

Individual nursery trees classification and segmentation using a point cloud-based neural network with dense connection pattern[☆]

Jie Xu, Hui Liu^{*}, Yue Shen, Xiao Zeng, Xinpeng Zheng

School of Electrical and Information Engineering, Jiangsu University, Zhenjiang 212000, China



ARTICLE INFO

Keywords:

Nursery tree species classification
Crowns and trunks segmentation
Point clouds
Neural network models,

ABSTRACT

Nurseries are used to cultivate a variety of tree species. Obtaining some specific information like the tree species, positions of crowns and trunks can enhance the efficacy of nursery management. Due to the robustness to illumination, the point cloud-based neural network models have become extensively employed in segmenting and classifying individual trees from large-scale data. However, few studies have focused on further processing the point clouds of individual trees. Therefore, D-PointNet++ (Dense PointNet++) is proposed in this paper for classifying tree species and segmenting different parts of trees (crowns, trunks, pots and supporting poles). D-PointNet++ utilizes a dense connection pattern in the feature extraction module, inspired by the architecture of DenseNet. Additionally, the proposed model uses a gating system and concatenation as fusion operations to combine point cloud features with different dimensions to improve accuracy. The point cloud data of seven different types of garden trees in the nursery was collected using a laser sensor. The experimental results demonstrate that D-PointNet++ surpasses two representative baseline methods, PointNet and PointNet++, in terms of both classification and segmentation accuracy. For the self-made nursery dataset, the classification overall accuracy (OA) and class accuracy (mAcc) of D-PointNet++ can reach 92.65% and 92.54%; the average Intersection over Union (mIoU) and mAcc can reach 89.90% and 92.18%, respectively. The proposed D-PointNet++ can provide more accurate information on each tree and is beneficial to the management of the nursery.

1. Introduction

Nurseries are utilized for the cultivation of diverse, high-quality trees to fulfill the requirements of various planting projects, holding significant economic value (Bruno J.L et al., 2022; Fabrfci et al., 2011; María et al., 2018). However, manual management and monitoring of tree growth in large nurseries can be labor-intensive and time-consuming. To ensure the production of high-quality trees, the implementation of advanced technologies becomes necessary for efficient management and monitoring of tree growth in nurseries. One such approach involves utilizing classification or segmentation technologies to get the categories of diverse tree species (Liu, 2022; Liu et al., 2021), detailed information on the crowns and trunks to accurately monitor the growth and development of trees (Andreas et al., 2022; Dersch et al., 2023), as well as implement targeted management strategies such as variable rate spraying or other maintenance operations. Moreover, incorporating such technologies into the nursery management system can lead to

significant reduction in labor costs while raising operational efficiency.

In recent years, the field of computer vision has witnessed significant advancements in traditional machine learning techniques as well as neural network models. Traditional machine learning methods mainly rely on manual features to complete related tasks (Herbert et al., 2008; Surasak et al., 2018), which have strong adaptability and require less computational time (Chen et al., 2022; Yang et al., 2019). Traditional methods may struggle with obtaining expressive representations and capturing complex relationships, resulting in poorer performance for challenging tasks. On the other hand, neural network models excel in learning more expressive features from rich data, allowing them to achieve superior results (Yutaro et al., 2020; Zhou et al., 2023).

Currently, the relevant network models can be mainly classified into two categories: image-based (Alexey et al., 2020; Ross et al., 2014) and point cloud-based models (Charles et al., 2017; Feng et al., 2018; Maturana and Scherer, 2015; Qi et al., 2016; 2017). Point clouds consist primarily of three-dimensional points that contain coordinate

[☆] This work was supported by the National Natural Science Foundation of China under Grant 32171908.

* Corresponding author at: School of Electrical and Information Engineering, Jiangsu University, Zhenjiang 212000, China

E-mail address: amity@ujs.edu.cn (H. Liu).

information (x, y, z). Unlike image-based models, point cloud-based models are not affected by various lighting conditions. Therefore, these kinds of neural network models have gained popularity in outdoor scenes like nurseries and orchards due to their illumination-invariant characteristics (Yang et al., 2022a; 2022b; Zhang et al., 2021). Presently, many point cloud-based neural network models have been extensively employed in various 3D tasks in the field of agriculture and forestry, such as classification, segmentation, registration, and other aspects (Chen et al., 2021; Kristina et al., 2015; Yuan et al., 2022).

Traditional 2D image-based convolutional networks face challenges when applied directly to point clouds due to their distinct characteristics of being disordered, irregular, and unstructured. To overcome these limitations, earlier researchers utilized conversion techniques such as projecting points into multiple 2D images from varied angles, or replacing points with voxels or grids (like MVCNN (Qi et al., 2016), VoxNet (Maturana and Scherer, 2015), GVCNN (Feng et al., 2018)). Subsequently, convolutional neural networks perform feature extraction operations on multiple images, voxels, or grids. Nevertheless, these methods often necessitate extra storage and computation time due to projection or voxelization operations, which can lead to critical information loss and destruction of the original 3D structure of the point clouds.

The emergence of PointNet in 2017 makes a significant breakthrough in the field of deep learning based on point clouds. PointNet (Charles et al., 2017) utilizes globally shared multilayer perceptrons (MLP) to directly extract features from point clouds. It also selects max pooling as the symmetry function and designs joint alignment networks to tackle the disordered nature of points. In comparison to projection-based or voxel-based neural networks, PointNet is more efficient because it does not require additional processing steps and preserves the original characteristics of point clouds. Based on PointNet, PointNet++ (Qi et al., 2017) extracts the local features by obtaining the relationships between the center point and its corresponding neighboring points in each local region. Many experimental results have demonstrated that local features can significantly enhance the classification or segmentation performance of the model, with PointNet++ demonstrating notably better performance than PointNet.

Recently, any researchers employ some specific methods to realize tree species classification and segmentation tasks using forest point cloud data. For example, Aada et al. (2023) proposes a method called layer-by-layer, which can realize individual tree detection and segmentation tasks from a group of dense point cloud data. According to the structural characteristics of trees, Liu et al. (2021) designs a neural network model called LayerNet to classify point clouds of different tree species collecting from the forest regions. Xu et al. (2020) combines images with point clouds to classify the individual tree species in natural forests. However, all above methods mainly realize the segmentation and classification tasks of the individual tree point clouds from a large-scale of point clouds. In practical applications, acquiring detailed information about each tree, such as species, crowns, and trunks, is critical for agricultural robots to facilitate tasks like management, navigation, and variable spraying. Hence, the main objective of this paper is to primarily concentrate on classifying individual tree species and segmenting the corresponding crowns, trunks, and other relevant parts.

