



Research Article

A fast registration method for multi-view point clouds with low overlap in robotic measurement

Chuangchuang Li^a, Xubin Lin^b, Zhaoyang Liao^{b,*}, Hongmin Wu^b, Zhihao Xu^b, Xuefeng Zhou^b^a College of Electronic and Information Engineering, Wuyi University, Jiangmen 529020, China^b Institute of Intelligent Manufacturing, Guangdong Academy of Sciences, Guangdong Key Laboratory of Modern Control Technology, Guangzhou 510095, China

ARTICLE INFO

Article history:

Received 26 September 2024

Revised 21 October 2024

Accepted 3 November 2024

Available online 26 November 2024

Keywords:

Point cloud registration

Feature interaction

Multi-view

Robotic measurement

ABSTRACT

With the rapid advancement of mechanical automation and intelligent processing technology, accurately measuring the surfaces of complex parts has emerged as a significant research challenge. Robotic measurement technology plays a crucial role in facilitating rapid quality inspections during the manufacturing process due to its inherent flexibility. However, the irregular shapes and viewpoint occlusions of complex parts complicate precise measurement. To address these challenges, this work proposes a point cloud registration network for robotic scanning systems and introduces a DBR-Net (Dual-line Registration Network) to overcome the issues of low overlap rates and perspective occlusion that currently impede the registration of certain workpieces. First, feature extraction is performed using a bilinear encoder and multi-level feature interactions of both point-wise and global features. Subsequently, the features are sampled through unanimous voting and fed into the RANSAC (Random Sample Consensus) algorithm for pose estimation, enabling multi-view point cloud registration. Experimental results demonstrate that this method significantly outperforms many existing techniques in terms of feature extraction and registration accuracy, thereby enhancing the overall performance of point cloud registration.

© 2024 The Author(s). Published by Elsevier B.V. on behalf of Shandong University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

In contemporary manufacturing and engineering domains, the measurement accuracy of intricate components directly impacts the overall performance and reliability of products. With the rapid advancement of robotic technology, robotic scanning systems have demonstrated significant potential in the measurement of complex parts, owing to their high efficiency, non-contact nature, and high-precision capabilities. However, when confronted with components featuring intricate geometries and varying structures, single-view scanning frequently fails to capture their three-dimensional information comprehensively, thereby constraining measurement accuracy. To mitigate this challenge, multi-view registration technology based on robotic scanning systems has emerged [1–3]. This technology employs a three-dimensional vision sensor mounted on a robot to scan parts from multiple perspectives, thereby acquiring multiple sets of point cloud data. Advanced registration algorithms are subsequently utilized to accurately integrate these data, reconstructing the three-dimensional morphology of the parts. This approach

not only overcomes the limitations of single-view scanning but also substantially enhances the comprehensiveness and accuracy of measurements [4,5]. Nonetheless, multi-view registration of complex parts is not a trivial task. Surface features of parts, noise interference during scanning, and discrepancies in data from different views collectively pose significant challenges to the registration algorithm. Consequently, the development of efficient and robust multi-view registration algorithms is essential for improving the practical effectiveness of robotic scanning systems in the measurement of complex parts. This paper concentrates on multi-view registration technology based on robotic scanning systems and introduces a dual-line multi-view point cloud registration algorithm specifically designed for robotic scanning systems.

Registration algorithms can currently be categorized into several major groups based on different registration principles.

Feature-based methods: These methods estimate the features of two input point clouds using networks and subsequently iteratively perform correspondence and transformation estimation to achieve the final solution. Some researchers focus on leveraging feature descriptors for point cloud registration. The GFOICP [6], proposed by He et al. employs optimized set features as constraints and then applies the ICP (Iterative Closest

* Corresponding author.

E-mail address: zy.liao@giim.ac.cn (Z. Liao).

Point) [7] algorithm for registration, improving both accuracy and speed compared to conventional ICP. The ORB-PC [8], developed by Mohammad et al. introduces an image registration method based on Ridgeline Invariant Feature Transform (RIFT), which has demonstrated favorable outcomes in horizontal and vertical transformations of multimodal images. Lei et al. [9] proposed a template matching method (GFTM) based on global depth features, utilizing deep convolutional networks to extract common global depth features from multimodal images before registration. Pan et al. [10] introduced the OCTRexpert algorithm for feature extraction from retinal images, designing features for each image pixel prior to registration. Sun et al. [11] extracted multiple feature descriptors from SAR images and conducted registration, finding that SURF features are advantageous for multi-temporal SAR image registration. Lee et al. [12] proposed a framework for remote sensing image registration based on deep learning, in which the network is trained on equivariant features rather than invariant features, thereby enhancing registration accuracy.

End-to-end approach: This approach can be divided into three parts: the correspondence prediction module, the weight prediction module, and the singular value decomposition (SVD) module. The main idea is to input the two point clouds to be aligned into a neural network and utilize an end-to-end framework to estimate the transformation matrix between them. Currently, various network frameworks utilize end-to-end registration. DeepVCR [13] is the first point cloud registration framework that employs an end-to-end architecture. It uses PointNet++ to extract the semantic features of each point from the two point clouds and generates keypoints. The accuracy is further improved by applying a loss function that incorporates local similarity and global geometric constraints. CCAG [14], proposed by Wang et al. utilizes a cross-convolutional attention module that combines cross-attention mechanisms with depth-separable convolution to capture the relationships between point clouds and integrate features. Associations between point clouds are established through cross-attention computation, while depth-separable convolution operations extract local features and spatial relationships. Additionally, the CCAG algorithm introduces an adaptive graph convolution MLP, enhancing the expressiveness of the nodes. DCP [15] is a single-step point cloud registration method based on deep learning. It learns features using DGCNN and employs an attention mechanism to process the correspondences and feature information of the point clouds to be aligned, predicting rotations and achieving point cloud matching through the SVD module. Finally, FMR [16] estimates the transformation matrix between two input point clouds by minimizing the projection error in the feature space, ensuring the accuracy of point correspondences even under noise interference.

