

Classification of rice seed variety using point cloud data combined with deep learning

Yan Qian^{1*}, Qianjin Xu¹, Yingying Yang¹, Hu Lu¹, Hua Li², Xuebin Feng², Wenqing Yin²

(1. College of Artificial Intelligence, Nanjing Agriculture University, Nanjing 210031, China;

2. College of Engineering, Nanjing Agriculture University, Nanjing 210031, China)

Abstract: Rice variety selection and quality inspection are key links in rice planting. Compared with two-dimensional images, three-dimensional information on rice seeds shows the appearance characteristics of rice seeds more comprehensively and accurately. This study proposed a rice variety classification method using three-dimensional point cloud data of the surface of rice seeds combined with a deep learning network to achieve the rapid and accurate identification of rice varieties. First, a point cloud collection platform was set up with a Raytrix light field camera as the core to collect three-dimensional point cloud data on the surface of rice seeds; then, the collected point cloud was filled, filtered and smoothed; after that, the point cloud segmentation is based on the RANSAC algorithm, and the point cloud downsampling is based on a combination of random sampling algorithm and voxel grid filtering algorithm. Finally, the processed point cloud was input to the improved PointNet network for feature extraction and species classification. The improved PointNet network added a cross-level feature connection structure, made full use of features at different levels, and better extracted the surface structure features of rice seeds. After testing, the improved PointNet model had an average classification accuracy of 89.4% for eight varieties of rice, which was 1.2% higher than that of the PointNet model. The method proposed in this study combined deep learning and point cloud data to achieve the efficient classification of rice varieties.

Keywords: rice seed, variety classification, point cloud data, deep learning, light field camera

DOI: 10.25165/j.ijabe.20211405.5902

Citation: Qian Y, Xu Q J, Yang Y Y, Lu H, Li H, Feng X B, et al. Classification of rice seed variety using point cloud data combined with deep learning. *Int J Agric & Biol Eng*, 2021; 14(5): 206–212.

1 Introduction

Rice is a major staple food in China and many other countries and is widely grown. The inspection and classification of seed varieties is an important link in the planting process, and it determines the yield and quality of rice. However, the continuous development of breeding technology has enabled an increasing number of rice varieties to be developed and put on the market to adapt to the planting environment and the tastes of people in different regions, increasing the difficulty of identifying rice varieties during the planting process^[1]. The development of machine vision technology and its application in agricultural product detection and identification can not only overcome the various shortcomings of traditional manual detection methods but also have the advantages of being noncontact, non-damaging, fast and accurate^[2]. Different varieties of rice seeds often show different external morphological characteristics, such as shape, color, and size^[3]. By inspecting the appearance characteristics of rice seeds,

the rice seed varieties can be classified. Kuo et al.^[4] used an optical microscope to obtain clear two-dimensional images of rice grains of thirty varieties, proposed a classification method based on image processing and sparse-representation-based classification (SRC), and finally achieved a recognition rate of 89.1%. Golpour et al.^[5] extracted thirty-six color features from the RGB, HIS, and HSV color spaces of rice grain images, and used a neural network with two hidden layers to classify rice varieties. Mittal et al.^[6] extracted geometric feature parameters from two-dimensional images of rice seeds based on image processing technology, and used support vector machines (SVM) to classify and evaluate rice varieties, the system achieved recognition accuracy of 93%. Fabiyi et al.^[7] extract the spatial and spectral feature values of rice seeds based on high-resolution RGB images and hyperspectral images, and use random forest classifiers for classification, which can effectively improve the purity of seeds. The two-dimensional image loses the high-dimensional information in the space during the imaging process, and the characteristics of rice seeds that can be extracted from it are limited, so the final recognition rate was not high. Compared with the two-dimensional image, the three-dimensional information obtained based on the surface of the rice seed could describe the appearance characteristics of the rice seed more completely and accurately and has more advantages in the task of classifying rice seed varieties. Qian et al.^[8] successfully constructed a three-dimensional model of rice seeds using the Depth from Focus (DFF) method. Based on a three-dimensional model of rice seeds, eight feature values were extracted and input to the BP neural network, which ultimately reached a recognition rate of 90%. Li et al.^[9] proposed a calculation method for the three-dimensional features of the surface shape of rice seeds based on the point cloud obtained by the three-dimensional laser scanning system to further improve the accuracy of the classification of rice varieties. Based

Received date: 2020-05-13 **Accepted date:** 2021-06-13

Biographies: Qianjin Xu, Bachelor, research interest: agricultural engineering, Email: xuqianjinnjau@163.com; Yingying Yang, Bachelor, research interest: agricultural engineering, Email: YangYingYingNJAU@163.com; Hu Lu, Bachelor, research interest: agricultural engineering, Email: luhu849765459@163.com; Hua Li, PhD, Professor, research interest: agricultural engineering, Email: lihua@njau.edu.cn; Xuebin Feng, PhD, Associate Professor, research interest: agricultural engineering, Email: fxb9510@njau.edu.cn; Wenqing Yin, PhD, Professor, research interest: agricultural engineering, Email: yinwq@njau.edu.cn.