This paper introduces a novel point-based network called D-PointNet++ (Dense PointNet++), which aims to achieve the identification of common tree species, as well as segmentation of tree parts such as crowns, trunks, and other relevant components. Compared with classic PointNet or PointNet++, the proposed neural network model can extract more valuable information from point clouds and improve the accuracy of results. Inspired by dense blocks in DenseNet (Huang et al., 2017), the connection structure of PointNet layers in PointNet++ is modified, and a new feature extraction layer called dense PointNet layer has been developed. Furthermore, to enhance the performance accuracy for both classification and segmentation tasks, D-PointNet++ concatenates global and local features from various set abstraction levels of

point clouds. This feature aggregation process is accomplished through a gated fusion operation. To perform the classification and segmentation tasks, point clouds were collected using Livox Horizon scanner from seven types of landscape trees in a nursery. The collected point clouds were then processed to create datasets for experiments. The results of classification and segmentation are compared with those of PointNet or PointNet++, demonstrating the highest performance of our proposed neural network.

2. Materials and methods

2.1. Data acquisition and processing

The experimental site selected for this study was a nursery located at coordinates 32.12N, 119.31E in Zhenjiang, China. The nursery provides a variety of common landscape tree species to support the implementation of projects such as the scenic area planning in the urban area of Zhenjiang and Jiangsu University. To ensure the healthy growth of the seedlings, all of them are planted in cultivation pots containing biochar soil and yellow soil, and supported by sturdy wooden stakes. Additionally, a distance of 2 to 5 meters is maintained between each tree trunk to allow for sufficient growing space.

To create experimental datasets, point cloud data of various tree species were collected at this nursery between September 2022 and March 2023. Fig. 1 illustrates the layout of the whole nursery (imagery: extracted from gaode map). The original point clouds in the self-made nursery dataset were captured using the Livox Horizon scanner at different areas of the nursery. The Livox Horizon is a high-performance scanner with accurate range precision (0.02 meter at 20 meters) and angular precision (less than 0.05°). In the process of collecting data, scanner was mounted on a tripod and the integration time was set as 3000 milliseconds to capture a sufficient number of points. More parameters of the Livox Horizon scanner are shown in Table 1.

This nursery contains a variety of common landscape trees, including *Llex cornuta varfortunei*, *Osmanthus fragrans*, *Malus halliana koehne*, *Prunus cerasifera*, *Acer palmatum thunb*, *Chimonanthus praecox*, *Eriobotrya japonica*, and others. The point cloud data of the above seven tree species were collected and used to the nursery datasets. Samples for each of the seven tree species are presented in Fig. 2. To realize the tasks of tree species classification and the segmentation of different parts of the trees, all the data gathered must be segmented and normalized. The specific process of point cloud data processing is illustrated in Fig. 3. Fig. 3a shows the point cloud data obtained by the Livox Horizon scanner. In addition to the point clouds of the targets (trees), the data also contains some useless point clouds, such as point clouds of ground, background, and noisy point clouds. CloudCompare software was employed to filter and segment these useless point clouds. The 3D coordinates (x, y, z) of all segmented objects were saved in TXT format. Fig. 3b shows the point clouds of a target after filtering out the useless point clouds.

For better models training and generalization, all point clouds needed to be further normalized. The initial normalization step ensures that each group of point clouds contained an equal number of points. In the experiments, every set of point clouds consists of 10000 points. If a group of point clouds comprised more than 10000 points, 10000 points were chosen via random sampling. Subsequently, each set of point clouds was normalized in accordance with the following formula to guarantee that the coordinates of all points fell within the range of [-1 1]:

$$\frac{(x - x_{\min}, y - y_{\min}, z - z_{\min})}{\max(x_{\max} - x_{\min}, y_{\max} - y_{\min}, z_{\max} - z_{\min})} \quad (1)$$

where X_{\max} , Y_{\max} , and Z_{\max} are the maximum X, Y, and Z coordinate values of each group of point clouds; X_{\min} , Y_{\min} , and Z_{\min} are the minimum X, Y, and Z coordinates of each group of point clouds. An example of a group of segmented and normalized point clouds is depicted in



Fig. 1. The layout of the whole study site location and the photo of the part study site location.

Table 1
The parameters of Livox Horizon scanner.

Sensor	Livox Horizon scanner
Laser Wavelength	905 nm
Field of view	Horizontal:81.7°;Vertical:25.1°
Detection Range(@100 klx)	90 m @ 10% reflectivity 130 m @ 20% reflectivity 260 m @ 80% reflectivity
Range Precision (1σ@20m)	2 cm ¹
Angular Precision(1σ)	< 0.05°
Point Rate	240,000 points/s (first return) 480,000 points/s (dual return)
Dimensions	77 × 115 × 84
Weight	1300 g

Fig. 3c.

To obtain a greater quantity of point cloud data for classification experiments, a total of 1046 trees in the nursery were selected as samples and scanned from various locations. A total of 1828 groups of point clouds were employed in classification experiments. For these experiments, the ratio of the training set to the testing set is approximately 7:3. A total of 1284 groups of point clouds were treated as the training set and 544 groups of point clouds were allocated for testing. Information regarding the number of tree species, corresponding groups of point clouds, training set, and testing set is presented in **Table 2**.

The normalized point cloud data were further processed for segmented experiments. With the aid of CloudCompare software, each group of point clouds was divided into two to four parts. The segmented

parts were labeled as follows: crown: 0; trunk: 1; pot: 2; supporting pole: 3. **Fig. 3d** presents the segmented diagram of the sample tree, point clouds of various colors represent distinct component parts. The corresponding examples of different types of point cloud trees following the segmentation process are shown in **Fig. 4**.

2.2. D-PointNet++ neural network design

This section presents detailed information on the proposed D-PointNet++ model. The novel feature extraction module is named the Dense PointNet layer, which refers to the connection pattern of DenseNet. The proposed multiple features fusion operation and more implementation details are also presented in this section.

Fig. 5 elaborates the model architecture of D-PointNet++, which consists of the classification branch and the segmentation branch. Considering the hierarchical framework can effectively utilize the advantages of global and local features, the proposed network primarily takes it as the basic structure. As the number of network layers increases, the number of sampled center points decreases, while the corresponding extracted features become more enriched. The input of the network model is a group of unordered point clouds with N points: $P = (p_1, p_2, \dots, p_N) \subseteq \mathbb{R}^F$, where F denotes the feature dimensionality of point clouds. For simplicity, in this paper, F is set as 3, that is, each input point of the model contains coordinate information (x, y, z).

The designed dense set abstraction levels replace the set abstraction levels in PointNet++ to extract global and local features of point clouds. More details about schematic diagrams of set abstraction level in



Fig. 2. Seven kinds of common landscape trees as collected targets.

