I PointNet++: Improved PointNet++ for Segmentation and Localization of Leather Grasp Points

by
Guang Jin,[1] Gongchang Ren,[1] Yuan Huan[1]and Jiangong Sun[1]
[1]*College of Mechanical & Electrical Engineering at Shaanxi University of Science & Technology*,
Xi'an, Shaanxi, 710021, China

1 - Abstract

In order to achieve accurate identification and positioning of leather grasp points during the process of robot grasp and spreading leather, this paper proposes a leather grasp point segmentation and positioning method based on improved PointNet++(IPointNet++). Taking leather in its natural falling state as the research object, a depth camera is used to collect point cloud data of the leather. Firstly, the preprocessing of leather point clouds is completed by removing background point clouds based on PassThroughFilter and eliminating noise based on Statistics Filter. Secondly, the octree sampling method is used to replace the farthest point sampling method of the original PointNet++, which is adapted to the non-rigid deformation characteristics of the leather itself. Thereby, the entire leather is divided into two parts: the main body and the grasp area. Lastly, the three-dimensional coordinates of the leather grasp points are obtained by solving the centroid of the point cloud data in the leather grasp area grasp. In the segmentation experiments, the improved PointNet++ has raised the mIoU by 11.8% and 2.5% comparing with PointNet and PointNet++ respectively, and the OA by 6.1% and 1.1%. In the grasp experiments, the success rate of leather grasp points identification grasp is 93.33%, and the success grasp rate grasp is 82.14%. The experimental results show that the proposed method has higher segmentation accuracy and good applicability.

2 - Background

The leather manufacturing industry is an important branch of the light industry and plays a significant role in the manufacturing industry. However, in the current stage, traditional methods are still used for handling and spreading leather in some processes and inter processes, which require a large amount of labor. High labor intensity and poor working conditions still exist in the whole process. The leather processing industry is at the critical period of transformation and upgrading, and it is urgent to solve the key problem of how to combine robot technology to achieve automation in various processes of leather production.

To address the automation of leather operations using robot technology, Huan et al.[1] designed a specifical robotic hand for grasp and spreading leather, and enabled the grasp and spreading operations on the leather by dual robotic arms. However, due to some factors of the leather itself, such as the irregular edges, deformational susceptibility under stress, and easy occlusion caused by bending, furthermore, the influence of external factors such as cluttered backgrounds and lighting changes in the processing environment, the recognition of the grasp area of the leather is a challenging problem during the dual-arm grasp and spreading process.

In order to solve the problem of difficult recognition and positioning of the grasp area and obtaining the corresponding three-dimensional coordinates grasp during the dual-arm grasp and spreading process, this paper uses point cloud data that can reflect spatial information as the foundation and adopts a deep learning-based 3D point cloud segmentation algorithm to recognize and segment the target area,[2-4] thereby the three-dimensional coordinates of the lowest grasp area of the leather is obtained grasp.

3D point cloud segmentation based on deep learning can be divided into indirect and direct methods. Indirect segmentation involves projecting the point cloud into multiple views of 2D images and performing convolutional operations on the projected 2D data to achieve point cloud segmentation. SU et al.[5] proposed MVCNN (multi-view convolutional neural network), which extracts features from 2D images of three-dimensional objects under multiple views, and finally aggregates the 2D images from multiple views through pooling and fully connected operations to obtain the segmentation result. However, this method loses the three-dimensional spatial features of the target, resulting in lower segmentation accuracy. FENG et al.[6] proposed GVCNN (group-view convolutional neural network), which groups the descriptors extracted from 2D images under different views, solving the problem of losing three-dimensional spatial features in MVCNN under multiple views.

Direct segmentation, as a method that directly operates on raw point cloud data, reduces computational complexity and processing errors. QI et al.[7] proposed the PointNet network, which directly processes the data and extracts the feature information of the target from the point cloud data. The core of PointNet is the adoption of Multi-Layer Perceptron (MLP) for feature extraction

Corresponding author email: rengc@sust.edu.cn
Manuscript received November 27, 2023, accepted for publication January 22, 2024.

of each point in the data, and the use of the symmetric function Maxpool to obtain information for each point, which solves the issues of unorderedness, permutation invariance, and rotation invariance of point clouds.[8] However, PointNet does not extract and process local features, without considering the relationships between points and their neighbors. Therefore, QI et al.[9] proposed PointNet++, which uses farthest point sampling (FPS) to define multiple local regions as clusters and selects the centroid of each cluster. By searching for the neighboring points of the centroids, multiple subsets of point clouds are obtained, and the features of these subsets are obtained through convolution and pooling operations. PointNet++ enables better extraction and processing of local features and addresses the issue of uneven sample distribution while considering the relationships between points. Zhao et al. [10] proposed Point Transformer, which is constructed based on the self-attention mechanism[11] model and incorporates position encoding. The proposed method can effectively handle unordered point cloud data and achieve point cloud segmentation by utilizing the mutual relationships between points in local neighborhoods. Indirect segmentation methods suffer from incomplete three-dimensional spatial information, while direct segmentation methods can obtain complete feature information and extract local feature information of the target[12]. Considering the characteristics of leather itself, in order to capture the local feature information of the grasp area of leather better, this study adopts the PointNet++ network. However, the farthest point sampling (FPS) used in PointNet++ cannot adapt to the non-rigid deformation of leather. To segment the leather point cloud better, the original algorithm's FPS method is replaced with octree sampling in this paper, which allows the network to adapt to the characteristics of leather under different grasp states better, the segmentation of leather point cloud data is achieved consequently.

