International Journal of Advanced Engineering and Technology Research
ISSN: 3106-8677 (Print), ISSN: 3106-8685 (Online) | Vol. 1, No. 1, 2026

Point cloud method for identifying the center pose of the pin shaft end face in the automated dismantling of oil derricks

Qitao Tu, Chong Xie

School of Mechanical and Electrical Engineering, Southwest Petroleum University, Chengdu 610500, China

Abstract: The derrick of oil drilling rigs is constructed by pin shafts connecting steel frames. Traditional manual disassembly methods pose safety risks and exhibit low efficiency. To achieve automated disassembly, precise identification of the end-face center poses (position and orientation) of pin shafts is essential. However, conventional registration and localization methods suffer from insufficient positioning accuracy due to incomplete point cloud acquisition of pin shafts, while existing deep learning approaches struggle to accurately calculate local object poses. To solve the above problem, this paper proposes a method that combines deep learning and traditional point cloud processing for the precise positioning of the center of the pin end face on the derrick of an oil rig. Specifically, a point cloud segmentation network based on PointNet++ is used in the deep learning part for effectively extracting the point cloud data of the pin shaft and its surrounding structures from the complex scene. In the point cloud processing part, an innovative method based on RANSAC plane fitting is proposed, in which a local coordinate system is established for the fitted plane, the attitude information is calculated, and the plane is sliced for extracting the center of the pin endface to realize the spatial localization of the pin endface.The experimental results show that the positioning error of the proposed method along the X/Y/Z axes is less than 0.5 mm, the directional error is less than 1°, and the accuracy is as high as 96%, which is better than that of the traditional method in terms of precision and accuracy. This study provides a high-precision pin shaft localization solution for automated derrick disassembly, effectively mitigating safety risks and enhancing operational efficiency.

Keywords: PointNet++; RANSAC algorithm; Normal vector coordinate system; Point cloud segmentation slicing.

1. Introduction

In the energy development industry, oil drilling rigs play a pivotal role in petroleum extraction. The derrick, a critical component of oil drilling rigs, requires repeated disassembly and reassembly for reuse, as illustrated in Figure 1. Current derrick disassembly operations predominantly rely on manual removal of pin shafts connecting ear plates, a process associated with significant safety risks and low efficiency. To achieve automated disassembly of pin shafts, precise localization of their end-face centers is imperative. Consequently, research on automated pose estimation of pin shafts holds substantial significance for advancing intelligent disassembly technologies.

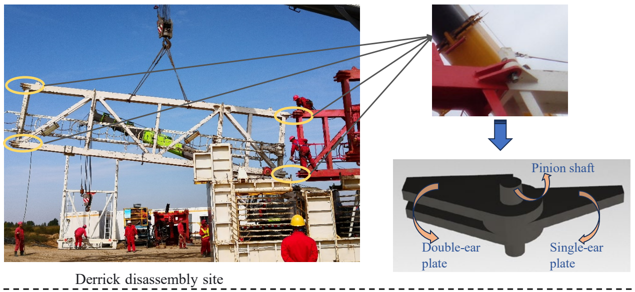


Figure 1. Derrick pin schematic

The research on pin axis positioning involves the determination of six positional information: X, Y, Z, RX, RY, and RZ. 6D object pose estimation is a fundamental problem in the field of computer vision. Traditional methods include feature matching and feature fitting [1] to calculate the pose. Feature matching first models the identified object, and uses the object point cloud and model point cloud to perform feature registration based on feature sub [2][3], ultimately obtaining the actual pose of the object. However, during the dismantling of the wellhead shaft, the structured light camera can only capture one side point cloud of the shaft, and the end face of the shaft may deform during use, resulting in differences between the established model and the actual object, making it difficult to accurately identify the pose of the shaft. In feature fitting, RANSAC [4] is usually used to fit the features for calculating the 6D pose of the object. However, the pin axis may be incomplete in different ways during use, making it impossible to fit the features.

With the rapid development of deep learning, methods have emerged to estimate the 6D pose of objects using RGB images [5][6][7] and point cloud information [8][9][10][11] . When using deep learning for pose estimation, there may be a choice to use optimizers [12][13]. Nugroho et al. [14] conducted experimental analysis on various optimization techniques and pointed out that the optimizer using Adadelta performs the best in both methods. Hoque et al.[15] proposed a new algorithm combining CNN and RNN, which optimizes the filtering of stationary or parked vehicles by introducing long short-term memory, and more effectively predicts the 6D pose of vehicles. Le et al.[16] proposed combining deep neural networks to recognize 2D objects, and then accurately and quickly estimates the pose of objects based on a point to point feature framework. Wu et al.[17] designed a pipeline to obtain complete bolt information and calculate its pose. Govi et al.[18] introduces a new convolutional neural network that uses AAE and LessAAE to handle reflection texture and symmetry issues, and utilizes a single image to extract hierarchical features for object pose recognition. Zhang et al.[19] fused RGB, normal map, and point cloud features to complete missing object information, and tested it on a public dataset with high pose accuracy. Zhuang et al. [20] established a method for generating center seeds through voting after clustering, which estimates the pose of objects in the absence of quaternions. G2 L-Net trained a network similar to PointNet, first extracting features from the point cloud, and then decoupling the pose of objects into three different localization methods [21]. Han et al.[22] constructed a network based on Transformer for optimizing RGB images and feature extraction of point clouds, and designed a network for pose estimation of objects [23]. Periyasamy et al.[24] proposed an improved version of YOLOPose based on keypoint regression and Transformer for estimating the 6D pose of objects. Although deep learning methods have made significant progress in 6D pose estimation, existing methods mainly focus on the pose of the entire object, rather than precise localization of non fixed points.