***Corresponding author:** Yan Qian, PhD, Associate Professor, research interest: agricultural engineering, machine vision and three-dimensional reconstruction. College of Artificial Intelligence, Nanjing Agriculture University, Nanjing 210031, China. Tel: +86-25-58606585, Email: qianyan@njau.edu.cn.

on the team's previous research, Feng et al.^[10] built a 3D laser scanning system to collect rice seed point clouds, extracted nine three-dimensional morphological surface features and nine cross-sectional projection features to input into the BP neural network for variety recognition experiments, and the average recognition accuracy reached 97%.

As a branch of machine learning, deep learning has made groundbreaking progress in many types of applications in recent years. A deep learning network transforms the features of the input signal into a new feature space through layer-by-layer feature transformation of the input data, and it can automatically learn a hierarchical feature representation, which can effectively classify the data^[11]. However, the convolutional structure that often appears in deep learning networks requires regularized data structures as input. Since point cloud data structure is a type of irregular data structure, regularized conversions are required before input to the network, and such conversions often cause problems of structural loss and resolution de-gradation^[12]. The proposal of the PointNet^[13] network solves this problem well, for the disordered point cloud data, the maximum pool function is used in the network structure to realize the direct use of the point cloud as the input of the network for classification and segmentation tasks. Ma et al.^[14] modified the PointNet network structure for the study of 3D hand posture estimation, introduced a jump structure to recombine features at different levels, and achieved an actual operating speed of 54 frames/s on the NYU hand posture dataset. Zhao et al.^[15] proposed a method of combining local features based on point extraction with global features based on original point cloud extraction to classify LiDAR point cloud features, which solved the problem of weak local feature extraction capabilities of PointNet and achieved better results than the classification results of PointNet.

The existing three-dimensional data-based rice variety recognition methods use traditional neural networks as the classification model, and the recognition effect is usually limited by the number of extracted rice characteristics. This study proposes a rice variety recognition method based on an improved PointNet model, eliminating the need for an artificially designed feature extraction process to obtain faster recognition speed and higher recognition accuracy.

2 Materials and methods

2.1 Sample preparation

Eight varieties of rice seeds and 210 samples of each variety were prepared, including Fengkang 30, Huajing 7, Tianyou 673, Zajiaodao, Liannuo 1, Nanjing 9108, Guiyu 11 and Liandao 6. To ensure the diversity of sample sources, these rice seed samples came from various regions in China. Liannuo 1, Huajing 7 and Nanjing 9108 were from Jiangsu Province; Tianyou 673 was from Fujian Province; Liandao 6 was from Heilongjiang Province; Guiyu 11 was from Guangxi Province. Some of the samples used in this work are shown in Figure 1.

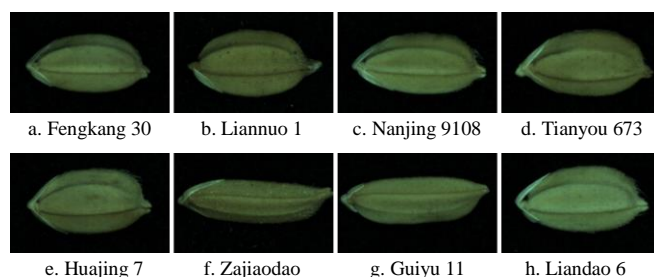


Figure 1 Images of different varieties rice seed samples

2.2 Point cloud collection system

The point cloud collection in this study was obtained with a 3D point cloud collection system. The 3D point cloud collection system was primarily composed of a light field camera (R42) manufactured by Raytrix and a high-speed GPU (NVIDIA GTX 1080). The light field camera was a focused plenoptic camera with 41.5 Megarays and a resolution of 7708×5352 pixels. The imaging lens is a 3D light field lens with a focal length of 50 mm and an aperture of $f/2.80$. High-speed GPUs were used for light field processing. The details of the camera model and depth estimation theory can be found in an article by Johannsen et al.^[16] The 3D light field camera was installed on the vision platform for support, and a ring light source with adjustable brightness was installed between the light field camera and the base of the vision platform. The process of point cloud collection was performed in a dark room. The structure of the experimental devices is shown in Figure 2.



Figure 2 Point cloud data collection system

2.3 Dataset generation method

2.3.1 Collection and preprocessing of rice seed surface point cloud data

Affected by the device accuracy, camera resolution, environmental factors, and operator experience, the collected point cloud may contain noise and voids. Preprocessing was performed using RxLive4.0 software. In the experiments, for filter processing, a bilateral filter was used, and the filter radius was set to 20 pixels. The 'standard' fill algorithm was used for fill processing, and the number of iterations was set to 16 with a lookup distance of 10 pixels. For the smoothing processing, the edge smoothing factor was set to 0.100.

The Raytrix R42 light field camera has not only the ability to record three-dimensional information but also the ability to record color information. Therefore, the RxLive4.0 processing software provided with the camera can export files in multiple formats. In this experiment, the data in point cloud format were selected for processing. Open a file in ply format with Mashlab software to display a three-dimensional model of a rice seed after preprocessing. This format file records both three-dimensional information and color information, as shown in Figure 3.