Unsupervised Methods: These approaches do not necessitate manual annotations or class labels. ATNet [17], inspired by adaptive graph convolution, dynamically learns point cloud features based on adaptive kernels to generate point clouds. FPN learns stable keypoints in the input point cloud, utilizing probabilistic chamfer loss and point-to-point loss to enhance the accuracy of keypoints localization and point cloud matching. R-PointHopR [18] determines local reference frames (LRFs) for points using their nearest neighbors. The LRF allows the hierarchical features of points to remain invariant under significant rotation and translation, thereby establishing reliable point correspondences in space. UPCR [19] models the input point cloud as a combination of pose-invariant and pose-related components, using the pose-related representation to learn the relative pose and subsequently deriving a rigid transformation from the learned relative poses. An unsupervised point cloud registration method based on learning a unified Gaussian mixture model (GMM) [20] aims to reconstruct a unified GMM while addressing

the registration problem. This method designs a new feature interaction module that utilizes both self and cross point cloud information to learn the posterior probabilities. Subsequently, it proposes two differential modules to compute the GMM parameters and the transformation matrix. Existing unsupervised methods often overlook the importance of feature interaction, resulting in ambiguities in feature matching. To address these challenges, a novel unsupervised point cloud registration framework called GTINet [21] is proposed. This framework includes a Global Structural Relation (GSR) module and a Contextual Topological Interaction (CTI) module. The CTI module primarily learns the geometric feature similarity and relative positional knowledge between the source and target point clouds. Additionally, the CTI module further learns contextual feature interaction through a topology-aware attention layer. VK-Net [22] is capable of generating semantically consistent and rotation-invariant keypoints between objects of the same category but different views. It uses VK-Net to construct corresponding relationships from the learned keypoints. These keypoints correspondences are then utilized to compute a good pose initialization, followed by using Iterative Closest Point (ICP) to optimize the registration.

Based on the identified advantages and disadvantages of existing point cloud registration algorithms, this paper introduces a novel framework named DBR-Net for point cloud registration, as illustrated in Fig. 1. Initially, the data undergoes filtering to optimize the point cloud representation. Subsequently, both the source point cloud and the target point cloud are input into the DR-Encoder within the registration framework, which generates translation and rotation features through point feature interaction. These generated features are then processed by the DR-Decoder, which outputs the translation and rotation parameters of the source point cloud via global feature interaction. The registration process is conducted utilizing the RANSAC algorithm, leading to the derivation of the final transformation matrix. The contributions of this paper are summarized as follows:

- This paper proposes a dual-line encoder and decoder designed for translation and rotation to enhance the extraction and interaction of point cloud features.
- The incorporation of hierarchical point feature interaction alongside global feature interaction optimizes the handling of details during feature interaction, thereby facilitating subsequent point cloud registration tasks.
- A comprehensive framework for point cloud registration algorithms is established based on the aforementioned components. Comparisons with existing algorithms demonstrate that this framework provides distinct advantages in terms of both accuracy and time efficiency.

2. Method

Relevant parameters are as follows: F represents the initial features, and H denotes the hybrid features generated through global feature interaction. P and Q denote the input source point cloud and target point cloud, respectively.

2.1. Data processing

Use a radius filter, which is a relatively simple and straightforward approach. This method requires that each point in the point cloud data has a sufficient number of neighboring points within a defined range; if this condition is not met, the point will be deleted. The implementation process is as follows: first, a K-D tree is constructed for the point cloud data to establish topological relationships. Then, the number of neighboring points within the specified radius is calculated for each point in the point cloud, as shown in Fig. 2. If the number of neighboring points is less than

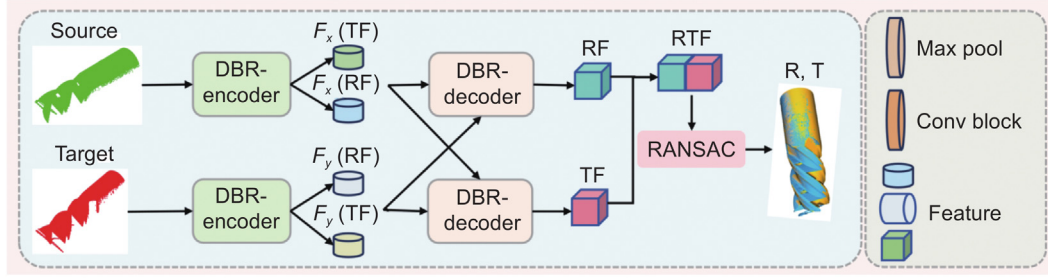


Fig. 1. Flowchart of DBR-Net.