To address the aforementioned challenges, this article proposes a method that combines PointNet++ and point cloud segmentation slicing to identify the center pose of the pin end face. By using a structured light camera to obtain point cloud data, the PointNet++ model is trained to extract the point cloud of the pin axis and ear plate. The RANSAC algorithm fits the ear plate plane and establishes a coordinate system to calculate the ear plate posture. After removing the planar point cloud, set a threshold to segment the pin axis point cloud, evenly slice the pin axis point cloud, and count the number of point clouds. Select the slice with the highest number of point clouds as the target point cloud, calculate its center coordinates to determine the center position of the pin axis end face. The workflow is shown in Figure 2.

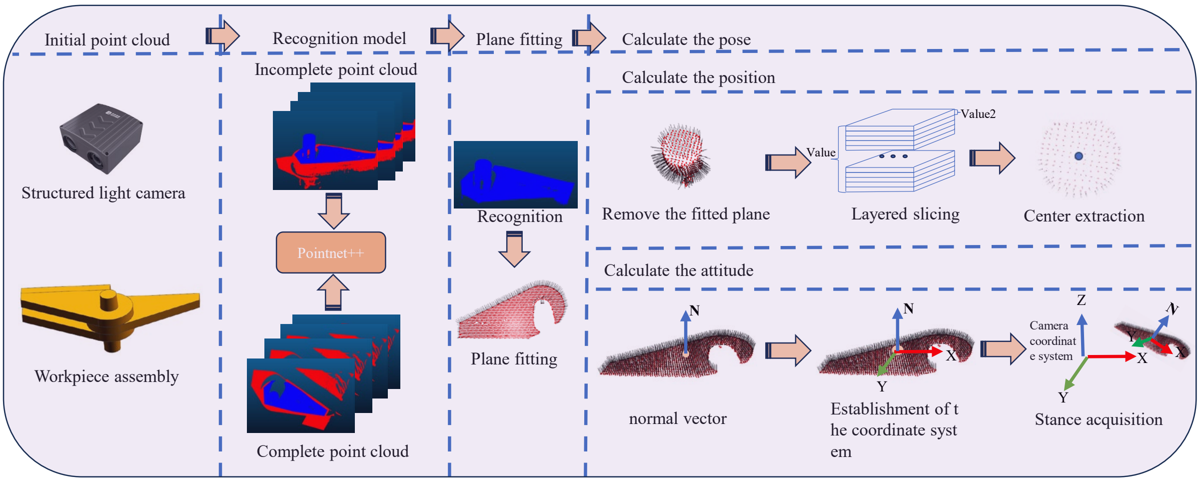


Figure 2. Methodology flow chart

2. Principles and Methods

2.1. Extracting point clouds based on PointNet++

When acquiring raw point cloud data of a target scene using a structured light camera, the captured scene point cloud inevitably contains environmental elements beyond the pin shaft and lug plate components. Consequently, this raw point cloud cannot be directly used for pin shaft localization. To extract the specific point clouds of lug plates and pin shafts from the original data, this paper employs deep learning techniques for point cloud processing.

PointNet++, as an enhanced version of the PointNet network architecture, is specifically designed for 3D point cloud processing. The network employs hierarchical feature extraction combined with multi-scale learning and multi-resolution analysis to process point cloud data. This architecture enables effective feature learning through progressive spatial abstraction while maintaining critical geometric details through local region sampling and grouping operations.

1. Hierarchical feature extraction

This paper utilizes the Farthest Point Sampling (FPS) algorithm to select uniformly distributed subsets from the raw point cloud as hierarchical centroids, ensuring comprehensive coverage of local regions. Centered at these sampled points, local neighborhoods are constructed through either k-Nearest Neighbors (k-NN) or Ball Query radius searches, forming localized point clusters. Within each local region, a PointNet sub-network extracts geometric features that are subsequently aggregated via Max Pooling operations. For classification and segmentation tasks requiring feature propagation across all points, PointNet++ employs interpolation methods such as inverse distance weighting to upsample high-level features while preserving spatial consistency during the feature propagation phase.

2. Multi-scale learning and multi-resolution learning

PointNet++ addresses point cloud density variations through its innovative Multi-Scale Grouping (MSG) and Multi-Resolution Grouping (MRG) strategies. The MSG method performs feature extraction using concentric neighborhoods of varying radii scales, enabling the model to learn local features across multiple spatial granularities while enhancing adaptability to non-uniform density distributions. Specifically, this multi-radius approach allows simultaneous capture of fine-grained geometric details and coarse structural patterns. Complementarily, the MRG mechanism implements cross-layer feature interpolation through hierarchical resolution fusion, where features from different network depths are intelligently combined using learnable weights.

This paper collects and labels multiple sets of scene point cloud data containing pin shafts and lug plates, and uses the PointNet++ network model to extract features from and train for the segmentation of these specific mechanical components, enabling the network model to accurately extract the point clouds of pin shafts and lug plates (data1).

2.2. Fitting Plane and Compute Pose

Following the successful extraction of pin shaft and lug plate point clouds from the previous step, plane fitting must be performed on the processed point cloud data (data1) to enable precise segmentation of the pin shaft point cloud and calculation of its spatial orientation. The RANSAC (Random Sample Consensus) algorithm serves as a robust mathematical tool for parameter estimation of geometric models. To implement RANSAC-based plane fitting on the target point cloud data1, we first define the general plane equation:

$$Ax + By + Cz + D = 0 \tag{1}$$

where (x, y, z) represents the coordinates of points in the point cloud, A, B, C constitute the components of the planes normal vector N = (A, B, C), and D denotes the planar offset.