Figure 3 Preprocessed rice seed model

2.3.2 Point cloud segmentation

In this work, the original point cloud data were collected by a focused plenoptic camera, including the platform base and target rice seed. To obtain the target rice seed point cloud, the data of the platform base must be removed. The random sample consensus (RANSAC) algorithm^[17] was used to calculate the parameters of the plain of the platform base. Then, the calculated parameters were used to obtain the point cloud of the plain and remove it. Then, the segmented target rice seed point cloud was stored for further processing. The point cloud segmentation results are shown in Figure 4.

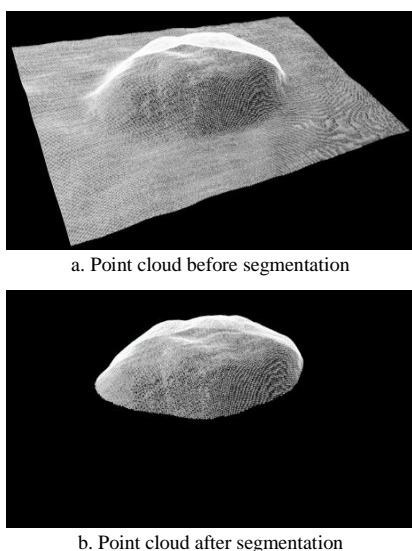


Figure 4 Point cloud segmentation results

2.3.3 Point cloud downsampling

The focused plenoptic camera has a high resolution and can capture very detailed features of objects. However, for the classification of rice seeds, it captures a large amount of redundant data. Storing, processing, and displaying these point cloud data will increase the processing load of the computer, occupying more computer resources and reducing the amount of point cloud data storage and operational efficiency.

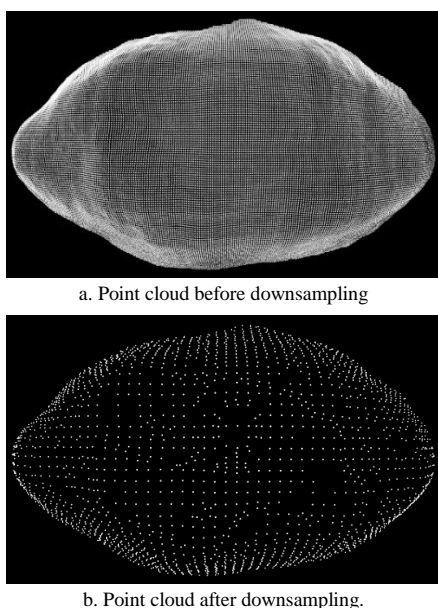


Figure 5 Downsampling of the point cloud

The voxel-based grid^[18] method and random sampling method were used to process the downsampling of the point cloud. First, a minimal three-dimensional voxel grid was created based on a point

cloud bounding box. Then, the voxel grid was divided into $m \times n \times l$ small grids with side length L . In each small grid, all the points were replaced by their centroids. The number of point cloud points was reduced to nearly, but more than, 2048 points. Finally, a random sampling method was used to reduce the number of point clouds to 2048 points specifically. The downsampled point cloud preserves the structural information of rice seeds well and maintains a clear edge contour, as shown in Figure 5.

2.3.4 Point cloud translation and normalization

To ensure that the similarity measure of the shape is not affected by the translation and scaling of the point cloud, point cloud translation and normalization are important preprocessing steps^[19,20].

To ensure translation invariance, the center of the mass of the rice seed point cloud needs to be translated to the position of the coordinate origin. To ensure the invariance of the scale size, the rice seed point cloud after the translation transformation also needs to be normalized to the standard cell size.

1) Translation. Input the rice seed point set $P = \{p_i | i = 1, 2, 3, \dots, n\}$, where p_i is the i -th point of the point cloud P . The number of point sets is n . First, calculate the center of the rice seed by Equation (1).

$$\begin{cases} x_{mid} = \frac{x_{max} + x_{min}}{2} \\ y_{mid} = \frac{y_{max} + y_{min}}{2} \\ z_{mid} = \frac{z_{max} + z_{min}}{2} \end{cases} \quad (1)$$

where, x_{max} , y_{max} and z_{max} are the maximum values among all points; x_{min} , y_{min} and z_{min} are the minimum values. Then, move the origin of the coordinate system of the point cloud to $p_{center}(x_{mid}, y_{mid}, z_{mid})$ to form the new point set $P' = \{p'_i | p'_i = p_i - p_{center}, p_i \in P\}$.

2) Point cloud normalization. First, calculate the scale of the transformation

$$l_{scale} = x_{max} - x_{min} \quad (2)$$

With the point set P' obtained in the previous step, compress the point cloud coordinates to between -1 and 1 , forming the last point set $P'' = \{p''_i | p''_i = p'_i / l_{scale}, p'_i \in P'\}$.