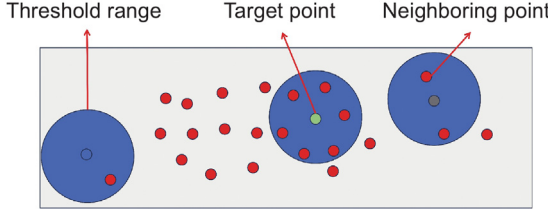


Fig. 2. Radius filter.

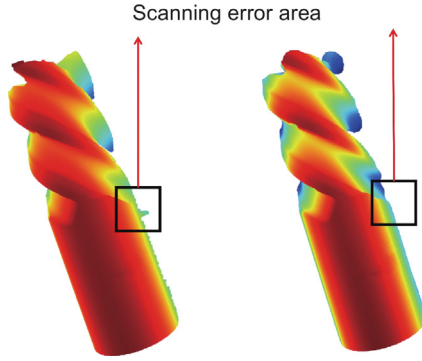


Fig. 3. Left:point cloud without pre-processing,Right:point cloud after radius filtering.

a predetermined threshold, the point is considered a noise point and is removed. This process is repeated until all points have been processed.

Set the threshold to 5. Thus, blue and gray points are to be deleted if the number of neighboring points is less than 5. In contrast, the green point meets the requirement and can be retained. This method effectively eliminates the noise points that lie outside the actual measured data of the spiral cutter in this study.

As shown in Fig. 3, It is evident that the noise points near the cutter have been successfully removed, while the other regions of the point cloud remain largely unaffected.

2.2. Multi-level feature interaction

Hierarchical feature interaction involves partitioning the operation into local feature interaction and global feature interaction. Initially, the raw feature data undergoes point feature interaction to optimize the feature information. Subsequently, the processed data is subjected to global feature interaction to yield the final feature information.

2.2.1. Point feature interaction

In the dual-line encoder, point feature interaction (PFI) is incorporated after multiple convolutional blocks. Initially, the

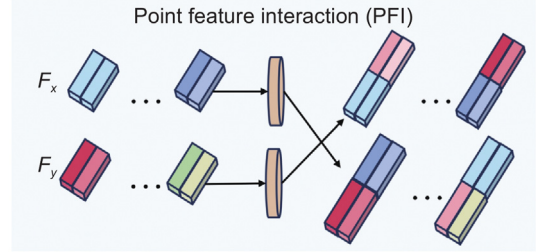


Fig. 4. Point feature interaction.

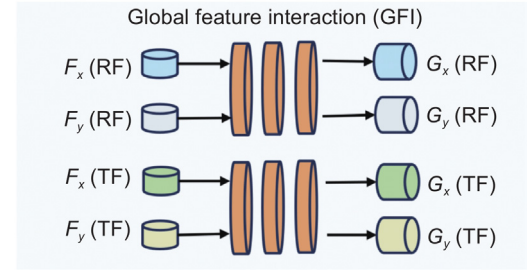


Fig. 5. Global feature interaction.

point-wise features of a point cloud are aggregated through per-channel max pooling, followed by broadcasting and concatenating with the point-wise features of another point cloud at the same level. Subsequently, the encoder features are updated accordingly.

$${}^{(k)}f_x = {}^{(k)}g_{\theta}(\text{cat}[\text{cat}^{(k-1)}f_x, \text{repeat}(\max^{(k-1)}f_y, N_x)]) \quad (1)$$

In the equation, the function ${}^{(k)}g_{\theta}(\cdot)$ represents the K th convolutional block, and $\text{repeat}(z, N)$ indicates that the vector z is repeated N times along the element dimension. When extracting the features of y , the positions of the input are reversed (see Fig. 4).

2.2.2. Global feature interaction

Before applying the regression transformation parameters, the information exchange between inputs is further enhanced through global feature interaction, as illustrated in Fig. 5. The rotation and translation features are concatenated with the MLPs to generate mixed global features. The entire process is defined as follows:

$$H_x^r = h_{\theta}^r(\text{cat}[F_x^r, F_y^r]), H_x^t = h_{\theta}^t(\text{cat}[F_x^t, F_y^t]) \quad (2)$$

Where $h_{\theta}^r(\cdot)$ and $h_{\theta}^t(\cdot)$ denote the GFI functions for rotation and translation, respectively. When generating the features of y , the positions of the inputs are reversed.

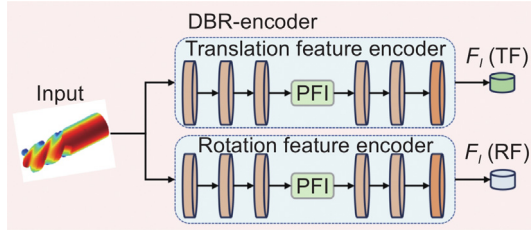


Fig. 6. Dual branch registration encoder.