3 - Materials and Methods

This paper proposes an improved PointNet++ point cloud segmentation model, which can segment leather in its natural drooping state and grasp area, and then locates the three-dimensional coordinates of the grasp area based on the segmentation results. The specific process is shown in Figure 1.

3.1 - Data Acquisition

The experiment utilized the Intel RealSense D435i depth camera for data collection. The camera has a resolution of 1280×720 and a depth field of view of 69.4°×42.5°. The camera was positioned 1.5m away from the object and fixed in front of the robotic arm, as shown in Figure 2. The data collection was conducted under conditions with no specific background or lighting, and the leather was grasped by the robotic arm in its natural droop state for 3D point cloud acquisition. Due to the diversity and variability of leather shapes,

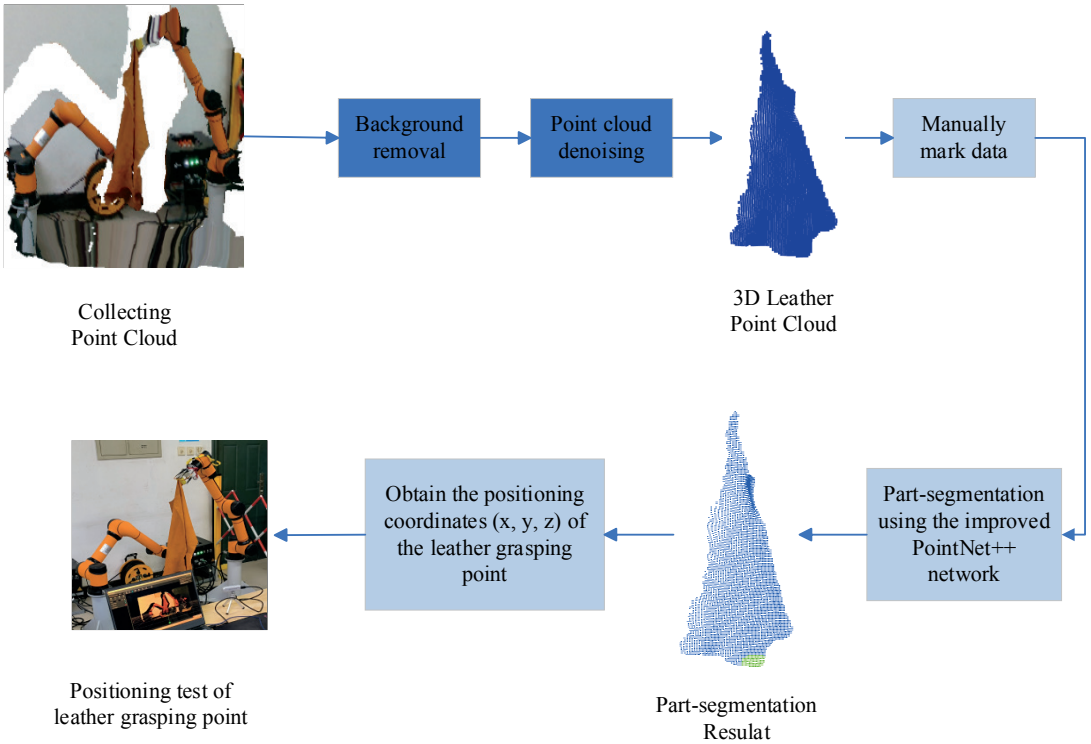

Figure 1. Location process of leather grabbing area based on point cloud information.