2.4 Classification models

2.4.1 Basic PointNet architecture

Since it is affected by the acquisition facilities and the spatial coordinate system, the order of point cloud data should be very different when objects are scanned at different facilities or different locations^[21]. To make a model invariant to the input set in terms of the data feeding order, PointNet applies a symmetric equation to the transformed elements in the set as follows:

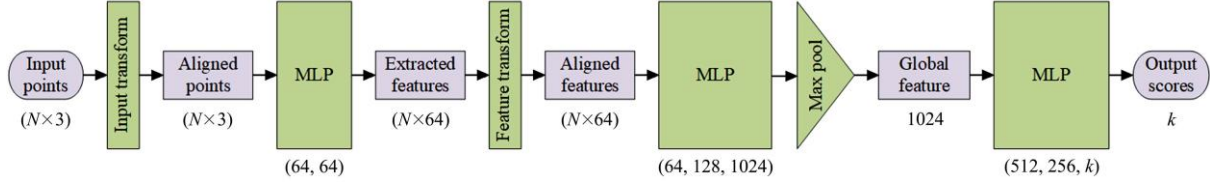
$$f(\{x_1, \dots, x_n\}) \approx g(h(x_1), \dots, h(x_n)) \quad (3)$$

where, $f: 2^{R^N} \rightarrow R$, $h: R^N \rightarrow R^K$ and $g: R^{nK} \rightarrow R$ is a symmetric equation.

The equation h is approximated by a multilayer perceptron network, and the function g is approximated by the composition of a single-variable function and a max-pooling function. Through the collection of h , the network can learn a number of f s to capture different properties of the set.

The classification network architecture of the basic PointNet is shown in Figure 6. The network input is the 3D coordinates ($N \times 3$) of a 3D point cloud containing N points. First, a mini-network (T-net) is used to predict the 3×3 affine transformation matrix, and this transformation is directly applied to the coordinates of the input set to obtain an aligned $N \times 3$ input set. This mini-network consists

of the basic modules of point-independent feature extraction, a max-pooling layer and a fully connected layer, and its scale is close to that of a large network. The aligned $N \times 3$ point set is extracted by a shared-parameter Multi-Layer Perceptron (MLP) model (64, 64) to obtain $N \times 64$ features. Then, through the feature transformation matrix prediction network T-net (64), the 64×64 transformation matrix is predicted to act on the extracted $N \times 64$ features to achieve



Note: MLP: Multi-Layer Perceptron, the same as below.

Figure 6 Basic PointNet classification network architecture

2.4.2 Improved PointNet architecture

This study proposes an improved PointNet network model for the task of rice variety recognition. To improve the feature extraction capability of the network, a cross-level feature connection structure was added to the original PointNet network structure, the features of different levels were merged, and the features of different levels were fully utilized to improve the classification performance of the network.

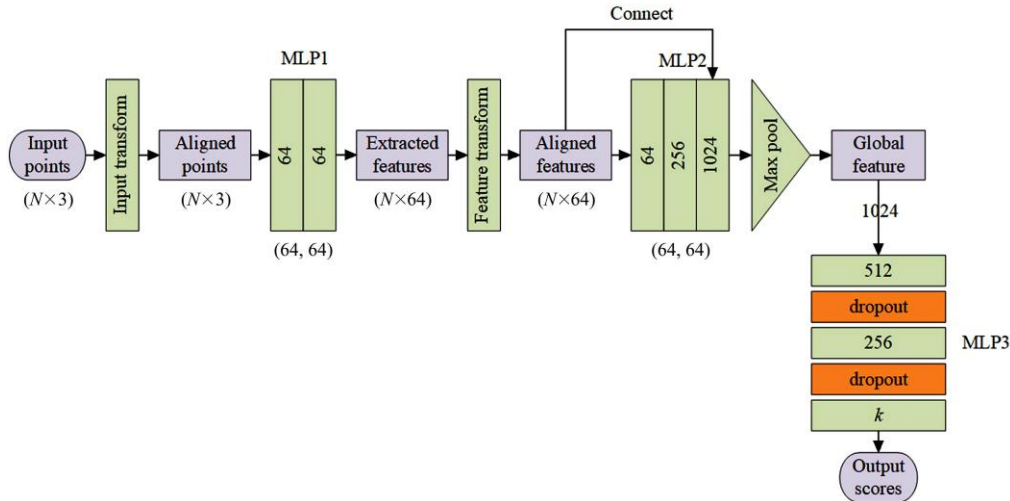


Figure 7 Improved PointNet network architecture

In addition to the input layer, the network structure has a total of 9 layers of networks, an input transformation small network T-net (3) and a feature alignment network T-net (64). The input transformation network and the feature alignment network are small networks in the original PointNet network model. T-net (3) aligns the point cloud by predicting a 3×3 dimensional affine transformation matrix and multiplying it with the original input point cloud. T-net (64) predicts the alignment features of the 64×64 -dimensional transform matrix in the same way. The input of the network is an $N \times 3$ point cloud, where N is the number of points in the point cloud, and each point is composed of three-dimensional coordinates (x, y, z) .

The operation process of the improved PointNet network is as follows:

- 1) The input transformation small network aligns the original input point cloud ($N \times 3$) to obtain an aligned point cloud ($N \times 3$);
- 2) Perform feature extraction on the point cloud through the first multilayer perceptron MLP1. MLP1 is composed of two convolutional layers with 64 channels, and finally, $N \times 64$ -dimensional point features are obtained;

feature alignment. Then, three layers of MLP (64, 128, 1024) are used to further extract the aligned $N \times 64$ features to obtain $N \times 1024$ features. The max-pooling layer aggregates the extracted feature vectors of the N points into 1024-dimensional global feature vectors that do not change the arrangement of the input points. Finally, a three-layer MLP network is used to map the 1024-dimensional global feature vector to a k -dimensional vector output.

