, X.H.; Yao, J.; Li, K.; Li, L. Cannylines: A Parameter-Free Line Segment Detector. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Quebec City, QC, Canada, 27–30 September 2015; pp. 507–511.
21. Fischler, M.A.; Bolles, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography—ScienceDirect. *Read. Comput. Vis.* **1987**, 726–740. [\[CrossRef\]](#)
22. Zheng, Y.Q.; Sugimoto, S.; Okutomi, M. ASPnP: An Accurate and Scalable Solution to the Perspective-n-Point Problem. *IEICE Trans. Inf. Syst.* **2013**, *E96D*, 1525–1535. [\[CrossRef\]](#)
23. Zheng, Y.Q.; Kuang, Y.B.; Sugimoto, S.; Astrom, K.; Okutomi, M. Revisiting the PnP Problem: A Fast, General and Optimal Solution. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Sydney, Australia, 1–8 December 2013; pp. 2344–2351.
24. Wang, P.; Xu, G.L.; Cheng, Y.H.; Yu, Q.D. A simple, robust and fast method for the perspective-n-point Problem. *Pattern Recognit. Lett.* **2018**, *108*, 31–37. [\[CrossRef\]](#)
25. Gong, X.R.; Lv, Y.W.; Xu, X.P.; Wang, Y.X.; Li, M.D. Pose Estimation of Omnidirectional Camera with Improved EPnP Algorithm. *Sensors* **2021**, *21*, 4008. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Wang, P.; Chou, Y.X.; An, A.M.; Xu, G.L. Solving the PnP problem using the hidden variable method: An accurate and efficient solution. *Vis. Comput.* **2022**, *38*, 95–106. [\[CrossRef\]](#)
27. Zhou, L.P.; Koppel, D.; Kaess, M. A Complete, Accurate and Efficient Solution for the Perspective-N-Line Problem. *IEEE Robot. Autom. Lett.* **2021**, *6*, 699–706. [\[CrossRef\]](#)
28. Pribyl, B.; Zemcik, P.; Cadik, M. Absolute pose estimation from line correspondences using direct linear transformation. *Comput. Vis. Image Underst.* **2017**, *161*, 130–144. [\[CrossRef\]](#)
29. Peng, D.D.; Zhu, G.L.; Xie, Z.; Liu, R.; Zhang, D.L. An Improved Monocular-Vision-Based Method for the Pose Measurement of the Disc Cutter Holder of Shield Machine. In Proceedings of the 14th International Conference on Intelligent Robotics and Applications (ICIRA), Yantai, China, 22–25 October 2021; pp. 678–687.
30. Peng, D.D.; Zhu, G.L.; Zhang, D.L.; Xie, Z.; Liu, R.; Hu, J.L.; Liu, Y. Pose Determination of the Disc Cutter Holder of Shield Machine Based on Monocular Vision. *Sensors* **2022**, *22*, 467. [\[CrossRef\]](#)

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.