2.3. Dual branch registration encoder

The shared encoder for both translation and rotation is bifurcated into dual branches dedicated to processing point clouds: one for translation feature extraction and the other for rotation feature extraction, as depicted in Fig. 6. Point feature interaction plays a pivotal role in this feature extraction process. Given that translation pertains to Euclidean space transformations and is not intimately linked to quaternion space transformations, using shared weights in a conventional encoder offers limited advantages, often leading to suboptimal feature interaction. Hence, this paper advocates for the use of a dual-branch encoder to independently extract translation and rotation features. Drawing from [23], point features undergo initial processing via interaction to derive individual point feature outputs from the convolutional blocks of the MLP. Subsequently, multi-level point features are amalgamated and further interacted through max pooling, optimizing the underlying information flow between the source and target point clouds. To bolster the encoder's capabilities, a point-to-point feature interaction module is incorporated. In each iterative step, the source point cloud X is substituted with the point cloud transformed in the preceding iteration. The encoder receives the transformed point cloud X_0 and the target point cloud Y as inputs. The global features are then formulated as follows:

$$F_m^n = \max(\text{cat}[\{f_m^k\}_{k=1 \dots K}]) \quad (3)$$

where $m \in \{x, y\}$, $n \in \{r, t\}$. The superscripts r and t denote the global features F belonging to the rotary encoder and translation encoder, respectively. Here, k represents the point feature f output from the k th convolutional block, with a total of k convolutional blocks. The notation $\max(\cdot)$ signifies channel maximum pooling, while $\text{cat}[\cdot, \cdot]$ denotes crosstalk. The encoder utilizes shared weights across the inputs.

2.4. Dual branch registration decoder

We perform regression processing separately on the translation features and rotation features. Using the global features, we interact the features extracted from both branches and then generate the translation and rotation matrices, as shown in Fig. 7.

Given the input features F^r and F^t , we use a global feature interaction module to generate the mixed features H^r and H^t for rotation and translation, respectively. Consistent with the encoder, we employ a dual-branch regression network to regress the rotation and translation parameters separately. Specifically, the rotation regression branch takes all global features as input and generates a 4D vector that represents the three-dimensional rotation $q^r q = 1$ in quaternion form $q \in \mathfrak{H}^4$. Meanwhile, the translation regression branch does not directly regress the translation vector $t \in \mathfrak{H}^3$, instead, it generates two three-dimensional vectors that represent two significant points from the source

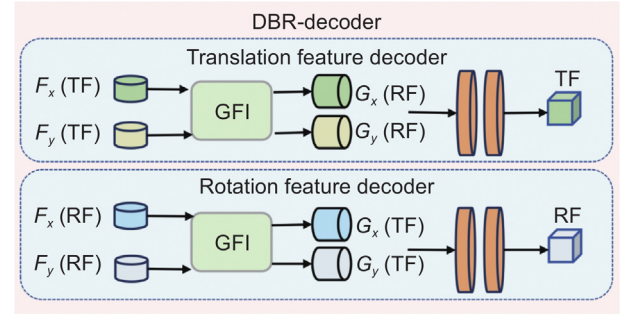


Fig. 7. Dual branch registration decoder.

point cloud and the reference point cloud, calculating the difference between them as t . In each iteration, the transformation is updated to $\{q, t\}$.

$$q = f_\theta^r(\text{cat}[H_x^r, H_x^t, H_y^r, H_y^t]), t = c_y - c_x \quad (4)$$

$$c_x = f_\theta^t(\text{cat}[H_x^r, H_x^t, H_y^t]) \quad (5)$$

$$c_y = f_\theta^t(\text{cat}[H_y^r, H_y^t, H_x^t]) \quad (6)$$

where the functions $f_\theta^r(\cdot)$ and $f_\theta^t(\cdot)$ represent the rotation and translation regression networks, respectively. The vectors $c_x \in \mathfrak{H}^3$ and $c_y \in \mathfrak{H}^3$ denote the coordinates of the significant points.

2.5. Loss function

To measure the similarity or dissimilarity of features between the source point cloud and the target point cloud during the registration process, the registration parameters are optimized by minimizing a loss function in order to achieve more accurate point cloud registration. This loss function guides the optimization process. A key aspect of the network training is that the encoder extracts attention features for rotation and translation, while the decoder uses these extracted features to reconstruct the point cloud. Optionally, the following losses can be implemented.

2.5.1. Reconstruction loss

The purpose of the reconstruction loss is to calculate the difference between the source point cloud and the reconstructed source point cloud and the difference between the target point cloud and the reconstructed target point cloud. As shown in (7) and (8), where P' and Q' denote the reconstructed source point cloud and the reconstructed target point cloud, and P and Q denote the two initial input point clouds. The minimum distance from any point p in P to Q . The second term represents the minimum distance from any point q in Q to P . The two points with the smallest distance in two points cloud are not necessarily corresponding points. Here, another issue to consider is whether to use $\text{loss}_{re} = d_{CD}(P', Q')$ instead. After several experiments, it was found that using this function works better, mainly because both P' and Q' are obtained after reconstructing with their corresponding features, and the feature metric difference is considered in the feature loss.

$$d_{CD}(P, Q) = \frac{1}{P} \sum_{p \in P} \min_{q \in Q} \|p - q\|_2^2 + \frac{1}{Q} \sum_{q \in Q} \min_{p \in P} \|q - p\|_2^2 \quad (7)$$

$$\text{loss}_{re} = d_{CD}(\tilde{P}, P') + d_{CD}(Q, Q') \quad (8)$$

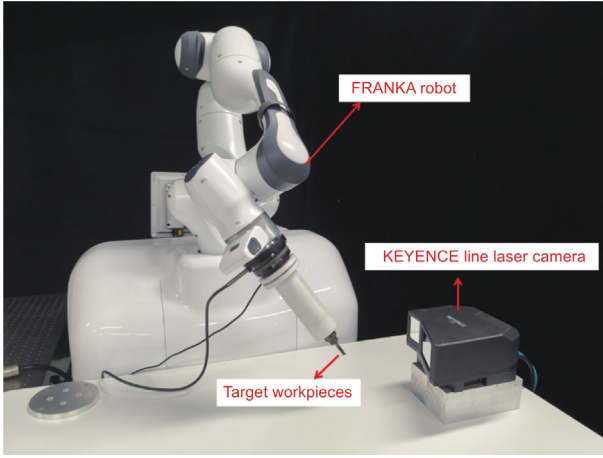


Fig. 8. Canning device.