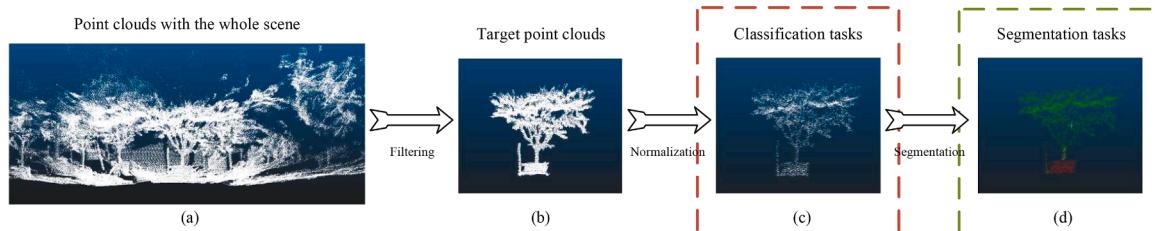


Fig. 3. The visualization of point cloud data processing. (a): Point clouds obtained by Livox Horizon scanner. (b): Filter out point clouds of ground, background, and noise point clouds to get the target point clouds. (c): Normalize the number and coordinates of each set of point clouds for classification experiments. (d): Segment different parts of the point clouds of the tree and represent them in different colors for segmentation experiments.

Table 2

Number of samples, groups of points, training and testing sets of different types of trees in the classification experiments.

Name	Total	Number of groups	Training set	Testing set
<i>Ilex cornuta varfortunei</i>	163	284	199	85
<i>Osmanthus fragrans</i>	120	179	126	53
<i>Malus halliana koehne</i>	144	283	201	82
<i>Prunus cerasifera</i>	164	287	201	86
<i>Acer palmatum thunb</i>	151	290	203	87
<i>Chimonanthus praecox</i>	158	290	203	87
<i>Eriobotrya japonica</i>	146	215	151	64

PointNet++ and the dense set abstraction level in D-PointNet++ are presented in Fig. 6a and b, respectively. The dense set abstraction level mainly comprises three components: the sampling layer, the grouping layer, and the dense PointNet layer.

At present, there are many existing heuristic and learning-based sampling approaches, including farthest point sampling, random sampling, generator-based sampling and so on. Compared with other sampling approaches, farthest point sampling has a better ability to cover the entire point set, which is selected to sample center points at the sampling layer in the proposed model.

The function of the grouping layer is to get neighboring points for each center point and construct local regions. Ball query can find neighboring points within the fixed region scale, which can make constructed local region features more generalizable. The ball query can obtain K neighboring points in each local region. If the number of neighboring points in the local area is less than K , repeat the first

neighboring point coordinates until the requirement is met. In addition, neighboring points in each local area need to be converted into corresponding local coordinates before extracting local features.

In the field of computer vision, the emergence of the deep residual network (ResNet) is of considerable significance, as it enhances the performance of deep neural networks with deep structures. Before ResNet, most neural networks mainly relied on continuously stacking convolutional layers and pooling layers to obtain new network structures. However, with the increasing number of layers, the models are more likely to suffer from the degradation problems and have higher training errors. To address these problems, ResNet adopts the residual structure by adding shortcut connections to the outputs of the stacking layers. Alternatively, the dense convolutional network (DenseNet) employs the dense connectivity pattern and outperforms ResNet. Inspired by the DenseNet, the connectivity pattern of the PointNet layer is changed, which is referred to as the dense PointNet layer and depicted in Fig. 6b. We assume that N_{in} points with the d_{in} dimensional features are given as input of the dense set abstraction level. Together with the coordinate information of the points, each point contains $(3 + d_{in})$ dimensional features. At the sampling layer, N points are selected as center points by the farthest point sampling. The ball query is utilized to identify K neighbors of each center point within a fixed radius to construct local regions. Through MLPs in the PointNet layer, the features dimensions of points will be raised from $(3 + d_{in})$ to d_{mlp1} or d_{mlp2} . According to the shortcut connections in the dense connectivity pattern, the features from all previous layers are concatenated to obtain $(3 + d_{in} + d_{mlp1})$ and $(3 + d_{in} + d_{mlp1} + d_{mlp2})$ dimensional features, respectively. Multiple dense set abstraction levels comprise the entire network,

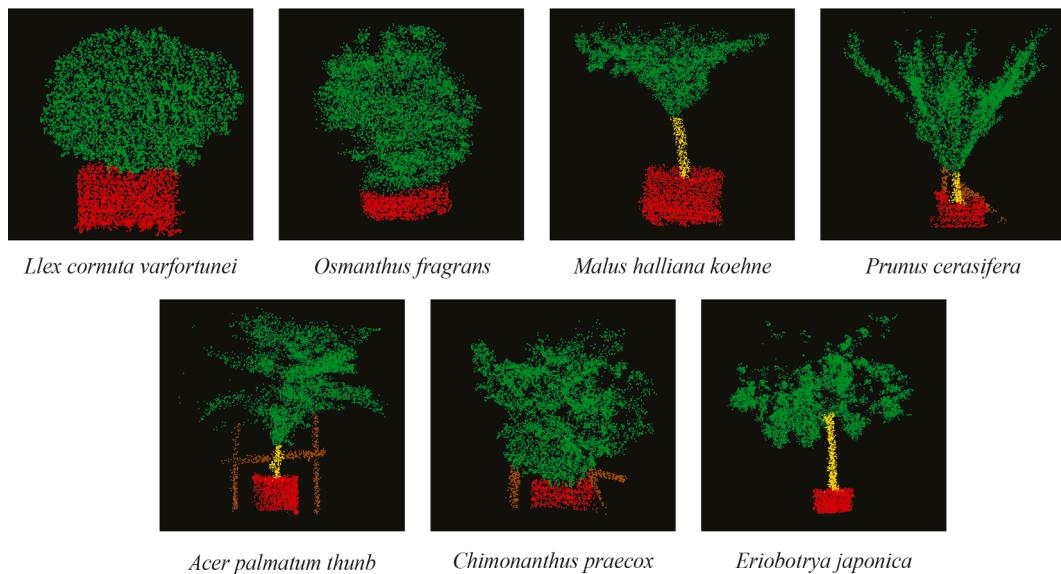


Fig. 4. The schematic diagrams of the segmented point clouds of seven tree species. (Crown: green point clouds; Trunk: yellow point clouds; Pot: red point clouds; Supporting rod: brown point clouds).

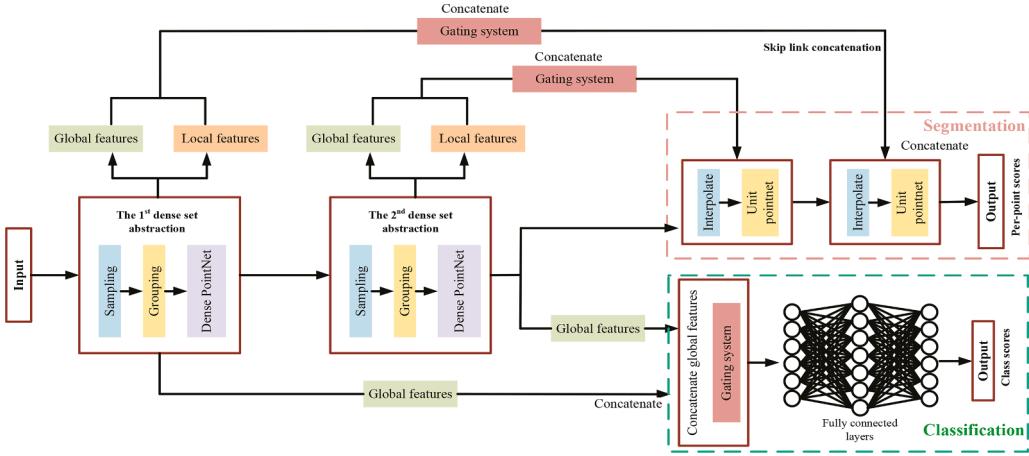


Fig. 5. Architecture of D-PointNet++ with classification branch and segmentation branch.