The RANSAC algorithm calculates plane parameters by randomly sampling three points $p_1(x_1,y_1,z_1)$, $p_2(x_2,y_2,z_2)$, and $p_3(x_3,y_3,z_3)$. As shown in Figure 3, the iterative optimization process using the RANSAC algorithm follows these steps.

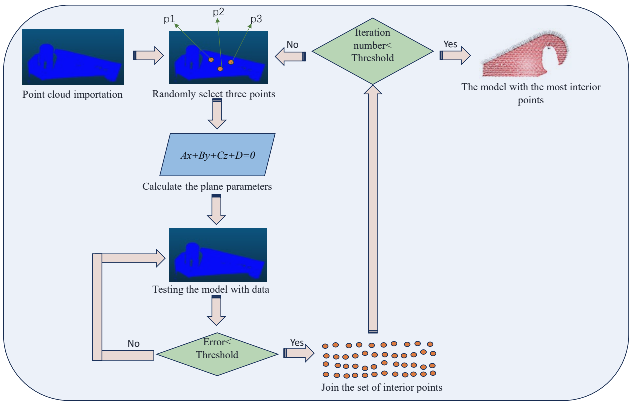


Figure 3. RANSAC fitting plane process

Due to the rigid connection between the pin axis and the ear plate, their postures are consistent. After plane fitting the ear plate, the ear plate plane equation can be obtained, which includes the normal vector of the ear plate plane. In the camera coordinate system, the coordinate system is established based on the normal vector of the ear plate plane, and the Z-axis of the new coordinate system is constructed with the plane parameter normal vector N=(A, B, C) as the Z-axis:

Normalized Normal Vector:

$$\boldsymbol{Z'} = \frac{\boldsymbol{n_C}}{\|\boldsymbol{n_C}\|} \tag{2}$$

Select the camera coordinate system with X-axis v=(1,0,0) as the reference vector and calculate the cross product:

$$\boldsymbol{X'} = \frac{\boldsymbol{v} \times \boldsymbol{z'}}{\|\boldsymbol{v} \times \boldsymbol{z'}\|} \tag{3}$$

Calculate Y-axis through cross product:

$$\boldsymbol{Y'} = \boldsymbol{Z'} \times \boldsymbol{X'} \tag{4}$$

Using X , Y , Z as column vectors, construct a rotation matrix R for the new coordinate system:

$$\boldsymbol{R'} = [\boldsymbol{X'}, \boldsymbol{Y'}, \boldsymbol{Z'}] \tag{5}$$

Write it in matrix form:

$$\boldsymbol{R'} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{6}$$

Convert the rotation matrix into Euler angles again:

$$\alpha = \operatorname{atan} 2(r_{21}, r_{11}) \tag{7}$$

$$\beta = \operatorname{atan} 2\left(r_1^2, \sqrt{r_{11}^2 + r_{21}^2}\right) \tag{8}$$

$$\gamma = \operatorname{atan} 2(r_{32}, r_{33}) \tag{9}$$

2.3. Divide and slice to calculate the center position

Assuming that the scene point cloud data (data1) contains three-dimensional coordinate data of M points, the final number of inliers obtained during the RANSAC fitting process of the plane is N, which are extracted from the scene point cloud data (data1). The inliers are removed from the scene point cloud data (data1);

$$Q = M - N \tag{10}$$

The remaining point cloud data (data2) obtained after final exclusion contains three-dimensional coordinate data of Q points. In this point cloud data, there will still be residual points outside the ear plate plane. Therefore, based on the residual point cloud data (data2) obtained above, a value (plane distance value) is set to the plane equation:

$$Ax + By + Cz + D = value \tag{11}$$

$$a < value < b \tag{12}$$

Among them, a and b are a range value set for value, which is the distance between the top of the pin and the ear plate. The data2 contains three different coordinate values. Assuming there are Q points, the different coordinate values of each point in data2 are named (X(i), Y(i), Z(i)), where i=1, 2, …, m. Substitute the coordinate values of each point in data2 into equation (11):

$$AX(i) + BY(i) + CZ(i) + D = value \tag{13}$$

Save the point records that satisfy equation (13) to obtain a new dataset (data3), which extracts the data belonging to the sales axis point cloud. Process the value again, divide it into n equal parts, and obtain the range value of each value2:

$$a + \frac{b-a}{n} * (k-1) < value2 < \frac{b-a}{n} * k \tag{14}$$

Where k is a positive integer, k must satisfy the following conditions:

$$k \le n \tag{15}$$

By substituting all value2 range values into equation (11), n plane slice equations with uniform thickness are obtained. Based on point cloud data3, all points in point cloud data3 are verified and compared with each plane slice equation to find the plane slice equation that each point conforms to. The corresponding points contained in each plane slice equation are statistically analyzed, resulting in n new datasets (data4). The generated n new datasets (data4) are hierarchical segmentation point cloud data of the sales axis point cloud.

Select the set data5 with the most data after slicing from date4, which is the center position of the pin end face. Assuming there are q points in the set data5, name the different coordinate values of each point in the data set date5 as (x (j), y (j), z (j)), where j=1, 2, …, q. Take the average of each coordinate value:

$$x(c) = \frac{1}{q}\sum_{j=1}^{q} x(j) \tag{16}$$

$$y(c) = \frac{1}{q}\sum_{j=1}^{q} y(j) \tag{17}$$

$$z(c) = \frac{1}{q}\sum_{j=1}^{q} z(j) \tag{18}$$

Through the above processing, the center position of the pin end face can be obtained as (x (c), y (c), z (c)), and the principle process is shown in Figure 4.