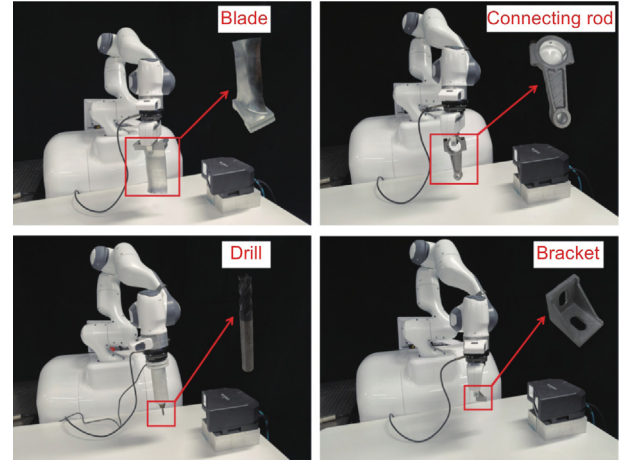


Fig. 9. Various workpiece types.

2.5.2. Feature loss

The purpose of feature loss is to calculate the difference between the features extracted from the source point cloud and those from the target point cloud. The feature loss function is evaluated using the Root Mean Square Error (RMSE) metric, which is computed as follows:

$$loss_{fea} = d_{RMSE}(F_x, F_y) = \sqrt{\frac{1}{N} \sum_{i=1}^N (F_{x'} - F_y)^2} \quad (9)$$

where F_x, F_y the features of the transformed source and target point clouds, and N denotes the dimension of the features, which is set to 1024 in the encoder.

2.5.3. Transformation loss

The purpose of the transform loss is to compute the difference between the estimated transform (R, t) and the actual transform (\bar{R}, \bar{t}) from the source point cloud to the target point cloud. The chord distance between R and \bar{R} is calculated by the Frobenius criterion of the rotation matrix. The angular error can be calculated as $\cos(\theta) = \frac{Tr(R^T \bar{R}) - 1}{2}$. From the rotation matrix R to extract the rotation angle θ , Tr of the rotation matrix. And the translation error is calculated as the Euclidean distance between t and \bar{t} . To facilitate the computation of the loss function, the estimated transformation matrix is set to $T = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$.

The transformation loss function is

$$loss_{gt} = \left\| T - \bar{T} \right\|_F \quad (10)$$

The final loss function for the training process is

$$loss = loss_{st} + loss_{gt} + \lambda_1 loss_{re} + \lambda_2 loss_{fea} \quad (11)$$

where $loss$ denotes the chamfer distance between the transformed source and target point clouds, denoted $loss_{st} = d_{CD}(\bar{P}, Q)$, which is the classical distance metric used to measure the registration effect. For the coefficients in the loss function, the coefficients of the first two are 1, which is a classical setting, and the coefficients of the last two need to be determined according to the situation.

3. Experiments

The experiments in this paper are based on a robotic scanning system for data acquisition, as illustrated in Fig. 8. The system

comprises a FRANKA robot and a KEYENCE line laser rangefinder. The robot holds the workpiece and adjusts its pose within the camera's field of view to capture point cloud data from multiple angles of the measured workpiece. This scanning system can acquire surface data from various complex parts. Combined with the registration algorithm proposed in this paper, it reconstructs the overall model of the target workpiece, as shown in Fig. 9. Based on the algorithm in this paper, the experimental workpiece used is a drill bit. Since drill bits are spiral in shape, the camera cannot capture the entire drill bit in a single scan. This introduces challenges such as low overlap, complex features, and occlusions, making it an ideal candidate for validating the proposed algorithm.

3.1. Algorithm experiment

This paper compares multiple registration methods, including MLMD [24], RC-RANSAC [25], PACNet [26], and the proposed DBR-Net, for point cloud registration of datasets such as Dragon, Armadillo, Happy Buddha, Room1, Room2, Room3, and Drill. Among these, Dragon, Armadillo, and Happy Buddha are point cloud datasets from Stanford University, while Room1, Room2, and Room3 are sourced from the 3DMatch dataset. The Drill dataset consists of point cloud data obtained from our own scans of drill bits.

The training datasets used in this paper are 3DMatch and 3DLoMatch, which are two widely used indoor datasets containing more than 30% and 10%–30% of partially overlapping scene pairs, respectively. We utilized a portion of these datasets for training, specifically 300 samples. The three aforementioned algorithms, along with our proposed algorithm, were trained using these samples. MSE (R) and RMSE (R) represent the squared error and root mean squared error between the true values and the predicted values of the rotational angles, respectively. Similarly, MSE (t) and RMSE (t) indicate the squared error and root mean squared error between the true and predicted values of translation distances. Smaller error values indicate higher registration accuracy.

We then applied the proposed method to the testing point cloud data and observed that it offers advantages in both processing time and registration accuracy. Fig. 10 presents the registration results of each method, while Table 1 compares the processing time, MSE, and RMSE for registration accuracy.