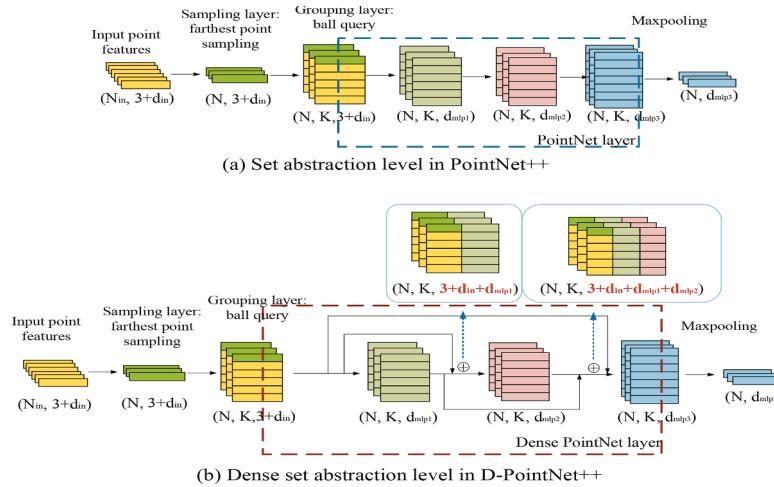


Fig. 6. The schematic diagram of set abstraction level in PointNet++ and dense set abstraction level in D-PointNet++. The dense set abstraction level is mainly composed of the sampling layer, the grouping layer and dense PointNet layer.

enhancing network efficiency by exploiting learning features from various layers.

Generally speaking, each point in different dense set abstraction levels typically contains different information. To make use of different levels of features, these features are firstly fused and get more representative features. Then, the neural network models can extract higher-dimensional information from the fusion features and realize the tasks of classification or segmentation efficiently. Hence, the proposed model implements a gating fuse operation to fuse features with varying information (global features at different levels or the global and local features). Taking the different levels of global features as an example to clarify the proposed gated fusion operation. Global features of point clouds play important roles in either classification or segmentation tasks. PointNet++ utilizes the max pooling function in the last set abstraction level to obtain global features and implement classification tasks. PointNet++ has several set abstraction levels, each yielding different global features of the point clouds. However, PointNet++ only uses the highest-level global features while neglecting corresponding global features in other set abstraction levels. Consequently, in D-PointNet++, global features from all set abstraction levels are integrated to improve the performance. More details are shown in Fig. 7, where N , N_1 , or N_2 represent the number of input points and center points in the first or the second set abstraction level, respectively.

In Fig. 7, the global features obtained from the two set abstraction levels are assumed as R_{global_low} and R_{global_high} , the corresponding

dimensions are C_{global_low} and C_{global_high} respectively. The global features are firstly fed into the convolutional networks to generate relevant coefficients α_{global_low} and α_{global_high} :

$$\begin{cases} \alpha_{global_low} = \sigma(\omega_{global_low}R_{global_low} + b_{global_low}) \\ \alpha_{global_high} = \sigma(\omega_{global_high}R_{global_high} + b_{global_high}) \end{cases} \quad (2)$$

Where σ is non-linear sigmoid function, ω_{global_low} , ω_{global_high} , b_{global_low} and b_{global_high} are parameters of the convolutional neural networks. Then the generated coefficients are multiplied by the input features and conduct concatenation operations to obtain the fused features X_{global}^F . The specific process can be represented by the following formula:

$$X_{global}^F = concat(\alpha_{global_low} \times R_{globalLow}, \alpha_{globalHigh} \times R_{globalHigh}) \quad (3)$$

Due to the gated fusion operation, the obtained global features are expanded from C_{global_high} to $(C_{global_low} + C_{global_high})$. The gated fusion operation is used to concatenate global features from different dense set abstraction levels, and then input them to the shared fully connected layers to obtain the prediction scores for k classes. The category of input point clouds is decided according to the highest prediction score. Furthermore, to prevent over-fitting, the dropout layers with the dropout rate of 0.4 are added to the fully connected layers in our paper.

Like PointNet++, in Fig. 5, D-PointNet++ adopts the classic U-Net style (Ronneberger et al., 2015) with distance-based interpolation and across-level skip links as the structure of the segmentation branch.

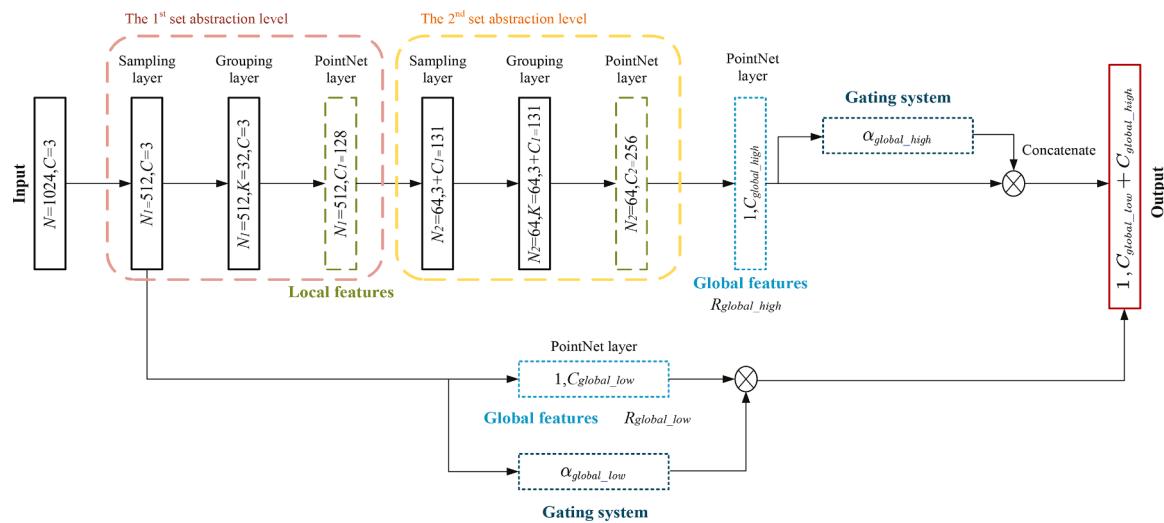


Fig. 7. Using the gated fusion operation generates parameters and concatenates global features from the first and the second set abstraction levels.

