agriculture

MDPI

Article

Tomato Stem and Leaf Segmentation and Phenotype Parameter Extraction Based on Improved Red Billed Blue Magpie Optimization Algorithm

Lina Zhang [1], Ziyi Huang [1], Zhiyin Yang [1], Bo Yang [2], Shengpeng Yu [1], Shuai Zhao [1], Xingrui Zhang [1], Xinying Li [1], Han Yang [1], Yixing Lin [1] and Helong Yu [1,*]

[1] College of Information Technology, Jilin Agricultural University, Changchun 130118, China; berylh@mails.jlau.edu.cn (Z.H.)
[2] College of Information Engineering, Changchun University of Finance and Economics, Changchun 130217, China
* Correspondence: yuhelong@jlau.edu.cn

Abstract: In response to the structural changes of tomato seedlings, traditional image techniques are difficult to accurately quantify key morphological parameters, such as leaf area, internode length, and mutual occlusion between organs. Therefore, this paper proposes a tomato point cloud stem and leaf segmentation framework based on Elite Strategy-based Improved Red-billed Blue Magpie Optimization (ES-RBMO) Algorithm. The framework uses a four-layer Convolutional Neural Network (CNN) for stem and leaf segmentation by incorporating an improved swarm intelligence algorithm with an accuracy of 0.965. Four key phenotypic parameters of the plant were extracted. The phenotypic parameters of plant height, stem thickness, leaf area and leaf inclination were analyzed by comparing the values extracted by manual measurements with the values extracted by the 3D point cloud technique. The results showed that the coefficients of determination (R^2) for these parameters were 0.932, 0.741, 0.938 and 0.935, respectively, indicating high correlation. The root mean square error (RMSE) was 0.511, 0.135, 0.989 and 3.628, reflecting the level of error between the measured and extracted values. The absolute percentage errors (APE) were 1.970, 4.299, 4.365 and 5.531, which further quantified the measurement accuracy. In this study, an efficient and adaptive intelligent optimization framework was constructed, which is capable of optimizing data processing strategies to achieve efficient and accurate processing of tomato point cloud data. This study provides a new technical tool for plant phenotyping and helps to improve the intelligent management in agricultural production.

Keywords: tomato plants; red-billed blue magpie optimization algorithm; elite strategy; point cloud segmentation; convolutional neural network; phenotype extraction analysis

check for updates

Academic Editor: Domenico Pignone

Received: 11 December 2024
Revised: 6 January 2025
Accepted: 12 January 2025
Published: 15 January 2025

Citation: Zhang, L.; Huang, Z.; Yang, Z.; Yang, B.; Yu, S.; Zhao, S.; Zhang, X.; Li, X.; Yang, H.; Lin, Y.; et al. Tomato Stem and Leaf Segmentation and Phenotype Parameter Extraction Based on Improved Red Billed Blue Magpie Optimization Algorithm. *Agriculture* **2025**, *15*, 180. https://doi.org/10.3390/agriculture15020180

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In the context of modern agricultural science research, structural analysis of tomato plants is a key component for understanding their growth and development habits, optimizing cultivation and management practices, and improving final crop yields [1]. By utilizing point cloud datasets captured by high-precision depth cameras, we are able to achieve an accurate reconstruction of the three-dimensional morphology of tomato plants, which encompasses the detailed spatial layout of components such as stalks, leaves, and branches, as well as quantitative descriptions of their geometric features. However, the processing of point cloud data is not flawless, and often faces challenges such as data occlusion, noise

pollution, and data sparsity [2], and these unfavorable factors may lead to partial loss of information and degradation of segmentation accuracy. With the vigorous advancement of deep learning technology, how to achieve accurate segmentation of tomato plant stem and leaf structure relying on point cloud data, and to efficiently extract geometrical information, it has become a research focus and hotspot in this field [3].

Currently, the common stem and leaf segmentation in the plant domain can be categorized into three types: supervised learning, unsupervised learning and weakly supervised learning. Supervised learning achieves efficient feature extraction and fine segmentation of plant structure point cloud data through deep learning models that rely on fully labeled data for training. Researchers achieved automatic identification of key features in the input point cloud and accomplished accurate segmentation of plant organs by deep neural network training. In recent years, deep networks based on semantic segmentation have demonstrated excellent performance in agriculture, especially in tasks such as crop species segmentation and field area demarcation, with significantly improved accuracy. Meanwhile, target detection techniques have also made breakthroughs in agricultural applications, e.g., in the tasks of disease identification and fruit counting, methods such as YOLO and Faster R-CNN have been widely adopted, demonstrating extremely high efficiency and accuracy [4–6]. In addition, the application of multimodal data fusion techniques, such as combining remote sensing imagery, UAV imagery and hyperspectral data [7], provides more comprehensive and rich support for agricultural classification and segmentation. These methods provide a technical foundation for agricultural intelligence [8]. Lou L et al. Proposed a novel weakly -supervised framework Eff-3DPSeg for 3D plant bud segmentation [9]. This method combines deep learning and point cloud techniques to address the challenges of 3D plant bud segmentation. Wang, Y et al. achieved an effective segmentation of stems and leaves of tomato plants by combining skeleton extraction with up-pixel clustering techniques to achieve a certain segmentation accuracy [10]. Xusheng Zhong used Graph Convolutional Neural Network (GCNN) as a basic framework for plant point cloud segmentation research. The study used 864 plant point cloud data collected as a dataset and verified the performance of the model [11]. These methods are based on the capability of deep learning in feature learning and model optimization to handle complex data and improve segmentation refinement.