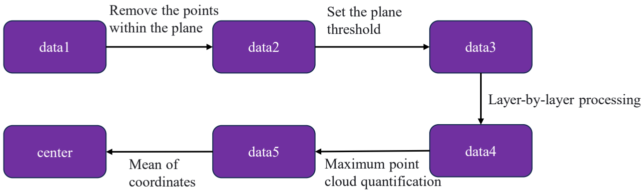
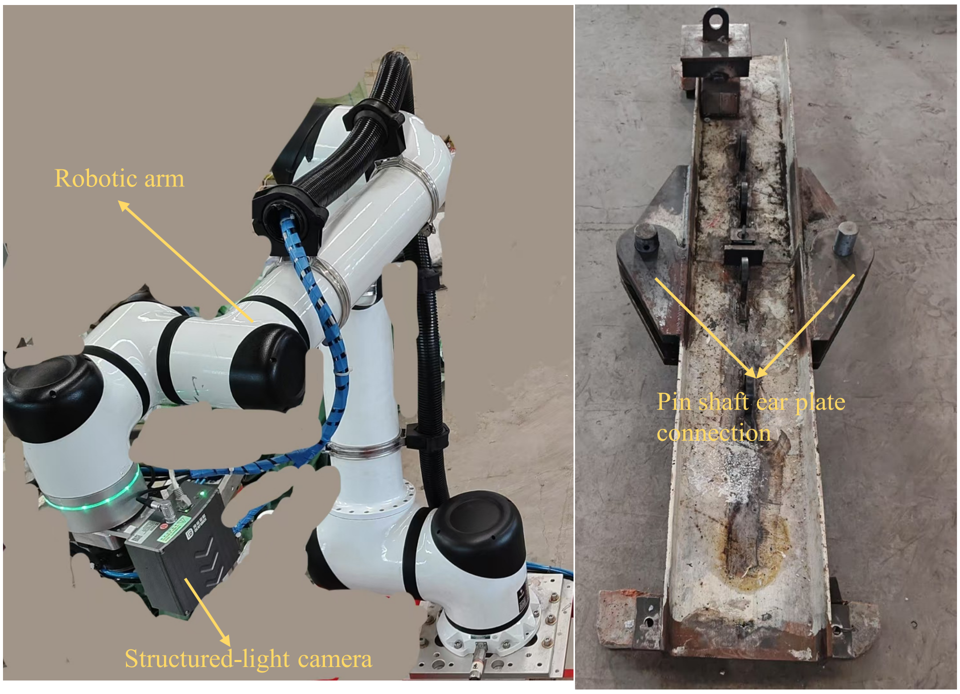


Figure 4. Segment and slice to find the center

3. Experiment and analysis

3.1. Experimental parameter configuration

In order to verify the pin axis positioning method described in this article, we constructed an experimental platform that includes several key components: S30 family robot, Xema cross dimensional structured light camera, and derrick pin axis experimental model, as shown in Figure 5.



(a) Execution system **(b)** Derrick model

Figure 5. Experimental system

3.2. PointNet++ Network Model Training and Validation

The training process of PointNet++ is shown in Figure 6. Firstly, a structured light camera was used to collect point clouds. 100 sets of scene point clouds containing pin ear plates captured from different angles were collected, which were divided into two types: complete and partially incomplete point clouds. The obtained scene point clouds were labeled, and the pin ear plate point clouds that needed to be recognized were boxed and labeled as 0. All scene point clouds were boxed as 1. The processed two sets of point clouds were merged to obtain a point cloud for point labeling classification. All processed datasets were trained using the PointNet++ point cloud network model to obtain a point cloud semantic segmentation network model. The detailed parameters during training are shown in Table 1.

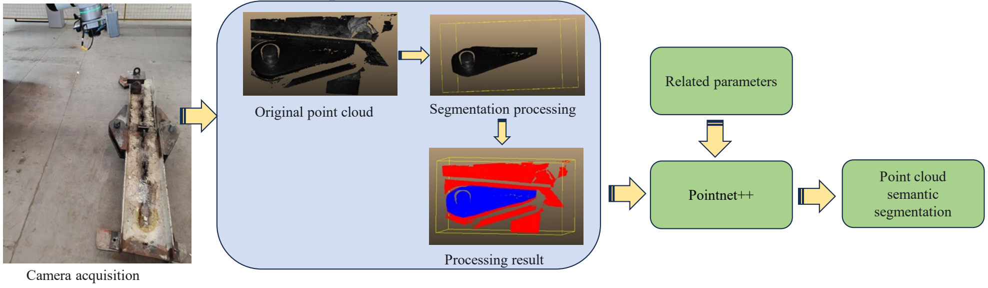


Figure 6. PointNet++ processing flow

Table 1. Model parameter setting

Parameter type	Parameter
Model name	PointNet++
Training data	80
Test data	20
Number of samples per batch	4
Total number of training times	250
Initial learning rate	0.001
Learning rate decay step	20
Optimizer type	Adam

After the sample training is completed, the trained recognition model is tested, and the detailed information is shown in Table 2.

Table 2. Accuracy table

Evaluation name	Data (%)
Mean Intersection over Union	95.82
Accuracy	97.88
Class avg accuracy	97.89
Class avg mIOU	95.82

From the analysis of the accuracy of the test, the accuracy value reached 97.88%, and the Miou value was as high as

95.82%. Therefore, using PointNet++ to preliminarily extract the point cloud data of the pin axis ear plate has extremely high accuracy. Figure 7 shows that the complete and incomplete pin axis point cloud maps obtained from different poses can be accurately recognized and segmented by the model.