Features of points in previous layers are propagated through interpolation with inverse distance based on K nearest neighbors. And local and global features in each dense set abstraction level are concatenated by gated fusion operation to propagated features. Finally, the unit pointnet (one-by-one convolution) is repeated until features are propagated to the original point clouds and get the predicted score of each point.

3. Results and discussions

3.1. Experimental equipments and evaluation metrics

All experiments were conducted on a computer with Intel(R) Xeon (R) Gold 6226R CPU, and NVIDIA RTX3090 GPU. And models were carried out in PyCharm Community Edition 2021.1.3 based on python 3.8, Pytorch 1.13 and CUDA 11.7.

Evaluation metrics can measure the performances of different neural network models. In the classification experiments, the confusion matrix is mainly used to compare the predicted results with the ground truths. The classified performances of the models can be visualized by using the confusion matrix. Besides, identifying confusion between similar classes is helpful for more error analysis.

Two commonly used metrics for classification, overall accuracy (OA) and mean class accuracy (mAcc), are employed to evaluate the performances of the models. Usually, the higher values of OA and mAcc mean better classification performance. OA and mAcc represent the overall accuracy rate of all data and the average classification accuracy rate of each class, and can be expressed by following formulas:

$$OA = \frac{TP}{W} \quad (4)$$

$$mAcc = \frac{1}{m} \sum_{i=1}^m \frac{TP_i}{W_i} \quad (5)$$

where TP or TP_i represents the true positive number of all testing data or of i th category; W or W_i indicates the total number of groups of point clouds of all testing data or of i th category, respectively; m is the number of categories in the data.

The purpose of the classification tasks is to predict the category of the input point clouds. While segmentation tasks belong to the domain of multi-class classifications on the point-level, which are needed to determine the category of each point. IoU (Intersection over Union) and $mIoU$ (the average IoU) are segmentation evaluation metrics. The expression formula of IoU is as follows:

$$IoU = \frac{TP}{TP + FP + FN} \quad (6)$$

where TP , FP , and FN represent the number of the true positive, false positive, and false negative points. $mIoU$ is the average value of IoU for all parts. In addition to $mIoU$, Precision, Recall, and $F1$ -Score are also common metrics used to evaluate the segmentation performances of models, which can be represented by the following formulas:

$$Precision = \frac{TP}{TP + FP} \quad (7)$$

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

$$F1-score = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (9)$$

In this paper, the segmentation performances of different models were evaluated in terms of $mIoU$, Precision, Recall and $F1$ -Score.

3.2. Comparative models

PointNet is the initial neural network model that directly extracts per-point features from point clouds using shared MLPs. It employs joint alignment networks and the symmetry function to address the irregularity of point clouds. Based on PointNet, PointNet++ adopts the hierarchical structure and extracts local features of point clouds. Both PointNet and PointNet++ are classic point-based neural network models, which have been widely used in classifying tree species, separating the structural components for crops, and other applications (Hu et al., 2023; Jayakumari et al., 2021; Liu et al., 2021; Wei et al., 2022). In fact, PointNet++ solely employs the highest-level global features, indicating potential for improvement on classification and segmentation tasks. In this paper, the performance of D-PointNet++ are compared with PointNet and PointNet++.

3.3. Classification experiments

The tree species classification experiments were carried out to compare the classification performances of PointNet, PointNet++ and D-PointNet++. D-PointNet++ utilizes concatenation and gated fusion operation to combine multiple features. Addition and concatenation are also commonly used fusion operations that are both simple and straightforward. These fusion strategies can aggregate more information through adding or concatenating multiple features. However, both

concatenation and addition operations will mix the useful information with a large amount of non-informative features. This part also discusses the influence of different fusion operations on the results. Except concatenation operation, addition and gating fuse operation proposed in this paper, the generated parameters from gating systems multiply with original features and add together as another kind of feature fusion operation. OA and mAcc were used as evaluation indicators in the experiments, and confusion matrices were utilized for further error analysis.

The limited quantity of annotated data commonly negatively impacts model performance. To address this issue, transfer learning first helps models grasp more features or knowledge by pre-training from the existing datasets. Then the structures of the models are adjusted, and then changed dataset to further train the model. ModelNet40 is a public point cloud dataset designed specifically for classification experiments, consisting of 12311 groups of point clouds divided into 40 categories. 9843 groups of point clouds are used for training and 2468 groups of point clouds are treated as testing dataset. In classification experiments, all models were first pre-trained on the ModelNet40 dataset. Then the self-made dataset (1828 groups of nursery trees point clouds) was utilized to train and test these models. During the process of classification experiments, 1024 points were sampled from each set of point clouds as the input. The loss and accuracy curves of the ModelNet40 dataset during the training processes are shown in Fig. 8. According to accuracy and loss curves, all models can reach convergence states before 150 epochs. The classification results on the ModelNet40 dataset are shown in Table 3. As we can see, D-PointNet++ has better classification performance than PointNet or D-PointNet++ using the public dataset.

After pre-training, the numbers of output nodes in the full connection layers of models were changed and other parameters remained. Then the self-made dataset was used to train all pre-trained models. The accuracy and loss curves are shown in Fig. 9, and the results are exhibited in Table 4. As shown in the classification results, the proposed D-PointNet++ model has outstanding classification performance because of the highest values of OA and mAcc. Compared with PointNet++, the OA of D-PointNet++ is improved by 2.39% and the mAcc of D-PointNet++ increased by 2.60%.

Fig. 10 represents the confusion matrices of PointNet++ and D-PointNet++, which illustrates the classification results of each tree species on the self-made testing set. Compared with PointNet++, D-PointNet++ can get higher classification accuracy on *Osmanthus fragrans*, *Prunus cerasifera*, *Acer palmatum thunb* and *Chimonanthus praecox*. According to the confusion matrices, all models have lower classification accuracy on *Acer palmatum thunb* and *Chimonanthus praecox*. One possible reason of the low accuracy is that *Acer palmatum thunb* is similar to *Eriobotrya japonica*, and *Chimonanthus praecox* resembles *Osmanthus fragrans* in tree shape. Ignoring color information and using only location information can also lead to classification errors.

Furthermore, as shown in Table 4, the results of the models with gated fusion operations are better than those without gated fusion operations. Concatenation can be regarded as a more suitable feature fusion operation compared to addition. And concatenation and gating system used in D-PointNet++ can get the best classification

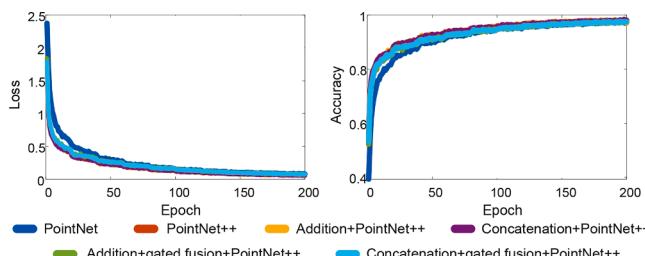


Fig. 8. The accuracy and loss curves of different models with ModelNet40 dataset during the training processes.