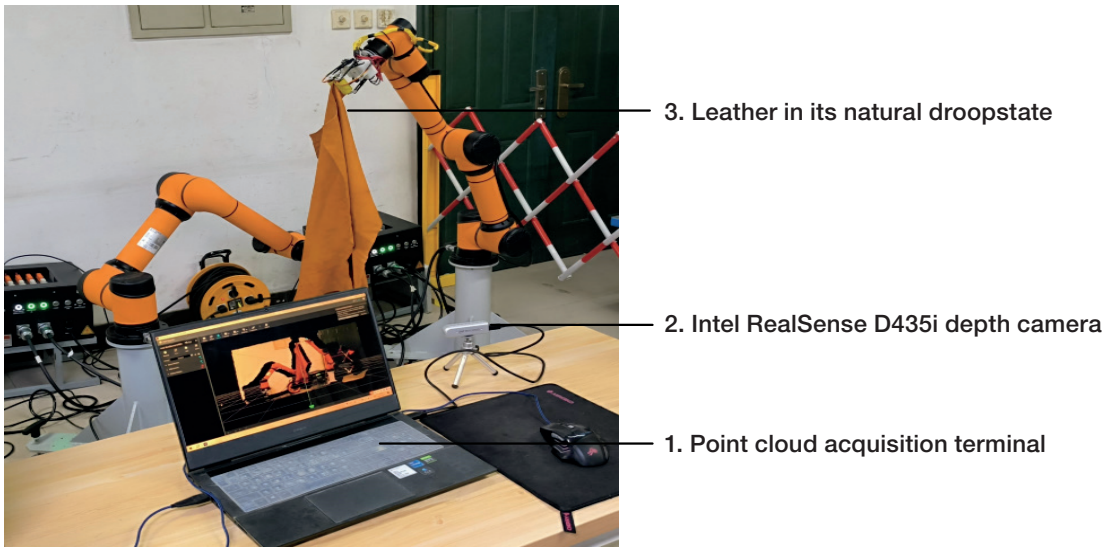

Figure 2. Illustration of leather point cloud acquisition

20 leather samples were selected, and each sample was randomly drooped 10 times, resulting in a total of 200 samples.

3.2 - Point Cloud Preprocessing

During the point cloud data acquisition, background and a certain amount of noise points are inevitably captured. Therefore, appropriate preprocessing operations are performed on the raw point cloud data before segmentation to remove redundancies and improve the training speed of the network. A passthrough filter is applied to remove most of the background, which is followed by a statistical filter to remove noise.

3.2.1 - Background Removal with Passthrough Filter

A passthrough filter was used to remove the background. The threshold values for the X, Y, and Z directions are set to $X \in [-0.25m, 0.05m]$, $Y \in [-0.2m, 0.6m]$ and $Z \in [-1.8m, 1.4m]$ to achieve effective background removal. The filtered result is shown in Figure 3, which demonstrates that the passthrough filter removes most of the background while preserving the relevant information of the leather effectively.

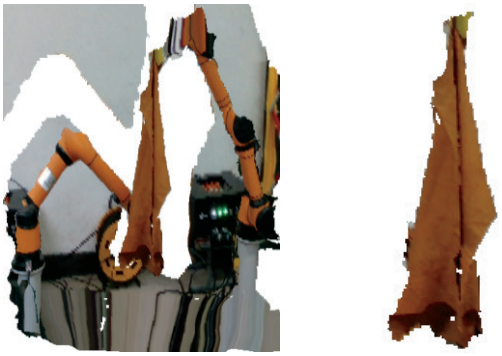

Figure 3. Effect of Passthrough Filter

3.2.2 Statistics Filtering for Noise Removal

Noise points in the point cloud can have a significant impact on the localization of the leather grasp region. Statistics filtering is used to remove noise points around the leather. First, the number of points in the point cloud data, denoted as n, and the number of neighboring points, denoted as m, are calculated. The distance between each point and its neighboring points, denoted as d_{ij}, is computed, where $i = \{1,2\cdots, n\}$ and $j = \{1,2\cdots, m\}$. The distances between all points and their neighboring points are then used to calculate the Gaussian distribution, which has mean distance μ and distance standard deviation σ.

The distance between the i-th point with coordinates $P_i = (X_i, Y_i, Z_i)$ and any neighboring point $P_j = (X_j, Y_j, Z_j)$ is calculated using the following formula:

$$d_{ij} = \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2 + (Z_i - Z_j)^2} \tag{1}$$

The mean distance μ and distance standard deviation σ for each point to its neighboring points are calculated by iterating through the point cloud, using the following formulas:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} d_{ij} \tag{2}$$

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (d_{ij} - \mu)^2} \tag{3}$$

Noise points are removed when the condition $\mu - \omega\sigma < d_{ij} < \mu + \omega\sigma$ (where ω is a multiple of the standard deviation) is met. Through experiments, it has been found that $m = 20$ and $\omega = 0.8$ can effectively remove noise points around the leather, as shown in Figure 4.

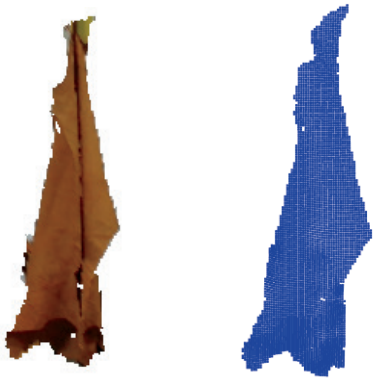

Figure 4. Point cloud data before and after statistical filtering

Figure 5. Segmentation diagram of the leather grasp region

3.3 - Data Labeling

Based on the analysis of the operation process of leather grasp and spreading by the dual robotic arm, it is necessary to identify the lowest corner points of the leather. Therefore, the leather needs to be divided into two parts: the leather body and the grasp region. The segmentation diagram is shown in Figure 5.

In the process of deep learning point cloud segmentation, the training set needs to be manually labeled. Due to the inherent characteristics of leather and the influence of external environmental factors, the grasp region of the leather in its natural droop state may exhibit concave and up wrap phenomena. Therefore, the lowest corner point of the leather is used for labeling the grasp region. The data set is labeled into two categories, with the leather body as "0" and the grasp region as "1". The specific labeling process is shown in Figure 6.