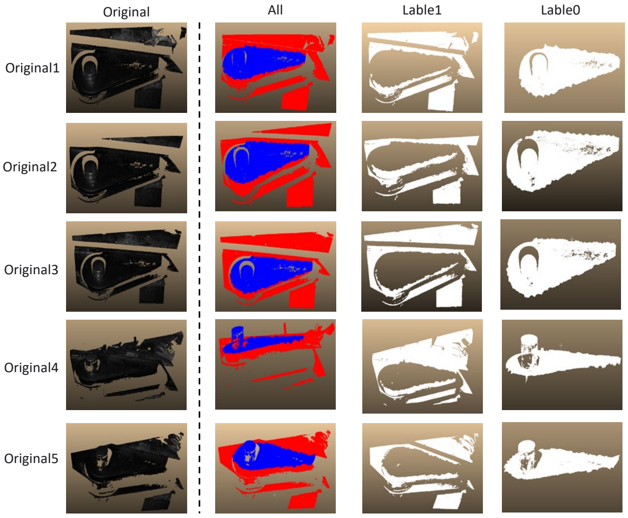


Figure 7. Point cloud extraction of pin lugs

3.3. Plane fitting and new coordinate system attitude experiment

To ensure the robustness and reasonableness of the plane fitting, we provide theoretical support and experimental validation of the key parameters of the RANSAC algorithm, the number of iterations and the interior point distance threshold.

RANSAC requires at least N iterations to successfully obtain a model containing full interior points with probability p. The theoretical formulation is as follows:

$$N = \frac{\log(1-p)}{\log(1-(1-\varepsilon)^s)} \tag{19}$$

Where p denotes the confidence level of at least one success, which is taken as 0.99 in this study, and ϵ is the proportion of outlier points, and after pre-processing statistics on multiple point clouds, the proportion of outlier points in the scene is estimated to be about 30% (ϵ=0.3), and the minimum number of samples for planar fitting is 3 (s=3), which can be obtained by substituting into the above equation:

$$N = \frac{\log(1-0.99)}{\log(1-(1-0.3)^3)} \approx 10.96 \tag{20}$$

To enhance the robustness of the algorithm in the presence of noise interference or outlier points, the maximum number of iterations is set to 50 in this paper. This setting is higher than the average convergence requirement, which is in line with the parameter specification of mainstream point cloud processing frameworks, and at the same time has a limited impact on the computational efficiency.

The inner point distance threshold affects the point set range of the fitting plane. Considering that the single-point measurement error of the structured light camera is about 0.5 mm, the initial threshold is set to 1 mm, and its reasonableness is verified by ablation experiments.

In order to select the optimal interior point distance threshold, different interior point distance thresholds are used after noise reduction and downsampling of the point cloud based on deep learning segmentation of the same attraction cloud, and each threshold is repeated for 20 experiments before calculating its related parameters.

Table 3. Comparison of different thresholds

Threshold (mm)	*Average number of in-points*	*RMSE (mm)*	*RMSE standard deviation*	*Normal vector fluctuation (°)*
0.5	*462*	*0.146*	*0.046*	*2.23*
1	*553*	*0.299*	*0.038*	*0.18*
2	*640*	*0.587*	*0.085*	*0.21*
5	*707*	*1.181*	*0.21*	*0.36*

As shown in Table 3, although the number of interior points rises with the increase of the distance threshold, the planar fitting error (RMSE) and normal vector fluctuation increase significantly. Among the various indicators, the fitting effect is most stable when the threshold value is 1 mm, the RMSE and normal vector fluctuation are kept at a low level, and the number of interior points is also representative, which is selected as the best threshold parameter after comprehensive consideration.

To explore the extent to which the stability of the structured light camera affects the results of the measurements, the point cloud data were repeatedly acquired 10 times for the same scene, and after a series of processing, a plane fit was performed on the 10 sets of data based on the 50 iterations chosen above with an interior point distance threshold of 1 mm.

Table 4. Multi-component data testing

Threshold (mm)	Average number of in-points	RMSE (mm)	RMSE standard deviation	Normal vector fluctuation (°)
1	566	0.268	0.021	0.032

According to the changes in the standard deviation of RMSE and normal vector fluctuations in Table 4, it can be seen that the system exhibits highly consistent estimation results, with very small changes in the normal vectors and very low fluctuations in the fitting error, indicating that the structured light camera used possesses good stability and repeatability in this measurement scenario, and is able to efficiently support the precise 3D geometry extraction and attitude estimation tasks.

To verify the completeness of the constructed coordinate system, taking the plane normal vector in the camera coordinate system as N=(0,0,1) as an example, a coordinate system was constructed and the attitude angles were calculated as roll=0 °, pitch=0 °, and yaw=-90 °. The schematic diagram of the camera coordinate system and the constructed coordinate system positions is shown in Figure 8.

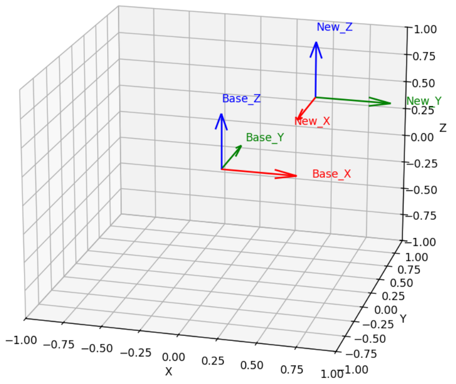


Figure 8. Camera coordinate system and physical coordinate system

To verify the accuracy of the pose calculation of the new coordinate system in the camera coordinate system, corresponding experiments were designed for validation. Due to the difficulty in controlling the stable angle of the pin ear plate, a robotic arm is used to control the rotation of the camera. The real-time attitude of the camera can be obtained through the controller of the robot arm by calibrating the camera with the robot arm by hand and eye. To control the variables, only the *X*-axis of the camera is rotated to evaluate the attitude error. The actual attitude angle and calculated attitude angle are shown in Table 5.