Table 3

The classification results of different models on Modelnet40 dataset.

	OA (%)	mAcc (%)
PointNet	90.73	86.71
PointNet+	92.23	88.77
Addition + PointNet++	91.98	89.57
Concatenation + PointNet++	92.26	89.94
Addition + gated fusion operation + PointNet++	92.50	89.68
Concatenation + gated fusion operation + PointNet++(D-PointNet++)	92.78	89.73

performance due to the highest values of OA and mAcc.

3.4. Segmentation experiments

Different from classification tasks, segmentation tasks need to determine the category of each point of the input point clouds. More useful information can be obtained from the segmented crown and trunk point clouds. During training or testing processes, 2048 points were sampled each time as input to models and the testing results are shown in Table 5. Table 5 displays the *mIoU*, Precision, Recall and *F1*-Score values achieved by D-PointNet++, which reached 89.90%, 92.18%, 94.92% and 93.50% respectively. These values are higher than that of both PointNet and PointNet++, suggesting D-PointNet++ presents superior segmentation performance.

Furthermore, Fig. 11 illustrates some visualization segmentation results on the testing set using PointNet, PointNet++ and D-PointNet++, showing the accuracy of the proposed module in dividing tree crowns and trunks. Among the three models, PointNet demonstrates the lowest segmentation accuracy and classifies the points belonging to the supporting pots into crowns and trunks. PointNet++ demonstrates better segmentation performance than PointNet. However, due to ignoring the low-dimensional features, PointNet++ is deficient in dealing with some detail information. For instance, certain points belonging to supporting pots may be erroneously classified as the crown or trunk, as can be seen in the segmentation result within the white box in Fig. 11. Due to the use of combination of low-dimensional and high-dimensional features, compared with other models, the segmentation performance of D-PointNet++ is the best.

In the segmentation experiments, we also evaluated the impact of different fusion operations on the results, which are shown in Table 5. Different from the classification experiments, concatenation can not improve the segmentation performances of the models obviously. Nevertheless, the use of gated fusion can improve the segmentation performances. Whether addition or concatenation, the *mIoU* improvement obtained by adding gated fusion operation is higher than 1%. In addition, the *mIoU*, Precision, and *F1*-Score acquired by the fusion operation used in this paper is higher than that of other fusion operations. Although the obtained recall is slightly lower than the maximum value by 0.08%, overall, the fusion method used in this paper can get the best segmentation performance.

4. Conclusion

Nurseries play a crucial role in urban greening construction and are specifically intended to cultivate various types of garden trees. To facilitate the efficient management of nurseries and the practical applications of agricultural robots, this paper proposes a point cloud-based neural network model, termed D-PointNet++ (Dense PointNet++), which enables the classification of individual tree point clouds and the segmentation of various components such as crowns, trunks, and other parts. Based on PointNet++, D-PointNet++ extracts features of point clouds referring to the dense connection structure of DenseNet. Additionally, gating and concatenation are employed as feature fusion way to fully leverage the low and high-dimensional features of the point cloud

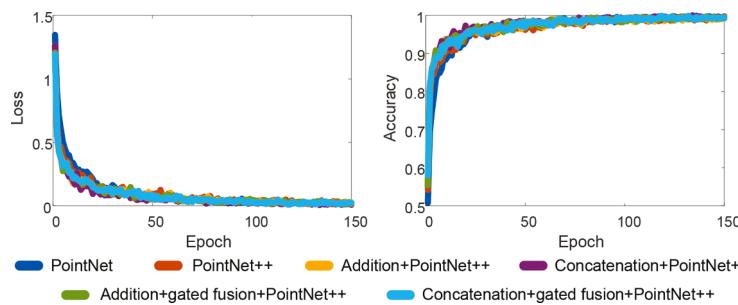


Fig. 9. The accuracy and loss curves of different models with self-made dataset during the training processes.

Table 4

The classification results of different models on self-made dataset.

	OA (%)	mAcc (%)
PointNet	81.25	80.50
PointNet+	90.26	89.94
Addition + PointNet++	90.63	90.42
Concatenation + PointNet++	91.36	91.08
Addition + gated fusion operation + PointNet++	91.91	91.75
Concatenation + gated fusion operation + PointNet++(D-PointNet++)	92.65	92.54

and improve accuracy.

We conducted classification experiments using a self-made nursery dataset. The experimental results demonstrated that D-PointNet++ achieved higher OA and mAcc values, 92.65% and 92.54%, respectively, outperforming classical PointNet and PointNet++ models. In the segmentation tasks, *mIoU*, precision, recall and F1-Score of D-PointNet++ were 89.90%, 92.18%, 94.92% and 93.50%. Considering the low-dimensional characteristics of point clouds, D-PointNet++ has better segmentation performance.

In practical applications, the proposed D-PointNet++ model can provide more accurate information about tree species, trunks, crowns, and other related information. Having accurate information on tree species is benefit to nursery management. Additionally, tree trunk and crown information can be used by agricultural robots to realize target location and autonomous variable rate spraying, resulting in improved efficiency. Thus, the proposed model holds significant practical significance in the field of agriculture. In subsequent studies, we intend to obtain target parameters (positions, canopy volumes, etc.) based on these data and realize autonomous navigation tasks and variable spraying operations via our hardware platform.

Ethical Statement

As an academic journal dedicated to publishing research related to plants, we pledge to uphold the highest level of ethical standards in all aspects of our operations. We follow the guidelines established by the Convention on Biological Diversity and the Convention on the Trade in Endangered Species of Wild Fauna and Flora.

The data collected and used in the experiment were obtained from a nursery (32.12N, 119.31E) in Zhenjiang, China, which often provides nursery trees to Jiangsu University and Zhenjiang urban greening projects.