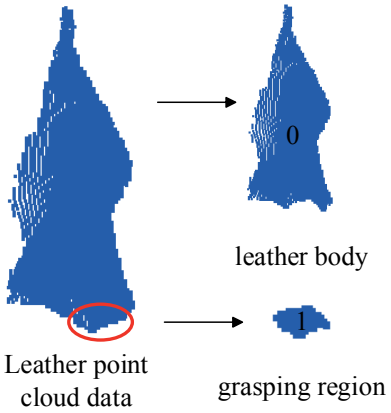

Figure 6. Partial manual labeling of leather point cloud

4 - Improvement of the PointNet++ Model (I PointNet++)

Octree sampling is a top-down progressive spatial partitioning structure. In order to address the flexible deformation characteristics of leather, octree sampling is used to process the point cloud data systematically, which can achieve a good topological relationship of the unordered point cloud data and enable k-nearest neighbor search for feature points.

The leather point cloud was divided using octree sampling. Firstly, the parent node of the preprocessed leather point cloud was determined. Then, the parent node was divided into eight sub-nodes along the directions of the spatial coordinate system. Each octree node represents a cube, and each parent node was divided into eight sub-nodes. All nodes were subdivided into eight equal parts until the side length of the cube was smaller than the set threshold, when the partitioning terminated. The illustration of octree partitioning is shown in Figure 7.

The IPointNet++ network structure is shown in Figure 8. It takes a preprocessed single leather point cloud data [*npoint* × 3] as input, where npoint is the number of points in the point cloud and 3 represents the point cloud coordinates (*X*, *Y*, *Z*). The leather point cloud was sampled and divided based on the feature points obtained from octree sampling. The octree has 5 layers, and the number of center points for sampling is n_1. The neighborhood of center points (nsample) is $n_2 = 64$. The local point clouds sampled and grouped by PointNet layers were encoded to obtain feature vectors of size (1024, 256). The octree sampling, grouping, and encoding of local point