From the figures and tables above, it is evident that the method proposed in this paper achieves significant improvements in both registration accuracy and processing time. However, for point cloud data with a high degree of overlap, the

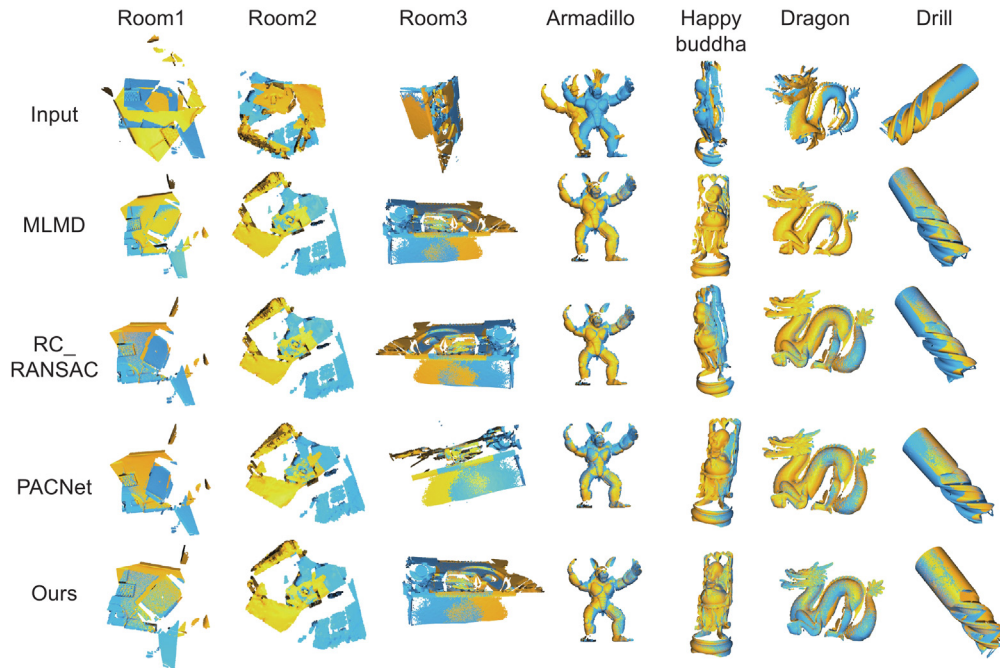


Fig. 10. Registration results of each method.

Table 1
Comparison of mean and standard deviation of time and registration accuracy of different datasets.

Datasets	MLMD			RC_RANSAC			PACNet			Ours		
	MSE (um)	RMSE (um)	t (s)	MSE (um)	RMSE (um)	t (s)	MSE (um)	RMSE (um)	t (s)	MSE (um)	RMSE (um)	t (s)
Dragon	2975	1725	2.31	2478	1578	2.25	2963	1723	2.15	1465	758	2.03
Armadillo	3370	1836	3.81	3444	1856	3.25	3659	1913	1.46	2314	756	1.45
Happy Buddha	6325	2515	3.52	6482	2546	3.12	2023	1425	2.55	1235	786	1.83
Room1	2473	4973	2.53	2481	4981	2.55	4042	2899	2.22	2352	1232	1.81
Room2	3606	1899	2.78	1645	4057	1.64	1630	4038	2.00	1965	2123	1.43
Room3	9391	9691	2.33	7649	8746	3.50	5228	7231	3.65	3242	3456	1.55
Drill	3456	3232	11.85	3484	3123	11.28	3458	2241	10.83	2112	1352	6.22

presence of many similar feature points allows both the proposed method and the comparison registration algorithms to achieve satisfactory registration results. Nonetheless, the proposed method demonstrates greater advantages in terms of processing time.

In order to more clearly observe the advantages of the proposed algorithm compared to other algorithms, we extracted different numbers of feature points from the Drill data and compared the parameters MSE, RMSE, and t , as shown in Fig. 11.

From the comparison of these three parameters, it can be seen that the algorithm proposed in this paper demonstrates certain advantages in terms of MSE (um), RMSE (um), and t (s). Both high and low numbers of feature points can lead to poor registration results, indicating that when performing point cloud registration, it is crucial to set the relevant parameters appropriately. This ensures that registration accuracy is not compromised while also achieving a certain level of efficiency in registration speed.

3.2. Multi-view registration

For the Drill data in this paper, we use the FRANKA robot to hold the workpiece and adjust its pose within the camera scanning area, collecting the relevant data. We enhance the algorithm presented in this paper by incorporating an initial pose adjustment algorithm for point clouds to perform initial registration. After adjusting the pose, the modified point clouds are input

into the DBR-Net algorithm framework for fine registration. This approach effectively addresses the aforementioned issues.

By calculating the motion relationships of each axis of the FRANKA robot, we derive the final motion model of the end-effector camera. When adjusting the camera pose based on the movements of the FRANKA robot, this motion model can be utilized to compute how the captured point cloud needs to undergo spatial matrix transformations. Consequently, we can reposition the scanned point cloud data to an appropriate initial position. Fine registration is then performed using DBR-Net, achieving multi-view point cloud registration.