Table 5. Different actual and calculated postures

Rotate X-axis (°)	Posture name	True attitude (°)	Calculate attitude (°)
0	Posture1	(0, 0.00, -90)	(0, 0.57, -90)
10	Posture2	(0, -10.00, -90)	(0, -9.84, -90)
20	Posture3	(0, -20.00, -90)	(0, -19.89, -90)
30	Posture4	(0, -30.00, -90)	(0, -30.11, -90)
40	Posture5	(0, -40.00, -90)	(0, -39.56, -90)

In Table 5, only the *X*-axis is rotated, so only the corresponding attitude angle changes. The maximum error of the attitude angle is 0.57 °, and the overall error is small, meeting the accuracy requirements for disassembling the positioning pin axis.

3.4. Segmentation slice centering experiment

Based on the calculated plane, remove the points in the point cloud that conform to the plane obtained after down sampling, and then segment the pin axis. Use the plane to set a slice thickness value for slicing the pin axis, select the slice with the highest number of point clouds, and calculate the center position coordinates of the slice.

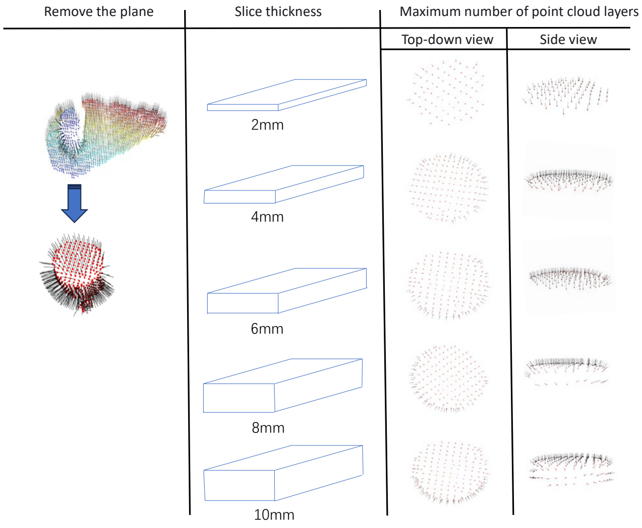


Figure 9. Segmentation slice extraction center

Table 6. Slicing center values of point clouds with different thicknesses

Slice thickness(mm)	*X-value(mm)*	*Y-value(mm)*	*Z-value(mm)*
Actual	*-103.50*	*-36.50*	*487.50*
2	*-103.39*	*-35.08*	*485.66*
4	*-103.05*	*-36.90*	*487.48*
6	*-103.10*	*-36.53*	*487.29*
8	*-102.84*	*-35.34*	*486.88*
10	*-102.80*	*-33.75*	*486.64*

In Figure 9, the point cloud of the pin axis was sliced using slices with thicknesses of 2mm, 4mm, 6mm, 8mm, and 10mm. At 2mm, the slice thickness was too thin, resulting in a missing point cloud at the top of the pin axis. At 4mm and 6mm slice thicknesses, the point cloud at the top of the pin axis was intact and had fewer interfering points around it. However, at 8mm and 10mm slice thicknesses, although the point cloud at the top of the pin axis was intact, it also cut out the point cloud on the side of the pin axis, resulting in interference. Due to the above reasons, in the center position coordinate data of Table 6, the *X*, *Y*, and *Z* coordinates of 4mm and 6mm are more stable than those of 2mm, 8mm, and 10mm, with a difference of 0.05mm for *X*, 0.37mm for *Y*, and 0.19mm for *Z*. The error between the center values of 4mm and 6mm slices and the actual values is shown in Table 7.

Table 7. Point cloud slice center error value

Slice thickness(mm)	ΔX*(mm)*	ΔY*(mm)*	ΔZ*(mm)*
4	*0.45*	*0.4*	*0.02*
6	*0.4*	*0.03*	*0.41*

In the range of 4-6mm, the maximum error of *X* is 0.45mm, the maximum error of *Y* is 0.40mm, and the maximum error of *Z* is 0.41mm. Therefore, it can be concluded that the optimal selection of slice thickness is within the range of 4mm-6mm.

3.5. Pose combination experiment

After collecting the point clouds for the pins at different positions, the deep learning is used to segment the appropriate point clouds, and then the normal vector coordinate system is constructed and the center value of the point cloud slicing layer is calculated to get the pose of the pin in the camera coordinate system, which is then converted to the base coordinate system of the robotic arm. The final pose results are shown in Table 8.

Table 8. Comparison of different pin positions with actual values

Point	*Actual pose(mm,°)*	*Calculate the pose(mm,°)*
Point1	*(-485,-1250,-800,-180,0,-90)*	*(-485.61,-1249.82,-800.30,-179.80,0.46,-90.5)*
Point2	*(-510,-1300,-800,-180,0,-90)*	*(-510.24,-1300.44,-800.40,-180.33,0.25,-90.30)*
Point3	*(-550,-1280,-800,-180,0,-90)*	*(-549.71,-1280.36,-800.20,-180.21,0.33,-89.91)*
Point4	*(-600,-1330,-800,-180,0,-90)*	*(-600.31,-1330.44,-799.82,-179.64,0.38,-90.43)*
Point5	*(-630,-1400,-800,-180,0,-90)*	*(-630.42,-1399.62,-799.73,-179.91,0.18,-90.36)*

Using mean square error (MSE) and mean absolute error (MAE) as evaluation metrics, the estimation errors of attitude angles (*Rx*, *Ry*, *Rz*) and positions (*X*-axis, *Y*-axis, *Z*-axis directions) were calculated separately.