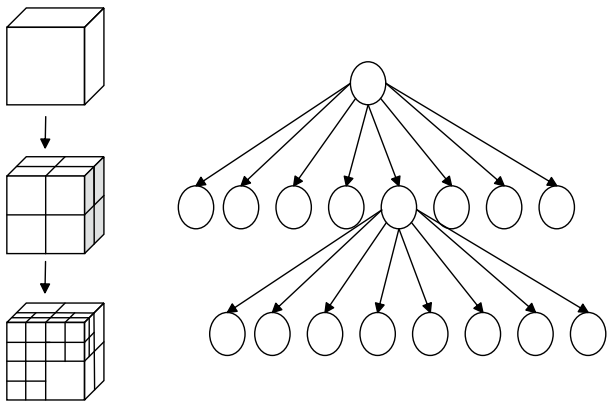

Figure 7. Illustration of octree partitioning

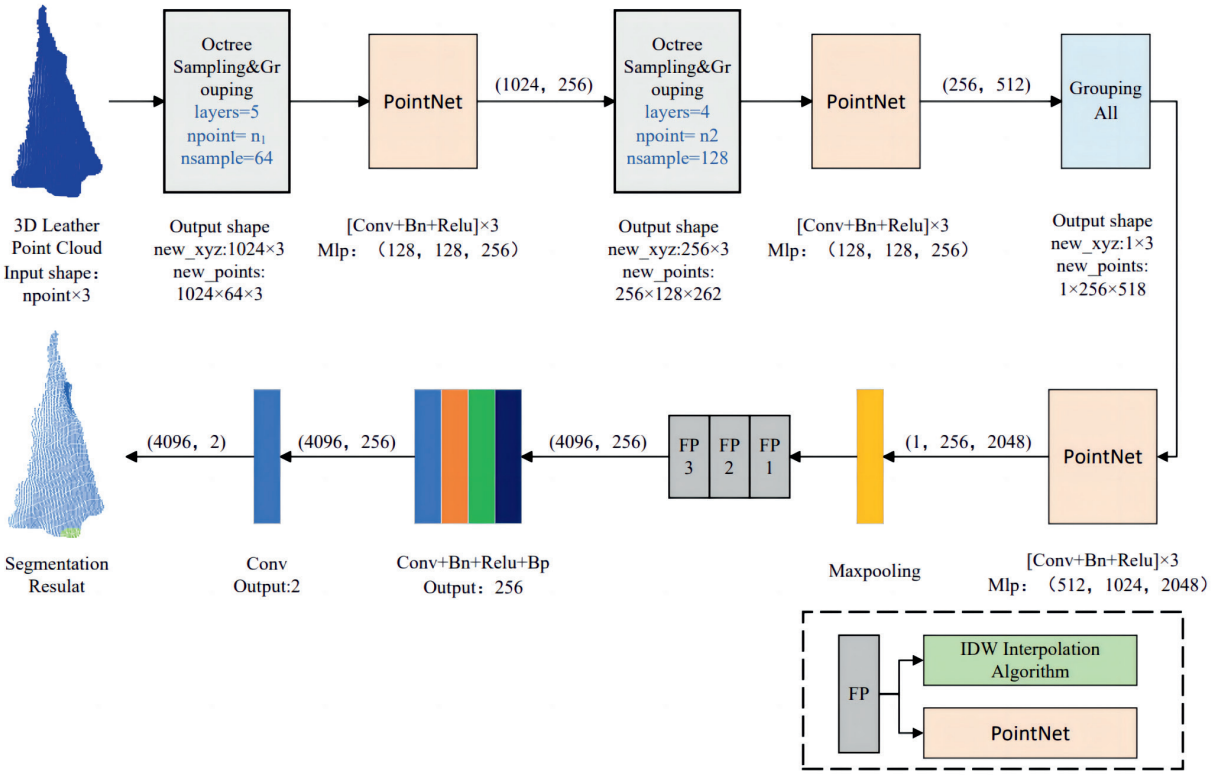

Figure 8. Schematic diagram of IPointNet++

clouds were repeated, resulting in feature vectors of size (256, 512). In this case, the octree has 4 layers and the number of neighborhood points for center points is n_2. The feature vectors were then encoded by PointNet layers. After undergoing max pooling and three FP (Feature Propagation) layers, where the FP layers involve Inverse Distance Weighted (IDW) interpolation algorithm and PointNet operations, the feature vectors of size (4096, 256) were obtained. The IDW interpolation formula is as follows:

$$f^{(i)}(x) = \frac{\sum_{i=1}^{k} w_i(x) f_i^{(j)}}{\sum_{i=1}^{k} w_i(x)} \tag{4}$$

$$w_i(x) = \frac{1}{d(x,x_i)^p} \tag{5}$$

Where f represents the interpolated feature value, $d(x, x_i)$represents the distance, p represents the index, k is the number of neighborhood points, and $j=1,2,\cdots,C$.

The obtained feature vectors of size (4096, 256) were then passed through a Conv+Bn+Relu+Bp (Convolution + Batch Normalization + Activation Function + Backpropagation) layer, the two classes of feature points (leather body and leather grasp region) was obtained by one Convolution operation grasp. The maximum value was taken as the output result for segmenting the leather point cloud. Point cloud segmentation is a classification problem for each segmentation point, the Cross Entropy Loss function was used as the loss function for the network model, with the formula as follows:

$$\text{Loss} = -\frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=0}^{K-1} y_{i,k} \ln p_{i,k} \tag{6}$$

Where $y_{i,k}$ represents the true label of sample i as k, with K label values and N samples, and $p_{i,k}$ represents the predicted label value for sample i as k.

5 - Localization of Leather Grasp Points

During the process of leather grasp point localization, the opening range of the robotic hand is 15cm. To ensure reliable grasp, the centroid of the leather grasp region is selected as the optimal grasp point. As shown in Figure 9, O-XYZ represents the spatial coordinate system of the target.

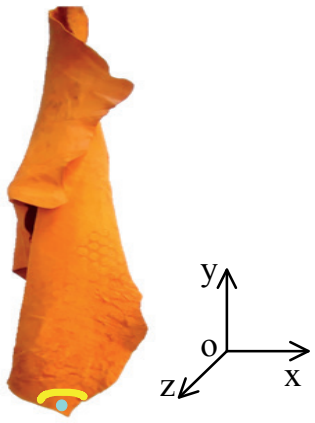

Figure 9. Coordinate system of the target

The formula for calculating the centroid of the leather grasp region is as follows:

$$P_c = \sum_{i=0}^{n} m_i r_i \tag{7}$$

where m_i represents the mass of the point cloud, $r_i = (x_i, y_i, z_i)$ and m_i is set to 1. Therefore, the formula for the centroid coordinates of the point cloud is as follows:

$$P_c\,(x_c, y_c, z_c) = \frac{1}{n}\left(\sum_{i=0}^{n} x_i, \sum_{i=0}^{n} y_i, \sum_{i=0}^{n} z_i\right. \tag{8}$$

where n represents the number of points in the point cloud, and $i = \{1, 2, ..., n\}$.

6 - Experimental Process and Result Analysis

6.1 - Experimental Conditions

The experiments were conducted in a Windows 11 operating system environment, using the Pytorch framework. For the localization experiments, the Intel RealSense D435i depth camera was used to capture the leather point cloud, and the AUBO-i5 dual-arm robot was used for the grasp localization experiments. The specific experimental configuration is shown in Table I.

6.2 - Segmentation Experiment

6.2.1 - Experimental Design

To verify the effectiveness of the proposed algorithm, comparative experiments were conducted on the IPointNet++ algorithm, the original PointNet algorithm, and the original PointNet++ algorithm under the same conditions which includes training data, number of iterations, optimizer, learning rate, batch size, and training device. The training parameters were set as shown in Table II.