As shown in Fig. 12, the initial point cloud scanned is depicted. It is evident that the point cloud data is largely overlapping, and the point cloud features are essentially the same. This causes feature-learning registration algorithms to be incapable of solving the problem. Therefore, the motion model is used to calculate and perform an initial pose adjustment on the point cloud data. From b, it can be seen that after the initial rotation and translation, the point cloud data has preliminarily achieved the shape of the drill bit. Subsequently, the DBR-Net algorithm is used for feature extraction and local feature matching to match the features of the overlapping areas between point cloud data. This process achieves the final model registration. As shown in the figure, the results of multiview point cloud registration are achieved, realizing the stitching and registration of the drill bit model.

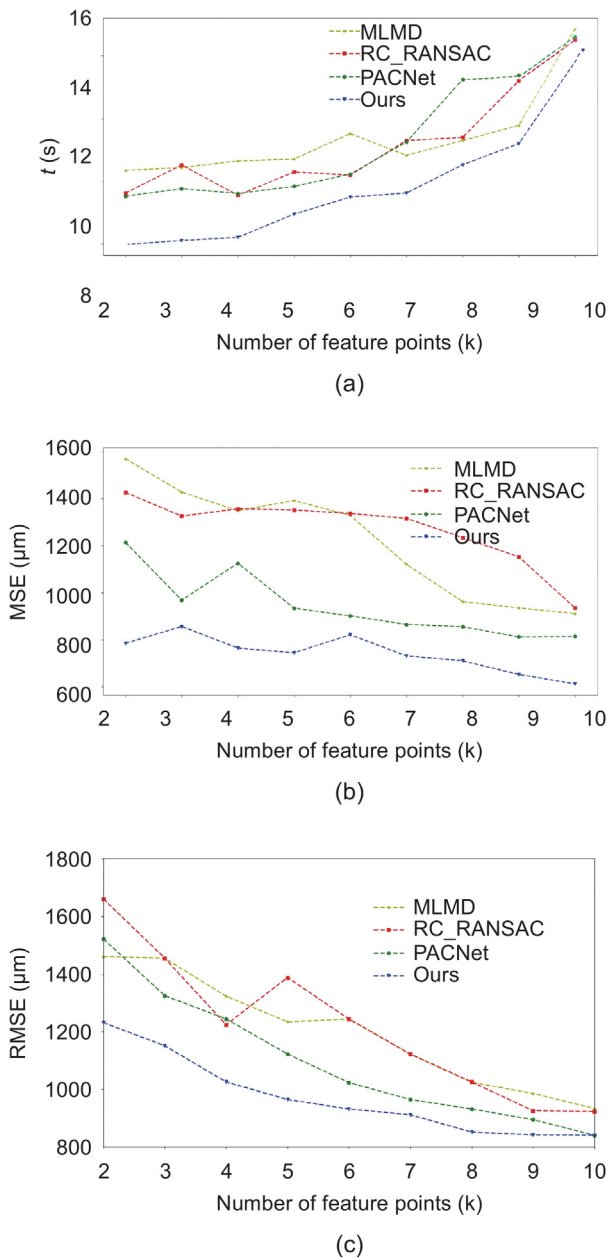


Fig. 11. Comparison diagram: (a) Time comparison. (b) MSE comparison. (c) RMSE comparison.

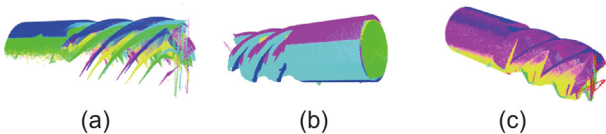


Fig. 12. (a) Initial scanning data. (b) Coarse registration results after rotation and translation of the motion model. (c) Final registration results.

4. Conclusion

This paper presents a multi-view point cloud registration algorithm based on a robotic measurement system, designed to tackle the challenge of registering low-overlap point cloud data obtained from multi-view scans of complex parts. The algorithm takes the source point cloud and the target point cloud as input,

processes them, and optimizes the point cloud data. The optimized point cloud data is then fed into the DBR-Net algorithm, which processes the translation features and rotation features of the point clouds using dual encoders. The translation features of the source and target point clouds are input into the dual decoders, while the rotation features are processed accordingly. Ultimately, the algorithm outputs the matching translation and rotation features between the source and target point clouds. These features are then fused and passed to the RANSAC algorithm to estimate the spatial transformation matrix. The source point cloud is aligned according to this matrix, resulting in the registered outcome. The proposed robotic scanning system effectively addresses the occlusion problem encountered with complex parts. Meanwhile, the registration algorithm refines the feature points during point cloud registration, optimizing both the time and accuracy of the registration process. This provides an effective method for multi-view point cloud registration.

CRedit authorship contribution statement

Chuangchuang Li: Writing – original draft, Validation, Investigation. **Xubin Lin:** Writing – review & editing, Methodology. **Zhaoyang Liao:** Writing – review & editing, Visualization, Methodology, Conceptualization. **Hongmin Wu:** Writing – review & editing, Visualization. **Zhihao Xu:** Writing – review & editing, Supervision. **Xuefeng Zhou:** Writing – review & editing, Supervision, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was co-supported by the National Natural Science Foundation of China (U22A20176), Guangdong Basic and Applied Basic Research Foundation, China (2022B1515120078), the Guangdong Basic and Applied Basic Research Foundation, China (2021A1515110898), GDAS' Project of Science and Technology Development, China (2022GDASZH-2022010108), and the Key Areas R&D Program of Dongguan City, China (20201200300062).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.birob.2024.100195>.