The improved PointNet and the original PointNet model were compared as follows: 1) The improved PointNet model retained the input transformation and feature transformation structure in the original PointNet model. 2) A feature connection structure was added to the improved PointNet model. 3) The number of convolution channels of some convolution layers in the improved PointNet model had been changed. The improved PointNet network architecture is shown in Figure 7.

- 3) The feature alignment network aligns the $N \times 64$ -dimensional features extracted in the second step to obtain aligned $N \times 64$ -dimensional point features;

- 4) In the second multilayer perceptron MLP2, the first two convolutional layers transform the $N \times 64$ dimensional features obtained in the previous step. The channel numbers of the first two convolutional layers are 64 and 256, that is, $N \times 256$ -dimensional features;

- 5) Connect the $N \times 256$ -dimensional feature obtained in the third step with the $N \times 64$ -dimensional feature obtained in the second step through a cross-layer connection structure to obtain an $N \times 320$ -dimensional feature;

- 6) The third convolutional layer in MLP2 transforms the $N \times 320$ -dimensional features obtained in the previous step to $N \times 1024$ -dimensional features, and the number of channels is 1024;

- 7) The max-pooling layer aggregates the $N \times 1024$ dimensional features obtained in the previous step into 1024 dimensional global features;

- 8) The 1024-dimensional global feature vector is reduced layer by layer through MLP3. MLP3 is composed of 3 fully connected

layers with nodes 521, 256 and k , and a layer of dropout structure is added between each pair of fully connected layers. The vector $k \times 1$ output by the last fully connected layer is the classification result.

The rectified linear unit (ReLU) function was used as the activation function, which solves the problem of vanishing gradients during training and decreases the occurrence of overfitting while speeding up the convergence process, as shown in Equation (4).

$$\text{ReLU}(x) = \begin{cases} 0, & x \leq 0 \\ x, & x > 0 \end{cases} \quad (4)$$

where, x represents the feature values of the neurons. Two dropout layers were added behind the first and second fully connected layers to prevent the network from overfitting^[22]. The dropout rates were both set to 0.7, which represents the probability of dropping neurons. The size and parameter calculation of each layer of the seven-layer network are shown in Table 1, Conv represents the convolutional layer, max-pooling is the max-pooling layer, FC is the fully connected layer, and the fully connected layer has no core size or step size parameters. N corresponds to the number of input point cloud points, and k is the number of classes.

Table 1 Sizes and parameters of each layer in the network model

No.	Layer	Kernel size	Stride	Output dimension	Number of parameters
1	Conv1	1×3	1×1	$N \times 64$	256
2	Conv2	1×1	1×1	$N \times 64$	4160
3	Conv3	1×1	1×1	$N \times 64$	4160
4	Conv4	1×1	1×1	$N \times 256$	16 640
5	Conv5	1×1	1×1	$N \times 1024$	328 704
6	Max-pooling	$N \times 1$	1×1	1024×1	1024
7	FC1			512×1	524 800
8	FC2			256×1	131 328
9	Output			$k \times 1$	257k

Note: Conv represents the convolutional layer; Max-pooling is the max-pooling layer; FC is the fully connected layer; the fully connected layer has no core size or step size parameters. N corresponds to the number of input point cloud points; k is the number of classes, the same as below.

2.4.3 Network optimization

The improved PointNet network was optimized from two aspects: changing the number of convolutional layer channels and network hyperparameters. The convolution structure calculated the original data, learned the characteristics of the input data, and combined the multilayer convolutional network to realize the layer-by-layer transformation of the features to obtain high-dimensional features. It generally has three parameters: the size of the convolution kernel, the step length of the convolution and the number of channels. The number of channels in the convolutional layer greatly affects the feature extraction ability of the network. Changing the number of channels in the convolutional layer can improve the feature extraction ability of the network. The optimized hyperparameters are batch size and learning rate: batch size refers to the number of point clouds trained in each learning batch during the network training process; learning rate refers to the amplitude of each parameter update during the training process.

2.5 Software tools

During the dataset generation process, point cloud library tools^[23] and the Visual Studio 2015 platform were used to implement point cloud segmentation, rotation normalization, and point cloud downsampling. Python 3.7 and the TensorFlow framework were used to build the network model, which ran in the

central processing unit (CPU). All software operations were based on a Windows 10 64-bit operating system with an Intel (R) Core(TM) i5-7200U CPU and 8 GB RAM.

3 Results and discussion

3.1 Experimental environment

The network training and testing environment configuration: Processor: Intel(R) Core(TM) i5-7200U; Memory: 8 GB; Operating system: Windows 10 64-bit; Programming platform: Python 3.7; Deep learning framework: TensorFlow1.14.0.

3.2 Dataset

In the experiment, each type of rice seed point cloud dataset was divided into a training set and a test set at a ratio of 5:2, as shown in Table 2. A total of eight varieties of rice seed point clouds were collected, there were 210 samples of each variety and 1680 samples in total. Among them, 150 point clouds were randomly selected from each group of 210 point clouds, and a total of 1200 point clouds were used as training samples. The remaining 60 point clouds of each rice seed were used as test samples, and there was a total of 480 test samples.