The average ratio of the intersection to the union of predicted values and true values for each category is represented by mIoU. The formula is as follows:

$$\mathrm{mIOU} = \frac{1}{N}\sum_{i=0}^{N} \frac{p_{ii}}{\sum_{j=0}^{N} pij + \sum_{j=0}^{N} pij - pij} \tag{9}$$

OA represents the ratio of correctly classified point clouds to the total number of point clouds. The formula is as follows:

$$\mathrm{OA} = \frac{\sum_{i=0}^{N} p_{ii}}{\sum_{i=0}^{N}\sum_{j=0}^{N} pij} \tag{10}$$

where N represents the number of categories in the point cloud dataset, p_{ii} represents the correctly predicted point clouds for a category, p_{ij} represents the incorrectly predicted point clouds for a

Table I

Experimental configuration

Parameters	Configuration
Operating System	Windows 11
Graphics Memory	12GB
RAM	64GB
GPU	NVIDIA GeForce RTX 3060
GPU Acceleration Environment	CUDA11.3
Training Framework	Pytorch
Programming Language Environment	Python 3.9
Depth Camera	Intel RealSense D435i
Dual Robotic Arms	AUBO-i5

Table II

Training parameter settings

Parameters	Value
epochs	200
batch-size	16
Learning rate	0.001
Attenuation rate	0.5

Evaluation metrics: Mean Intersection over Union (mIoU) and Overall Accuracy (OA).

category, and p_{ij} represents the point clouds of category j predicted as category i.

6.2.2 - Results and Analysis

The obtained training results are shown in Table III. Compared with the PointNet algorithm, the IPointNet++ algorithm has enhanced mIoU by 11.8% and OA by 6.1%. Compared with the original PointNet++ algorithm, the IPointNet++ algorithm has enhanced mIoU by 2.5% and OA by 1.1%. From the results of the model training, it can be seen that the performance of the IPointNet++ algorithm has been improved greatly, and the effectiveness of the proposed algorithm has been validated. The proposed IPointNet++can provide leather grasp point recognition and segmentation for subsequent grasp localization experiments.

To further validate the applicability and feasibility of the proposed algorithm, unsampled leather was selected for verification. As shown in Figure 10, when facing different leather point cloud samples, it can be seen that the proposed algorithm can still effectively segment the leather into the main body and the leather grasp region accurately and clearly.

To evaluate the comprehensive segmentation capability of the proposed algorithm further, a comparison was made with the PointNet algorithm and the original PointNet++ algorithm for the segmentation task using unsampled leather. The segmentation results of the three algorithms are shown in Figure 11. It can be observed that the IPointNet++ algorithm can effectively segment the leather precisely and accurately, which can meet the subsequent localization of leather grasp regions. Due to its inability to adapt well to the non-rigid deformation, the original PointNet++ algorithm segments the grasp region of the leather into a larger area, which can lead to excessive segmentation of the grasp region, which will affect the localization accuracy of the leather grasp points. The PointNet algorithm shows either small or incorrect segmentation results for the leather, ascribing to its inability to handle the relationships between points and their neighbors well, and cannot perform local feature extraction which can greatly affect the localization of the grasp region and processing specifically for the leather grasp.

6.3 - Grasp Localization Experiment

To verify the effectiveness of the proposed localization algorithm, the three-dimensional coordinates of the segmented leather grasp

Table III

Segmentation accuracy results (%)

Model	mIoU	OA	Single-Class Classification	
			Leather Main Body	Leather Grasp Region
PointNet	70.8	89.2	92.3	89.6
PointNet++	80.1	94.2	95.1	92.7
Improved PointNet++	82.6	95.3	96.2	93.1