References

- [1] W. Peng, Y. Wang, Z. Miao, M. Feng, Y. Tang, Viewpoints planning for active 3-D reconstruction of profiled blades using estimated occupancy probabilities (EOP), *IEEE Trans. Ind. Electron.* 68 (5) (2021) 4109–4119.
- [2] H. Xie, W. Li, H. Liu, General geometry calibration using arbitrary free-form surface in a vision-based robot system, *IEEE Trans. Ind. Electron.* 69 (6) (2022) 5994–6003.
- [3] J. Li, et al., The real-time local surface model construction method of unknown-model workpieces for robotic polishing, *IEEE/ASME Trans. Mechatronics*.
- [4] Z. Wu, et al., A systematic point cloud edge detection framework for automatic aircraft skin milling, *IEEE Trans. Ind. Inform.* 20 (1) (2024) 560–572.
- [5] Z. Wu, et al., Gravitational discriminative optimization for multiview reconstruction of free-form surface, *IEEE/ASME Trans. Mechatronics* 28 (6) (2023) 3226–3237.
- [6] L. He, et al., GFOICP: Geometric feature optimized iterative closest point for 3-D point cloud registration, *IEEE Trans. Geosci. Remote Sens.* 61 (2023) 5704217, 1–17.

- [7] P.J. Besl, N.D. McKay, A method for registration of 3D shapes, *IEEE Trans. Pattern Anal. Mach. Intell.* 14 (2) (1992) 239–256.
- [8] M.A. Alnagdawi, S.Z. Mohd Hashim, ORB-PC feature-based image registration, in: 2019 IEEE International Conference on Signal and Image Processing Applications, ICSIPA, Kuala Lumpur, Malaysia, 2019, pp. 111–115.
- [9] R. Lei, et al., Deep global feature-based template matching for fast multi-modal image registration, in: 2021 IEEE International Geoscience and Remote Sensing Symposium, IGARSS, Brussels, Belgium, 2021, pp. 5457–5460.
- [10] L. Pan, et al., OCTRexpert: A feature-based 3D registration method for retinal OCT images, *IEEE Trans. Image Process.* 29 (2020) 3885–3897.
- [11] J. Sun, et al., Multi-view SAR image registration based on feature extraction, in: 2021 2nd China International SAR Symposium, CISS, Shanghai, China, 2021, pp. 1–8.
- [12] W.-J. Lee, S.-J. Oh, Remote sensing image registration using equivariance features, in: 2021 International Conference on Information Networking, ICOIN, Jeju Island, Korea (South), 2021, pp. 776–781.
- [13] W. Lu, et al., DeepVCP: An end-to-end deep neural network for point cloud registration, in: 2019 IEEE/CVF International Conference on Computer Vision, ICCV, Seoul, Korea (South), 2019, pp. 12–21.
- [14] Y. Wang, et al., CCAG: End-to-end point cloud registration, *IEEE Robot. Autom. Lett.* 9 (1) (2024) 435–442.
- [15] Y. Wang, J. Solomon, Deep closest point: Learning representations for point cloud registration, in: 2019 IEEE/CVF International Conference on Computer Vision, ICCV, Seoul, Korea (South), 2019, pp. 3522–3531.
- [16] X. Huang, et al., Feature-metric registration: A fast semi-supervised approach for robust point cloud registration without correspondences, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, Seattle, WA, USA, 2020, pp. 11363–11371.
- [17] J. Du, et al., ATNet: Unsupervised point cloud registration network integrating adaptive graph convolution and transformer, in: 2023 China Automation Congress, CAC, Chongqing, China, 2023, pp. 5202–5207.
- [18] P. Kadam, et al., R-PointHop: A green, accurate, and unsupervised point cloud registration method, *IEEE Trans. Image Process.* 31 (2022) 2710–2725.
- [19] Z. Zhang, et al., A representation separation perspective to correspondence-free unsupervised 3-D point cloud registration, *IEEE Geosci. Remote Sens. Lett.* 19 (2022) 7003005, 1–5.
- [20] X. Huang, et al., Unsupervised point cloud registration by learning unified Gaussian mixture models, *IEEE Robot. Autom. Lett.* 7 (3) (2023) 7028–7035.
- [21] Y. Jiang, et al., GTINet: Global topology-aware interactions for unsupervised point cloud registration, *IEEE Trans. Circuits Syst. Video Technol.* 34 (7) (2024) 6363–6375.
- [22] Z. Chen, et al., VK-Net: Category-level point cloud registration with unsupervised rotation invariant keypoints, in: 2021 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, Toronto, ON, Canada, 2021, pp. 1900–1904.
- [23] Z. Huang, et al., PF-Net: Point fractal network for 3D point cloud completion, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, Seattle, WA, USA, 2020, pp. 7659–7667.
- [24] Q. Mei, et al., PACNet: A high-precision point cloud registration network based on deep learning, in: 2021 13th International Conference on Wireless Communications and Signal Processing, WCSP, Changsha, China, 2021, pp. 1–5.
- [25] B. Eckart, et al., MLMD: Maximum likelihood mixture decoupling for fast and accurate point cloud registration, in: 2015 International Conference on 3D Vision, Lyon, France, 2015, pp. 241–249.
- [26] J. Han, et al., An improved RANSAC registration algorithm based on region covariance descriptor, in: 2015 Chinese Automation Congress, CAC, Wuhan, 2015, pp. 746–751.