Table 9. The MSE and MAE

Pose	*MSE*	*MAE*
X estimation	*0.157*	*0.374mm*
Y estimation	*0.139*	*0.36mm*
Z estimation	*0.079*	*0.27mm*
Rx estimation	*0.066*	*0.238°*
Ry estimation	*0.112*	*0.32°*
Rz estimation	*0.133*	*0.336°*

The data in Table 9 shows that the *Z* position estimation has the highest accuracy (MSE=0.079, MAE=0.27 mm), while the error of the Rx attitude angle is the smallest (MSE=0.066, MAE=0.238°). In contrast, the errors in *X*/*Y* position and *Rz* attitude angle are relatively large, especially with Rzs MAE reaching 0.336 °, indicating a higher challenge in estimating rotational degrees of freedom. Overall, the positioning error is limited to within 0.5 mm, and the attitude error is controlled within 1 °. These differences include the accuracy error of the shooting distance of the structured light camera, the error caused by external lighting on the formation of point clouds, and the error of coordinate transformation. Although there are errors, the estimation error of the pin axis pose is relatively small, and the method stability is high, which meets the positioning accuracy for disassembling the pin axis.

3.6. Comparison of other methods used

Due to the limitations of deep learning methods in accurate pose estimation of pin axes, this paper adopts the traditional feature-based registration method [25] for experimental comparison. To ensure the fairness of the experimental results, all experiments were conducted under the same conditions, including the configuration of the structured light camera and the external lighting environment.

In order to verify the stability and success rate of this papers method to determine the pins position, this paper acquired the field point cloud of the pin ear plate 50 times at different angles and distances, and only one side of the pins point cloud could be acquired when shooting. Different methods are utilized to verify that when the pin is disassembled, the clearance between the disassembly tool and the pin is 1.5 mm, and the tolerable angle is within 1.5°, so the error of the three-axis coordinates is within 1.5 mm, and the error of the attitude is within 1.5°, which is defined as the success of obtaining the pins positional attitude.

Table 10. Different methods and success rates

Experimental method	Number of experiments	Number of successes	success rate
FPFH+ICP	50	28	56%
FPFH+RANSAC	50	31	62%
OURS	50	48	96%

When the traditional algorithm is used, mismatches and inaccuracies occur due to missing pin point clouds, as shown in Figure 10.

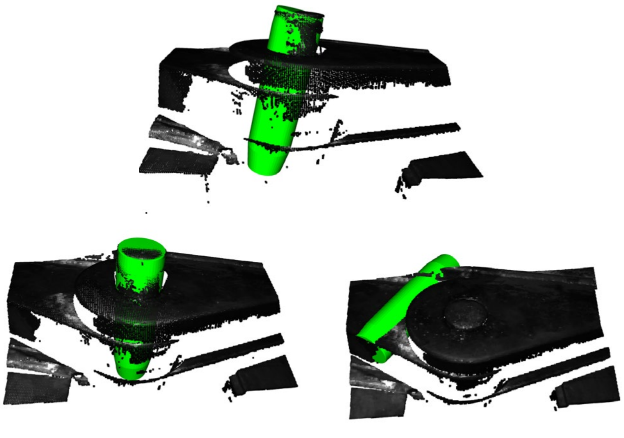


Figure 10. Point cloud alignment error

As shown in Table 10, the conventional method exhibits relatively low accuracy when applied to high-precision pin positioning. This performance degradation can be attributed to two main factors:

1. The pin has undergone repeated hammering, causing structural deviations from the ideal CAD model;

2. The structured-light camera captures point clouds from only one side of the pin, and that surface is often contaminated with oil stains, resulting in incomplete point cloud data and degraded point-to-model matching performance.

In contrast, the method proposed in this paper demonstrates significant advantages by directly localizing the pin based on its point cloud, without relying on prior models. It achieves a positioning accuracy of up to 96%, and yields notable improvements in precision, making it particularly suitable for high-precision pin localization in drilling rig derrick environments.

4. Conclusion

In summary, this paper proposes a pin end face center localization method that combines deep learning and traditional point cloud processing for complex industrial scenarios such as oil rig derricks. This method adopts the point cloud segmentation network based on PointNet++ to realize the efficient extraction of the pin and its surrounding structures; meanwhile, a point cloud processing strategy based on RANSAC plane fitting is proposed to accurately locate the center of the pin endface by establishing a local coordinate system to obtain the plane attitude and slicing and analyzing the fitted plane.The main contributions are as follows:

Firstly, when using structured light cameras to capture point clouds of targets, there are point clouds of other components present. The Pointnet++ network model was used to train the collected dataset, and the trained Miou value reached 95.82%, which can effectively extract point clouds of the pin axis and ear plate.

Secondly, based on RANSAC for plane fitting, a coordinate system is established using plane normal vectors. Through experiments, the maximum error of attitude angle is obtained to be 0.57 °. The extracted point cloud is segmented and sliced to find the center position, and the most accurate center position is obtained when the slice thickness is between 4mm-6mm.

Finally, the method of calculating position and attitude was combined, and the pin axis coordinates were converted to the mechanical arm base coordinate system. Experiments show that the overall error of positional localization is kept within 0.5 mm, while the attitude error is controlled within 1 °, with a success rate of 96 percent.Compared with traditional registration positioning methods, this paper has the smallest error and the highest stability, making it more suitable for the positioning of derrick pin axes and providing a precise positioning module for automated disassembly of pin axes.

However, this method still has the following limitations:

1. The incomplete end face of the pin affects the positioning accuracy: When the end face of the pin is severely damaged, the slice thickness needs to be adjusted to ensure the extraction of suitable end face point clouds.

2. A large amount of butter coverage leads to the loss of point cloud on the sales axis: all the butter adheres to the surface of the sales axis, affecting the point cloud acquisition of the structured light camera, resulting in complete loss of point cloud information on the sales axis.

Future research will focus on the extraction method of end face normal vector features, improve the stability of pin axis center pose estimation, and consider using high-precision radar instead of structured light cameras to enhance adaptability to complex environments (such as butter coverage), thereby improving the accuracy of position information acquisition between the pin axis and ear plate.