Table 2 Rice seed point cloud dataset

No.	Type	Number of training	Number of tests
1	Fengkang 30	150	60
2	Huajing 7	150	60
3	Tianyou 673	150	60
4	Zajiaodao	150	60
5	Liannuo 1	150	60
6	Nanjing 9108	150	60
7	Guiyu 11	150	60
8	Liandao 1	150	60

3.3 Training process

3.3.1 Effects of batch size on model performance

When the batch size is set to 8, 16, and 32, the training curve of the model is shown in Figure 8. The abscissa is the number of iterations in the model training process, and the ordinate is the loss value. It can be seen from Figure 8 that the training curves of the three batch sizes have very obvious differences, indicating that the batch size has a greater impact on network performance. As the batch size increases, the loss value convergence speed also increases. When the batch size is 32, the loss value converges fastest, and the training loss value is the smallest. Therefore, the improved PointNet model finally selected a batch size of 32 for training.

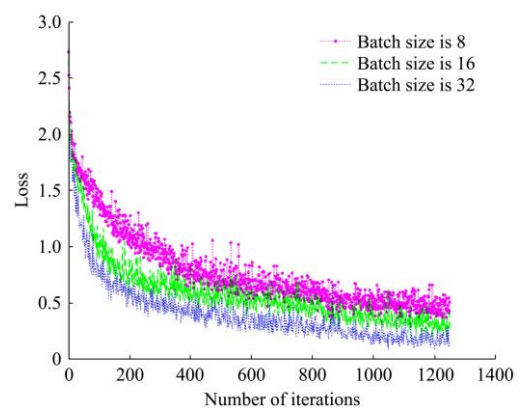


Figure 8 Influence of batch size on the training effect of the improved PointNet model

3.3.2 Effects of learning rate on model performance

When the learning rate is set to 0.01, 0.001, and 0.0001, the

training curve of the model is shown in Figure 9. The abscissa is also the number of iterations in the model training process, and the ordinate is the loss value. The convergence speed of the training loss value when the learning rate is 0.001 and 0.0001 is faster than the speed when the learning rate is 0.01. As the learning rate decreases, the convergence speed of the loss value accelerates, and the training rate is 0.001 and 0.0001. The latter half of the curve is closer and has little effect on the further convergence of the loss value. To ensure the performance of the model and make the model train faster, the learning rate of the improved PointNet model is set to 0.001.

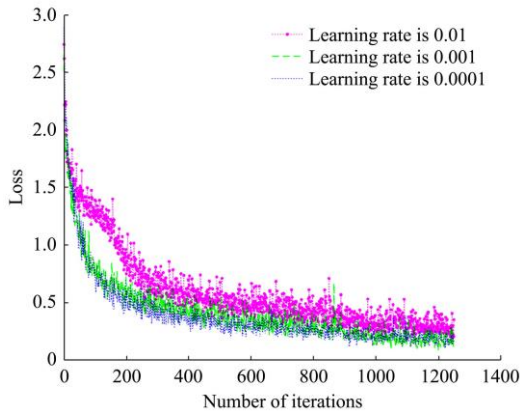


Figure 9 Influence of the learning rate on the training effect of the improved PointNet model

3.3.3 Influence of the number of convolutional layer channels on model performance

To study the influence of the number of convolutional layer channels on network performance, this study set up 4 combinations of different convolutional layer channel numbers for the improved PointNet model MLP2, as shown in Table 3. The number of channels of the convolutional layer of the improved PointNet model was set according to these four combinations, and the model was trained.

Figure 10 shows the training process that the curves of the 4 network models are relatively steep in the first 200 iterations, indicating that the loss value converges quickly. After 200

iterations, the curve gradually stabilizes, and the loss value slowly converges and gradually stabilizes. In comparison, the loss curves of the four models with different numbers of convolutional layer channels are relatively close, and the network performance gap is small.

Table 3 Combinations of the number of channels in the convolutional layer of MLP2 of the improved PointNet model

No.	Conv3	Conv4	Conv5
1	256	512	1024
2	128	256	1024
3	64	128	1024
4	64	256	1024

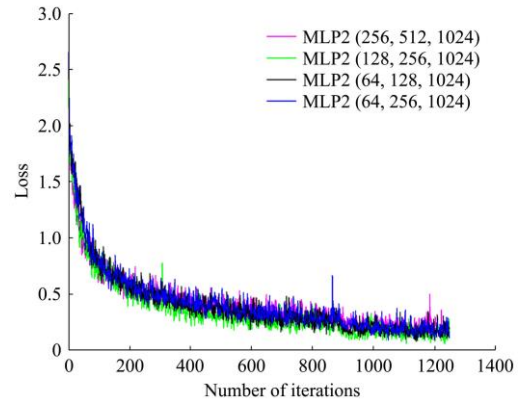


Figure 10 Influence of the number of convolutional layer channels on the training effect of the improved PointNet model

The above four models were tested on the test set, and the results are shown in Table 4. The results show that when the number of channels of the convolutional layer of MLP2 is set to 64, 256 and 1024, the improved PointNet model has a recognition rate of more than 90% for four rice varieties, three varieties above 80%, and only one variety below 80%. The average recognition rate is the highest among the four combinations, and the classification effect is the best. Therefore, the number of channels in the convolutional layer in MLP2 of the finally improved PointNet model is set according to Combination 4.