CRediT authorship contribution statement

Jie Xu: Conceptualization, Methodology, Software, Validation, Writing – original draft, Writing – review & editing. **Hui Liu:** Conceptualization, Validation, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Yue Shen:** Validation, Formal

Table 5

The segmentation results of PointNet, PointNet++ and D-PointNet++ on self-made dataset.

	mIoU (%)	Precision (%)	Recall (%)	F1-Score (%)
PointNet	79.12	76.87	82.67	79.39
PointNet+	86.92	88.87	93.92	91.23
Addition + PointNet++	88.51	89.90	94.63	92.12
Concatenation + PointNet++	88.59	90.16	94.20	92.07
Addition + gated fusion operation + PointNet++	89.16	90.18	95.00	92.45
Concatenation + gated fusion operation + PointNet++(D-PointNet++)	89.90	92.18	94.92	93.50

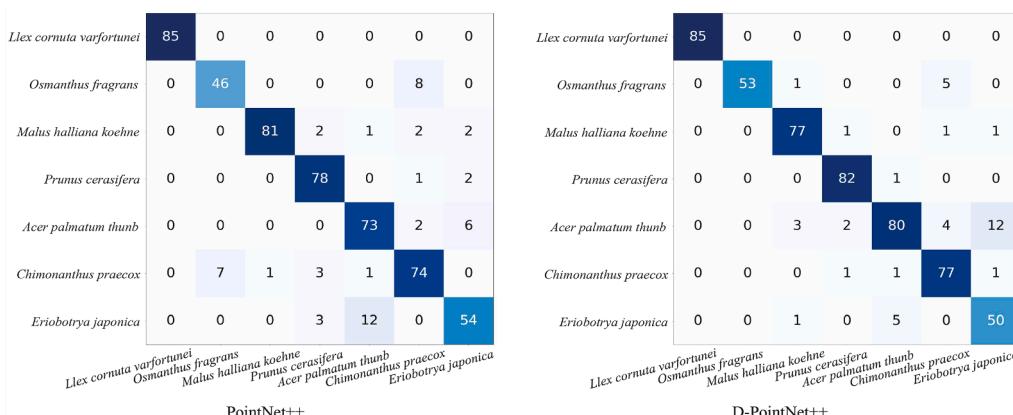


Fig. 10. Confusion matrices of the classification results using baseline PointNet++ and proposed D-PointNet++.

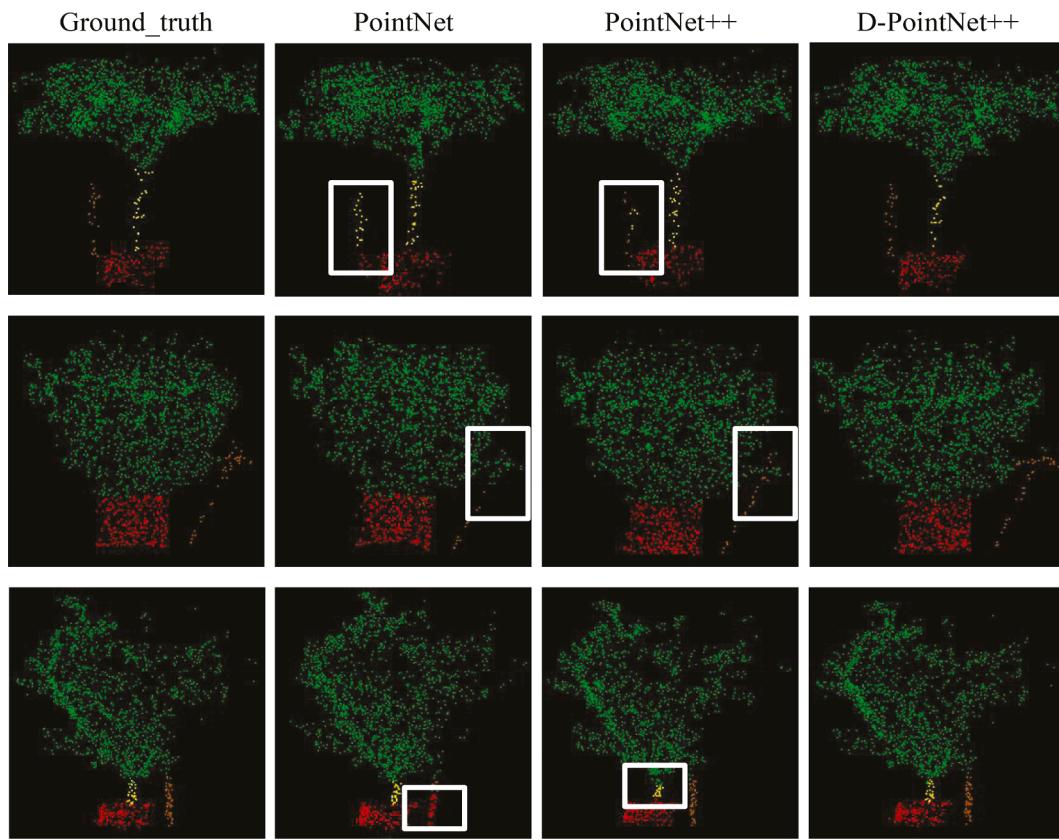


Fig. 11. visualization segmentation results on the testing set from different models. The point clouds in white boxes are the wrong segmentation results.

analysis, Investigation. **Xiao Zeng:** Data curation, Writing – review & editing. **Xinpeng Zheng:** Data curation, Writing – review & editing.

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.

Data availability

The data that has been used is confidential.