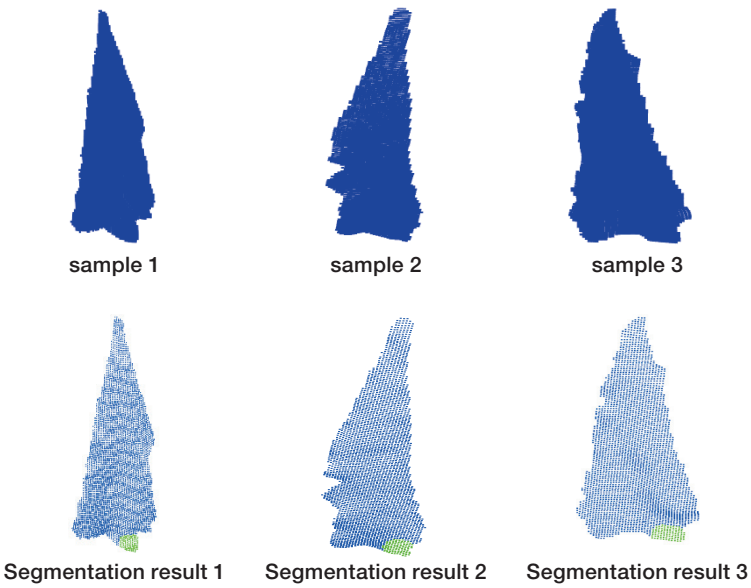

Figure 10. Segmentation results of unsampled leather point cloud

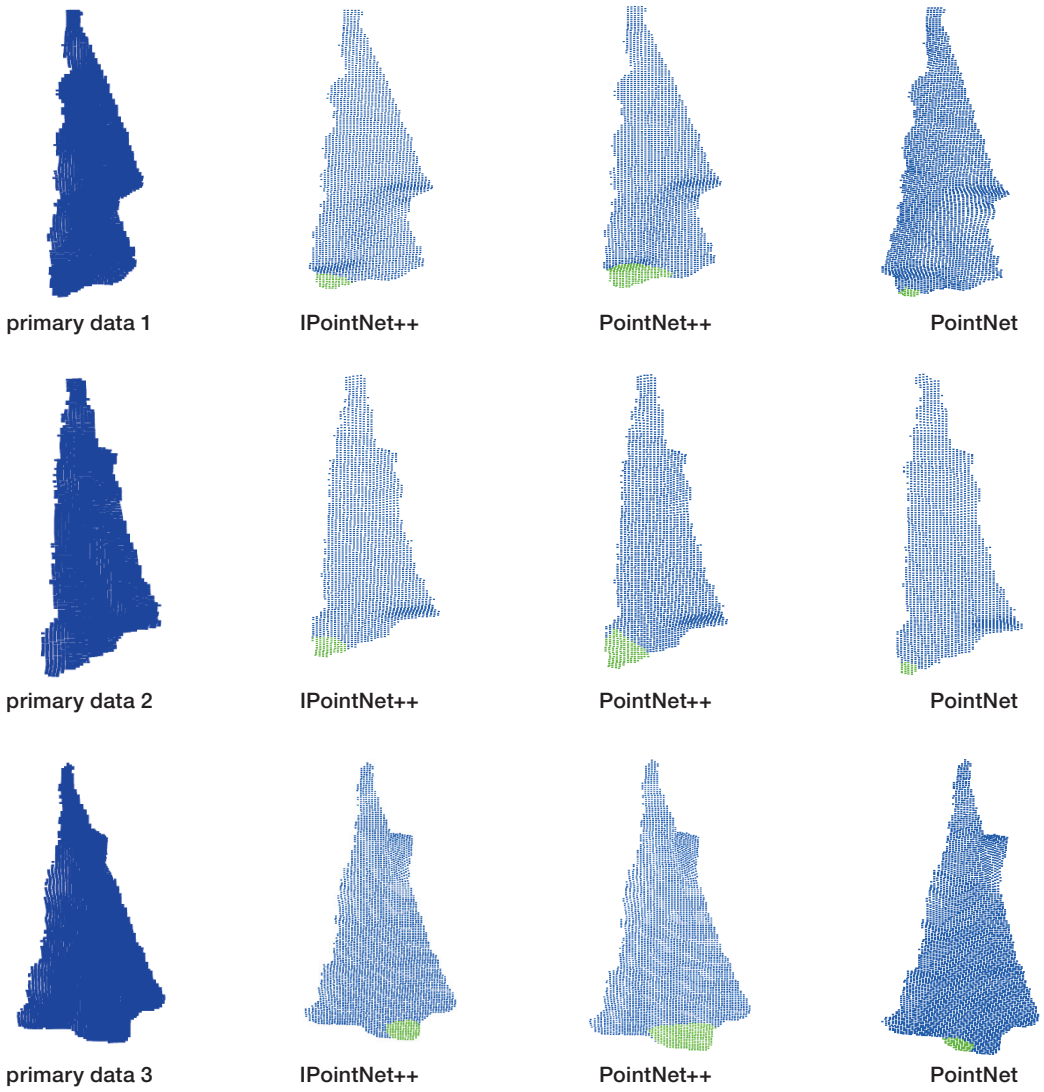

Figure 11. Comparison of segmentation results using different algorithms

region centroids were passed to the robotic arm for the leather grasp. The check whether it could accurately grasp the leather based on the input coordinate values was tested. Thirty random leather samples were selected for the experiment, and the leather was placed on the test bench. The robotic arm was used to grasp the leather and place it in a natural drop state. Then, the proposed algorithm was used to recognize and segment the leather grasp region, and the three-dimensional coordinates of the leather grasp region centroid were obtained. The coordinates were then passed to another robotic arm to check whether it could accurately grasp the leather grasp point.

Evaluation metrics: The success rate, which was calculated as the ratio of the successful grasp number to the total experimental number.

6.3.1 - Results and Analysis

The overall experimental process is shown in Figure 12.

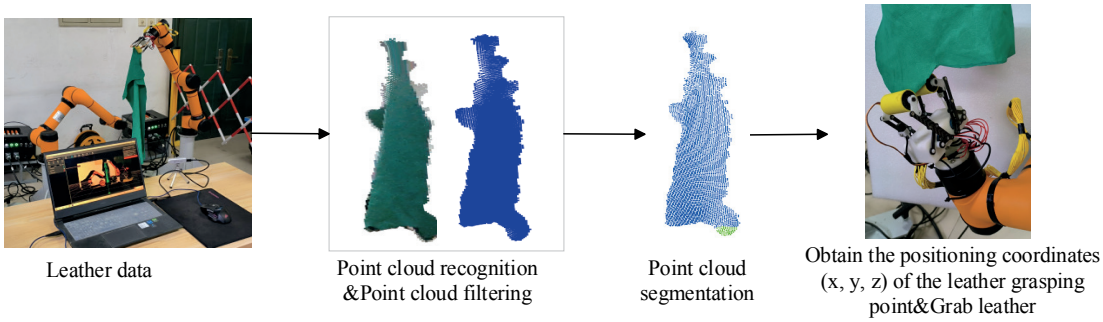

Figure 12. Overall experimental process

Table IV

Test data for grasp localization experiment

Identification experiment/times	Successful identification/times	Failure identification/ times	Success rate/%
30	28	2	93.33
Grab experiment/times	Successfully grabbed/ times	Failed fetching/times	Success rate/%
28	23	5	82.14

The test data is shown in Table IV.