References

[1] Ji, X., Zhang, X., & Hu, H. (2021). Point cloud segmentation for complex microsurfaces based on feature line fitting. Multimedia Tools and Applications, 80, 4553-4578.

[2] Rusu, R. B., Blodow, N., & Beetz, M. (2009, May). Fast point feature histograms (FPFH) for 3D registration. In 2009 IEEE international conference on robotics and automation (pp. 3212-3217). IEEE.

[3] Zhong, J., Dujovny, M., Park, H. K., Perez, E., Perlin, A. R., & Diaz, F. G. (2003). Advances in ICP monitoring techniques. Neurological research, 25(4), 339-350.

[4] Chum, O., Matas, J., & Kittler, J. (2003, September). Locally optimized RANSAC. In Joint pattern recognition symposium (pp. 236-243). Berlin, Heidelberg: Springer Berlin Heidelberg.

[5] Varia, N., Dokania, A., & Senthilnath, J. (2018, November). DeepExt: A convolution neural network for road extraction using RGB images captured by UAV. In 2018 IEEE Symposium Series on Computational Intelligence (SSCI) (pp. 1890-1895). IEEE.

[6] Liu, Z., Shi, S., Duan, Q., Zhang, W., & Zhao, P. (2019). Salient object detection for RGB-D image by single stream recurrent convolution neural network. Neurocomputing, 363, 46-57.

[7] Hu, S., Hou, R., Ming, L., Meifang, S., & Chen, P. (2023). A hyperspectral image reconstruction algorithm based on RGB image using multi-scale atrous residual convolution network. Frontiers in Marine Science, 9, 1006452.

[8] Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (2017). Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 652-660).

[9] Ge, L., Ren, Z., & Yuan, J. (2018). Point-to-point regression pointnet for 3d hand pose estimation. In Proceedings of the European conference on computer vision (ECCV) (pp. 475-491).

[10] Aoki, Y., Goforth, H., Srivatsan, R. A., & Lucey, S. (2019). Pointnetlk: Robust & efficient point cloud registration using pointnet. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 7163-7172).

[11] Qi, C. R., Yi, L., Su, H., & Guibas, L. J. (2017). Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems, 30.

[12] He, S., Wu, Q. H., & Saunders, J. R. (2006). A group search optimizer for neural network training. In Computational Science and Its Applications-ICCSA 2006: International Conference, Glasgow, UK, May 8-11, 2006, Proceedings, Part III 6 (pp. 934-943). Springer Berlin Heidelberg.

[13] Bello, I., Zoph, B., Vasudevan, V., & Le, Q. V. (2017, July). Neural optimizer search with reinforcement learning. In International Conference on Machine Learning (pp. 459-468). PMLR.

[14] Nugroho, B., Suciati, N., & Fatichah, C. (2024). Analysis of Optimization Techniques in 6D Pose Estimation Approaches using RGB Images on Multiple Objects with Occlusion. Procedia Computer Science, 234, 1689-1696.

[15] Hoque, S., Xu, S., Maiti, A., Wei, Y., & Arafat, M. Y. (2023). Deep learning for 6D pose estimation of objects—A case study for autonomous driving. Expert Systems with Applications, 223, 119838.

[16] Le, T. T., Le, T. S., Chen, Y. R., Vidal, J., & Lin, C. Y. (2021). 6D pose estimation with combined deep learning and 3D vision techniques for a fast and accurate object gras**. Robotics and Autonomous Systems, 141, 103775.

[17] Wu, C., Fu, H., Kaiser, J. P., Barczak, E. T., Pfrommer, J., Lanza, G., Heizmann, M., & Beyerer, J. (2024). 6d pose estimation on point cloud data through prior knowledge integration: A case study in autonomous disassembly. Procedia CIRP, 122, 193-198.

[18] Govi, E., Sapienza, D., Toscani, S., Cotti, I., Franchini, G., & Bertogna, M. (2024). Addressing challenges in industrial pick and place: A deep learning-based 6 Degrees-of-Freedom pose estimation solution. Computers in Industry, 161, 104130.

[19] Zhang, H. B., Hong, J. X., Liu, J. H., Lei, Q., & Du, J. X. (2025). Images, normal maps and point clouds fusion decoder for 6D pose estimation. Information Fusion, 117, 102907.

[20] Zhuang, C., Wang, H., & Ding, H. (2024). AttentionVote: A coarse-to-fine voting network of anchor-free 6D pose estimation on point cloud for robotic bin-picking application. Robotics and Computer-Integrated Manufacturing, 86, 102671.

[21] Chen, W., Jia, X., Chang, H. J., Duan, J., & Leonardis, A. (2020). G2l-net: Global to local network for real-time 6d pose estimation with embedding vector features. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 4233-4242).

[22] Han, K., **ao, A., Wu, E., Guo, J., Xu, C., & Wang, Y. (2021). Transformer in transformer. Advances in neural information processing systems, 34, 15908-15919.

[23] Hong, J. X., Zhang, H. B., Liu, J. H., Lei, Q., Yang, L. J., & Du, J. X. (2024). A transformer-based multi-modal fusion network for 6d pose estimation. Information Fusion, 105, 102227.

[24] Periyasamy, A. S., Amini, A., Tsaturyan, V., & Behnke, S. (2023). YOLOPose V2: Understanding and improving transformer-based 6D pose estimation. Robotics and Autonomous Systems, 168, 104490.

[25] Pan, J., & Zhang, X. (2025, February). Research on 3D point cloud registration algorithm based on FPFH and ColorICP. In International Conference on Optical and Photonic Engineering (icOPEN 2024) (Vol. 13509, pp. 33-38). SPIE.