References

- Aada, H., Lassi, R., Matti, L., Yu, X., Antero, K., Harri, K., Josef, T., Leena, M., Eric, H., Ville, L., Markus, H., Ville, K., Juha, H., 2023. Individual tree segmentation and species classification using high-density close-range multispectral laser scanning data. *ISPRS Open J. Photogramm. Remote Sens.* 100039.. <https://doi.org/10.1016/j.photo.2023.100039>.
- Alexey B., Chien-Yao W., Liao H.-Y.M., 2020. YOLOv4: Optimal Speed and Accuracy of Object Detection. [arXiv preprint arXiv:2004.10934](https://arxiv.org/abs/2004.10934).
- Andreas, T., Christoph, G., Ralf, K., Tim, R., Arne, N., 2022. Automatic tree crown segmentation using dense forest point clouds from Personal Laser Scanning (PLS). *Int. J. Appl. Earth Obs. Geoinf.* 114, 103025. <https://doi.org/10.1016/j.jag.2022.103025>.
- Bruno, J.L.P., Lorence, R.O., Jared, S., Evans, R.Y., 2022. A nursery system nitrogen balance for production of a containerized woody ornamental plant. *Sci. Hortic.* 291, 110569. <https://doi.org/10.1016/j.scienta.2021.110569>.
- Charles, R.Q., Su, H., Kaichun, M., Guibas, L.J., 2017. PointNet: deep learning on point sets for 3D classification and segmentation. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, Honolulu, HI, pp. 77–85. <https://doi.org/10.1109/CVPR.2017.16>.
- Chen, D., Zhang, J., Zhang, Z., Wan, X., Hu, J., 2022. Analyzing the effect of light on lettuce Fv/Fm and growth by machine learning. *Sci. Hortic.* 306, 111444. <https://doi.org/10.1016/j.scienta.2022.111444>.
- Chen, Y., Xiong, Y., Zhang, B., Zhou, J., Zhang, Q., 2021. 3D point cloud semantic segmentation toward large-scale unstructured agricultural scene classification. *Comput. Electron. Agric.* 190, 106445. <https://doi.org/10.1016/j.compag.2021.106445>.
- Dersch, S., Schöttl, A., Krzystek, P., Heurich, M., 2023. Towards complete tree crown delineation by instance segmentation with Mask R-CNN and DETR using UAV-based multispectral imagery and lidar data. *ISPRS Open J. Photogramm. Remote Sens.* 8, 100037. <https://doi.org/10.1016/j.jphoto.2023.100037>.
- Fabrício Packer, G., Eduardo Sanches, S., Simone Rodrigues da, S., Eduardo Toller, R., Lilian, A., 2011. Role of healthy nursery plants in orange yield during eight years of Citrus Variegated Chlorosis epidemics. *Sci. Hortic.* 129 (2), 343–345. <https://doi.org/10.1016/j.scienta.2011.03.038>.
- Feng, Y., Zhang, Z., Zhao, X., Ji, R., Gao, Y., 2018. GVCNN: group-view convolutional neural networks for 3D shape recognition. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE, Salt Lake City, UT, pp. 264–272. <https://doi.org/10.1109/CVPR.2018.00035>.
- Herbert, B., Andreas, E., Tinne, T., Luc, V., 2008. Speeded-Up Robust Features (SURF). *Comput. Vis. Image Understanding* 110 (3), 346–359. <https://doi.org/10.1016/j.cviu.2007.09.014>.
- Hu, H., Yu, J., Yin, L., Cai, G., Zhang, S., Zhang, H., 2023. An improved PointNet++ point cloud segmentation model applied to automatic measurement method of pig body size. *Comput. Electron. Agric.* 205, 107560. <https://doi.org/10.1016/j.compag.2022.107560>.
- Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q., 2017. Densely connected convolutional networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, pp. 2261–2269. <https://doi.org/10.1109/CVPR.2017.243>.
- Jayakumari, R., Nidamanuri, R., Ramiya, A., 2021. Object-level classification of vegetable crops in 3D LiDAR point cloud using deep learning convolutional neural networks. *Precis. Agric.* 22 (5), 1617–1633. <https://doi.org/10.1007/s11119-021-09803-0>.
- Kristina, K., Bernhard, H., Martin, H., Thomas, J., Bastian, S., Holger, L., 2015. Comparative classification analysis of post-harvest growth detection from terrestrial LiDAR point clouds in precision agriculture. *ISPRS J. Photogramm. Remote Sens.* 104, 112–125. <https://doi.org/10.1016/j.isprsjprs.2015.03.003>.
- Liu, H., 2022. Classification of urban tree species using multi-features derived from four-season RedEdge-MX data. *Comput. Electron. Agric.* 194, 106794. <https://doi.org/10.1016/j.compag.2022.106794>.
- Liu, M., Han, Z., Chen, Y., Liu, Z., Han, Y., 2021. Tree species classification of LiDAR data based on 3D deep learning. *Measurement* 177, 109301. <https://doi.org/10.1016/j.measurement.2021.109301>.
- María José, J.-M., María del, C.-M.-M., Inmaculada, M.-A., Hava, R., Ricardo, F.-E., 2018. Interaction between mycorrhization with Glomus intraradices and phosphorus in nursery olive plants. *Sci. Hortic.* 233, 249–255. <https://doi.org/10.1016/j.scienta.2018.01.057>.

- Maturana, D., Scherer, S., 2015. VoxNet: a 3D convolutional neural network for real-time object recognition. 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, Hamburg, Germany, pp. 922–928. <https://doi.org/10.1109/IROS.2015.7353481>.
- Qi, C.R., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L.J., 2016. Volumetric and multi-view CNNs for object classification on 3D data. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, Las Vegas, NV, USA, pp. 5648–5656. <https://doi.org/10.1109/CVPR.2016.609>.
- Qi C.R., Yi L., Su H., Guibas L.J., 2017. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. *arXiv preprint arXiv:1706.02413*.
- Ronneberger O., Fischer P., Brox T., 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. *10.1007/978-3-319-24574-4_28*.
- Ross G., Jeff D., Trevor D., Jitendra M., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. *arXiv preprint arXiv:1311.2524*.
- Surasak, T., Takahiro, I., Cheng, C.-h., Wang, C.-e., Sheng, P.-y., 2018. Histogram of oriented gradients for human detection in video. 2018 5th International Conference on Business and Industrial Research (ICBIR).
- Wei, H., Xu, E., Zhang, J., Meng, Y., Wei, J., Dong, Z., Li, Z., 2022. BushNet: effective semantic segmentation of bush in large-scale point clouds. *Comput. Electron. Agric.* 193, 106653. <https://doi.org/10.1016/j.compag.2021.106653>.
- Xu, Z., Shen, X., Cao, L., Nicholas, C.C., Tristan, R.H.G., Zhong, T., Zhao, W., Sun, Q., Ba, S., Zhang, Z., Wu, X., 2020. Tree species classification using UAS-based digital aerial photogrammetry point clouds and multispectral imageries in subtropical natural forests. *Int. J. Appl. Earth Obs. Geoinf.* 92, 102173. <https://doi.org/10.1016/j.jag.2020.102173>.
- Yang, T., Ye, J., Zhou, S., Xu, A., Yin, J., 2022. 3D reconstruction method for tree seedlings based on point cloud self-registration. *Comput. Electron. Agric.* 200, 107210. <https://doi.org/10.1016/j.compag.2022.107210>.
- Yang, X., Zhang, R., Zhai, Z., Pang, Y., Jin, Z., 2019. Machine learning for cultivar classification of apricots (*Prunus armeniaca* L.) based on shape features. *Sci. Hortic.* 256, 108524. <https://doi.org/10.1016/j.scienta.2019.05.051>.
- Yang, Y., Liang, X., Wang, B., Xie, Z., Shen, X., Sun, X., Zhu, X., 2022. Biophysical parameters retrieval of mangrove ecosystem using 3D point cloud descriptions from UAV photographs. *Ecol. Inf.* 72, 101845. <https://doi.org/10.1016/j.ecoinf.2022.101845>.
- Yuan, W., Choi, D., Bolkas, D., 2022. GNSS-IMU-assisted colored ICP for UAV-LiDAR point cloud registration of peach trees. *Comput. Electron. Agric.* 197, 106966. <https://doi.org/10.1016/j.compag.2022.106966>.
- Yutaro, O., Hisayo, Y., Shu-Yen, L., Po-An, C., Ryutaro, T., 2020. Cultivar discrimination of litchi fruit images using deep learning. *Sci. Hortic.* 269, 109360. <https://doi.org/10.1016/j.scienta.2020.109360>.
- Zhang, C., Zhang, K., Ge, L., Zou, K., Wang, S., Zhang, J., Li, W., 2021. A method for organs classification and fruit counting on pomegranate trees based on multi-features fusion and support vector machine by 3D point cloud. *Sci. Hortic.* 278, 109791. <https://doi.org/10.1016/j.scienta.2020.109791>.
- Zhou, X., Sun, G., Xu, N., Zhang, X., Cai, J., Yuan, Y., Huang, Y., 2023. A method of modern standardized apple orchard flowering monitoring based on S-YOLO. *Agriculture* 13 (2), 380. <https://doi.org/10.3390/agriculture13020380>.