According to Table IV, for the 30 experiments, the leather grasp points were successfully recognized 28 times, with a success rate of 93.33%. The result showed that the recognition success rate was affected by the defects (damage, rough edges) of the leather which can lead to recognition errors and inability to segment the leather grasp region. The total number of grasp attempts was 28 times, with 23 successful grasp and a success rate of 82.14%. The analysis showed that the grasp success rate was affected by factors which can make it difficult to accurately grasp the leather grasp point such as the recognition success rate, the grasp posture of the robotic arm, and the opening angle of the robotic arm. In summary, the proposed algorithm can meet the recognition, segmentation, and localization of leather grasp points better, but future research is still needed to determine the optimal grasp posture and opening angle of the robotic arm to improve the overall grasp success rate furtherly.

7 Conclusion

To achieve the segmentation and localization of leather grasp points during the grasp process, the IPointNet++ was proposed in this paper, which replaced the farthest point sampling method of the original PointNet++ algorithm with the octree sampling method, achieved segmentation of the leather that is prone to non-rigid deformation, obtained the leather grasp region, calculated the centroid of the leather grasp point, obtained the three-dimensional coordinates of the grasp point and grasped the leather using a dual-arm robot.

1. In the segmentation experiment, the proposed IPointNet++ algorithm increased the MIoU by 2.5% and the OA by 1.1% compared with the original PointNet++ algorithm.

2. In the grasp localization experiment, the recognition success rate of the leather grasp points was 93.33%, and the grasp success rate was 82.14%. The Analysis showed the grasp posture and opening angle of the robotic arm are the main factors which can affect the grasp success rate.

In summary, the proposed IPointNet++ algorithm can recognize, segment, and locate leather grasp points effectively, which has significant implications for advancing the use of robots in leather processing.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Author contributions

GJ and YH conceptualized the project, developed the methodology, prepared the original draft, and carried out the experiments. GR supervised the project and was responsible for the project administration. JS assisted in the experiment. All the authors have read and agreed to the published version of the paper.

Competing interests

The contact author has declared that none of the authors have any competing interests.

Financial support

This research has been supported by Xi'an Science and Technology Plan Project (grant no. 23ZDCYJSGG0016-2022) and Shaanxi Provincial Key R&D Program Funding Project (grant no. 2022GY-250).

References

1. Huan, Y.; Ren, G.; Su, X.; Tian, W., A versatile end effector for grabbing and spreading of flaky deformable object manipulation. *Mech. Sci.* **2023,** *14* (1), 111-123.
2. Rusu, R. B.; Marton, Z. C.; Blodow, N.; Dolha, M.; Beetz, M., Towards 3D Point cloud based object maps for household environments. *Robotics and Autonomous Systems* **2008,** *56* (11), 927-941.
3. Li, J.; Chen, B. M.; Lee, G. In *SO-Net: Self-Organizing Network for Point Cloud Analysis*, 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 18-23 June 2018; 2018; pp 9397-9406.

4. Zhao, H.; Jiang, L.; Fu, C. W.; Jia, J., Pointweb: Enhancing local neighborhood features for point cloud processing. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019; Vol. 2019-June, p 5560.
5. Su, H.; Maji, S.; Kalogerakis, E.; Learned-Miller, E. In *Multi-view Convolutional Neural Networks for 3D Shape Recognition*, 2015 IEEE International Conference on Computer Vision (ICCV), 7-13 Dec. 2015; 2015; pp 945-953.
6. Feng, Y.; Zhang, Z.; Zhao, X.; Ji, R.; Gao, Y. In *GVCNN: Group-View Convolutional Neural Networks for 3D Shape Recognition*, 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 18-23 June 2018; 2018; pp 264-272.
7. Charles, R. Q.; Su, H.; Kaichun, M.; Guibas, L. J. In *PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation*, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 21-26 July 2017; 2017; pp 77-85.
8. Mao, J.; Wang, X.; Li, H. In *Interpolated Convolutional Networks for 3D Point Cloud Understanding*, 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 27 Oct.-2 Nov. 2019; 2019; pp 1578-1587.
9. Qi, C. R.; Yi, L.; Su, H.; Guibas, L. J., PointNet++: deep hierarchical feature learning on point sets in a metric space. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Curran Associates Inc.: Long Beach, California, USA, 2017; pp 5105–5114.
10. Zhao, H.; Jiang, L.; Jia, J.; Torr, P.; Koltun, V. In *Point Transformer*, 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 10-17 Oct. 2021; 2021; pp 16239-16248.
11. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; Polosukhin, I., Attention Is All You Need. *arXiv e-prints* **2017**, arXiv:1706.03762.
12. Li, Y.; Bu, R.; Sun, M.; Wu, W.; Di, X.; Chen, B., PointCNN: convolution on X-transformed points. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Curran Associates Inc.: Montréal, Canada, 2018; pp 828–838.