Table 4 Variety recognition results of improved PointNet models with different combinations of MLP2 convolutional layer channels

No.	Fengkang 30	Zajiaodao	Tianyou 673	Huajing 7	Liannuo 1	Nanjing 9108	Guiyu11	Liandao 1	Average accuracy
1	98.3%	98.3%	63.3%	90.0%	80.0%	78.3%	91.7%	78.3%	84.8%
2	98.3%	98.3%	76.7%	76.7%	90.0%	93.3%	93.3%	68.3%	86.9%
3	98.3%	98.3%	83.3%	70.0%	88.3%	90.0%	93.3%	86.7%	88.5%
4	98.3%	98.3%	83.3%	86.7%	86.7%	91.7%	95.0%	75.0%	89.4%

3.4 Classification results

The improved PointNet model is tested for the classification of rice varieties. The test set contains eight varieties of rice, 60

samples of each variety, and a total of 480 rice seed point cloud samples. The classification results are compared with the PointNet, PointNet++^[24] and DGCNN^[25] models, as shown in Table 5.

Table 5 Classification results of the eight rice varieties based on different models

Models	Fengkang 30	Zajiaodao	Tianyou 673	Huajing 7	Liannuo 1	Nanjing 9108	Guiyu11	Liandao 1	Average accuracy
Improved PointNet	98.3%	98.3%	83.3%	86.7%	86.7%	91.7%	95.0%	75.0%	89.4%
PointNet	98.2%	98.2%	73.2%	85.7%	78.6%	89.3%	94.6%	87.5%	88.2%
PointNet++	100%	93.3%	66.7%	85.0%	80.0%	91.7%	98.3%	96.7%	89.0%
DGCNN	98.3%	38.3%	16.7%	98.3%	93.3%	70.0%	98.3%	75.0%	73.5%

Comparing the classification results of rice varieties, it can be seen that:

1) The improved PointNet model was better than the PointNet model in the classification of rice varieties. Compared with the

PointNet model, the improved PointNet model had higher recognition accuracy for seven rice varieties except Liandao 1. In particular, the recognition rate of the improved PointNet model of Tianyou 673 increased from 73.2% to 83.3%, and the recognition rate of

Liannuo 1 increased from 78.6% to 86.7%, a huge improvement. The average accuracy rate was 1.2% higher than that of PointNet.

2) The performance of the improved PointNet model in the classification of rice varieties was similar to that of PointNet++. The improved PointNet model had higher recognition rates for Zajiaodao, Tianyou 673, Huajing 7 and Liannuo 1 than PointNet++, but PointNet++ had higher recognition rates for Fengkang 30, Guiyu 11 and Liandao 1. The average classification accuracies of the two models were close to each other.

3) The classification accuracy of the DGCNN model for Huajing 7, Liannuo 1 and Guiyu 11 was the highest among the four models, while the classification accuracy for Zajiaodao, Tianyou 673 and Liandao 1 were lower. Among the four models, the average classification accuracy of rice seeds was the lowest.

The above results showed that the performance of the improved PointNet model proposed in this study was better than that of PointNet and DGCNN, comparable to PointNet++, and could be used in rice variety classification tasks with three-dimensional point cloud data as input.

4 Conclusions

This study classifies rice varieties based on 3D point cloud data and deep learning algorithms. In the experiment, a Raytrix light field camera was used to collect the original 3D point cloud data of eight varieties of rice seeds. The 3D model of the rice seeds obtained after preprocessing with RxLive software completely and accurately expresses the shape characteristics of the rice seeds. The construction method of a three-dimensional point cloud dataset for rice variety identification was studied, including point cloud segmentation, downsampling, translation and normalization processing algorithms. An improved PointNet model was proposed: by adding a cross-level feature connection structure to PointNet, low-level features and high-level features were connected, feature fusion was realized, and the utilization of features by the network was improved. Finally, testing on the test set shows that the improved PointNet model had an accuracy rate of 1.2% higher than that of the PointNet model, which was 89.4%, and the accuracy rate of the seven varieties was higher than that of the PointNet. Compared with PointNet++ and DGCNN, the average classification accuracy of the improved PointNet model was also higher. Regarding future work, it is planned to expand the dataset to achieve more classification and identification of rice varieties and to ensure the accuracy of the classification.

Acknowledgements

This work was financially supported by the National Natural Science Foundation of China Youth Fund Project (Grant No. 51305182) and the Ministry of Agriculture Key Laboratory of Modern Agricultural Equipment (Grant No. 201602004).

[References]

- Qiu Z, Chen J, Zhao Y, Zhu S, He Y, Zhang C. Variety identification of single rice seed using hyperspectral imaging combined with convolutional neural network. *Applied Sciences*, 2018; 8(2): 212. doi: 10.3390/app8020212.
- Li X H, Ma X, Li Z H, Deng X W, Li H W. Identification of rice variety based on multi-feature fusion and SVM. *Journal of Chinese Agricultural Mechanization*, 2019; 40(7): 97–102. (in Chinese)
- Weng S Z, Tang P P, Yuan H C, Guo B Q, Yu S, Huang L S, et al. Hyperspectral imaging for accurate determination of rice variety using a deep learning network with multi-feature fusion. *Spectrochimica Acta Part A, Molecular and Biomolecular Spectroscopy*, 2020; 234: 118237. doi: 10.1016/j.saa.2020.118237.
- Kuo T Y, Chung C L, Chen S Y, Lin H A, Kuo Y F. Identifying rice grains using image analysis and sparse-representation-based classification. *Computers and Electronics in Agriculture*, 2016; 127: 716–725.
- Golpour I, Parian J A, Chayjan R A. Identification and classification of bulk paddy, brown, and white rice cultivars with colour features extraction using image analysis and neural network. *Czech Journal of Food Sciences*, 2018; 32(3): 280–287.
- Mittal S, Dutta M, Issac A. Non-destructive image processing based system for assessment of rice quality and defects for classification according to inferred commercial value. *Measurement*, 2019; 148: 106969. doi: 10.1016/j.measurement.2019.106969.
- Fabiyi S D, Vu H, Tachtatzis C, Murray P, Harle D, Dao T, et al. Varietal classification of rice seeds using RGB and hyperspectral images. *IEEE Access*, 2020; 8: 22493–22505.
- Qian Y, Yin W Q, Lin X Z, Ding Y Q, Feng X B. Variety identification of rice seed based on three-dimensional reconstruction method of sequence images. *Transactions of the CSAE*, 2014; 30(7): 190–196. (in Chinese)
- Li H, Qian Y, Cao P, Yin W Q, Dai F, Hu F, et al. Calculation method of surface shape feature of rice seed based on point cloud. *Computers and Electronics in Agriculture*, 2017; 142(Part A): 416–423.
- Feng X B, He P J, Zhang H X, Yin W Q, Qian Y, Cao P, et al. Rice seeds identification based on back propagation neural network model. *Int J Agric & Biol Eng*, 2019; 12(6): 122–128.
- Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets. *Neural Computation*, 2006; 18(7): 1527–1554.
- Niu C G, Liu Y J, Li Z M, Li H. 3D object recognition and model segmentation based on point cloud data. *Journal of Graphics*, 2019; 40(2): 274–281. (in Chinese)
- Charles R Q, Su H, Kaichun M, Guibas L J. PointNet: Deep learning on point sets for 3d classification and segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA: IEEE, 2017; pp.77–85. doi: 10.1109/CVPR.2017.16.
- Ma L, Jin S S, Niu B. 3D hand pose estimation method based on improved PointNet. *Application Research of Computers*, 2020; 37(10): 3188–3192. (in Chinese)
- Zhao Z Y, Cheng Y L, Shi X S, Qin X X, Li X. Terrain classification of LiDAR point cloud based on multi-scale features and PointNet. *Laser & Optoelectronics Progress*, 2019; 56(5): 251–258. (in Chinese)
- Johannsen O, Heinze C, Goldluecke B, Perwaß C. On the calibration of focused plenoptic cameras. In: *Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications. Lecture Notes in Computer Science*, Berlin: Springer, 2013; 8200; 302–317. doi: 10.1007/978-3-642-44964-2_15.
- Schnabel R, Klein W R. Efficient RANSAC for point-cloud shape detection. *Computer Graphics Forum*, 2007; 26(2): 214–226.
- Dong H W. Study for cell grid methods finding k nearest neighbors. *Computer Engineering and Applications*, 2007; 43(21): 52–56. (in Chinese)
- Zheng B C, Peng W, Zhang Y, Ye X Z, Zhang S Y. A survey on 3D model retrieval techniques. *Journal of Computer - Aided Design & Computer Graphics*, 2004; 16(7): 873–881. (in Chinese)
- Yang Y B, Lin H, Zhu Q. Content-based 3D model retrieval: A survey. *Chinese Journal of Computers*, 2008; 27(10): 1297–1310. (in Chinese)
- Armeni I, Sener O, Zamir A R, Jiang H L, Brilakis I, Fischer M, et al. 3D semantic parsing of large-scale indoor spaces. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE, 2016; pp.1534–1543. doi: 10.1109/CVPR.2016.170.
- Srivastava N, Hinton G, Krizhevsky A, Sutskever L, Salakhutdinov R. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 2014; 15(1): 1929–1958.
- Rusu R B, Cousins S. 3D is here: Point cloud library (PCL). In: *IEEE International Conference on Robotics and Automation*. Shanghai, China: IEEE, 2011; pp.1–4. doi: 10.1109/ICRA.2011.5980567.
- Qi C R, Yi L, Su H, Guibas L J. PointNet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv*, 2017; arXiv:1706.02413v1.
- Phan A V, Nguyen M L, Nguyen Y L H, Bui L T. DGCNN: A convolutional neural network over large-scale labeled graphs. *Neural Networks*, 2018; 108: 533–543.