Unsupervised learning does not use any labeled data but looks for patterns in the data [12]. Pre-trained models or machine learning methods can autonomously extract features from large-scale unlabeled point cloud data and process and analyze the point cloud data using algorithms such as clustering and dimensionality reduction [13–16]. Miao, T et al. proposed an automated segmentation method for maize shoot stems and leaves based on 3D point cloud. The method used skeleton information as a priori knowledge to assist point cloud segmentation and realized an automated segmentation process [17]. Weakly supervised learning uses partially labeled data to guide the model for training, which is suitable for those scenarios where labeling resources are limited but supervised information is still needed [18]. Wu J et al. proposed a weakly supervised 3D point cloud semantic segmentation (WS3DSS) method, which is to develop a robust model that is able to learn from limited labeled data and effectively utilize the unlabeled data to enhance segmentation performance [19]. However, unsupervised learning has no labeling information, and the learned model may not be directly applicable to specific classification or regression tasks with limited performance enhancement and low accuracy. Weakly supervised learning may not be sufficient to effectively guide model learning due to incomplete or uncertainty in the labeling information, thus affecting the accuracy and stability of segmentation results [20–22].

Aiming at the challenges faced by the current segmentation, this paper proposes an optimization algorithm based on the improved red-billed blue magie, which aims to solve the following problems:

1. deep learning models generally have the problem of large parameter sizes. although the demand for computational resources can be significantly reduced through model lightweighting techniques, this process often leads to a decline in model performance, i.e., loss of accuracy, which affects the accuracy of the final segmentation results.
2. the segmentation accuracy of some deep learning models shows a certain degree of instability in complex environments, and this instability limits the applicability of the model in practical application scenarios, making it difficult to meet the demand for highly reliable segmentation results in agricultural production.

Aiming at the current problems, this paper aims to propose a method that can still obtain high segmentation accuracy with fewer convolutional layers and fewer number of parameters. To achieve this goal, the following strategies are adopted in this study:

1. the Red-billed Blue Magpie Optimization (RBMO) Algorithm is optimized using an elite strategy to improve the stability and search capability of the algorithm. The elite strategy improves the overall performance of the algorithm by retaining the best individuals in successive generations and guiding the evolution of the algorithm towards better solutions.
2. the optimized red-billed blue magpie algorithm is fused with a deep learning network model to optimize the effect of stem and leaf segmentation of tomato plants. This fusion strategy aims to ensure the robustness of the model when dealing with complex point cloud data while reducing the dependence on a large number of parameters.
3. the algorithm encapsulate customized convolutional layers by combining geometric and curvature features of the point cloud data. This approach makes the network model more attuned to the processing characteristics of 3D spatial data, thus improving the ability to recognize the details of plant structures while keeping the model lightweight.

2. Materials and Methods

2.1. Data Sources and Collection

In the fourth greenhouse of the Jilin Vegetable and Flower Scientific Research Institute, five independent growing blocks were set up in this study with uniform cultivation conditions in all blocks. Ninety-six tomato plants were cultivated in each block, and the overall sample size amounted to 480 plants. Within these blocks, plants were arranged in two rows with row spacing maintained at 0.1 m, while row length extended to 2.4 m (Figure 1a). Within the same row, the spacing between plants was approximately 0.05 m. In order to obtain the 3D structural data of tomato plants, a 3DScanner-630w (measurement size error: 0.001~0.03 mm; maximum lens pixels: 6.3 million) device was used for point cloud data acquisition (Figure 1b,c), and special attention was paid to avoiding the effects of shadows and reflections during the acquisition process to ensure the integrity of the data. Pre-processing operations were performed on the acquired image data using CloudCompare software 2.14. Using scaling, noise and disturbance enhancement, 8640 point cloud files were finally acquired. The dataset is divided into a ratio of 7:2:1. Ultimately, the preprocessed image data were visualized and displayed in 3D through the Python programming language (Figure 1d).

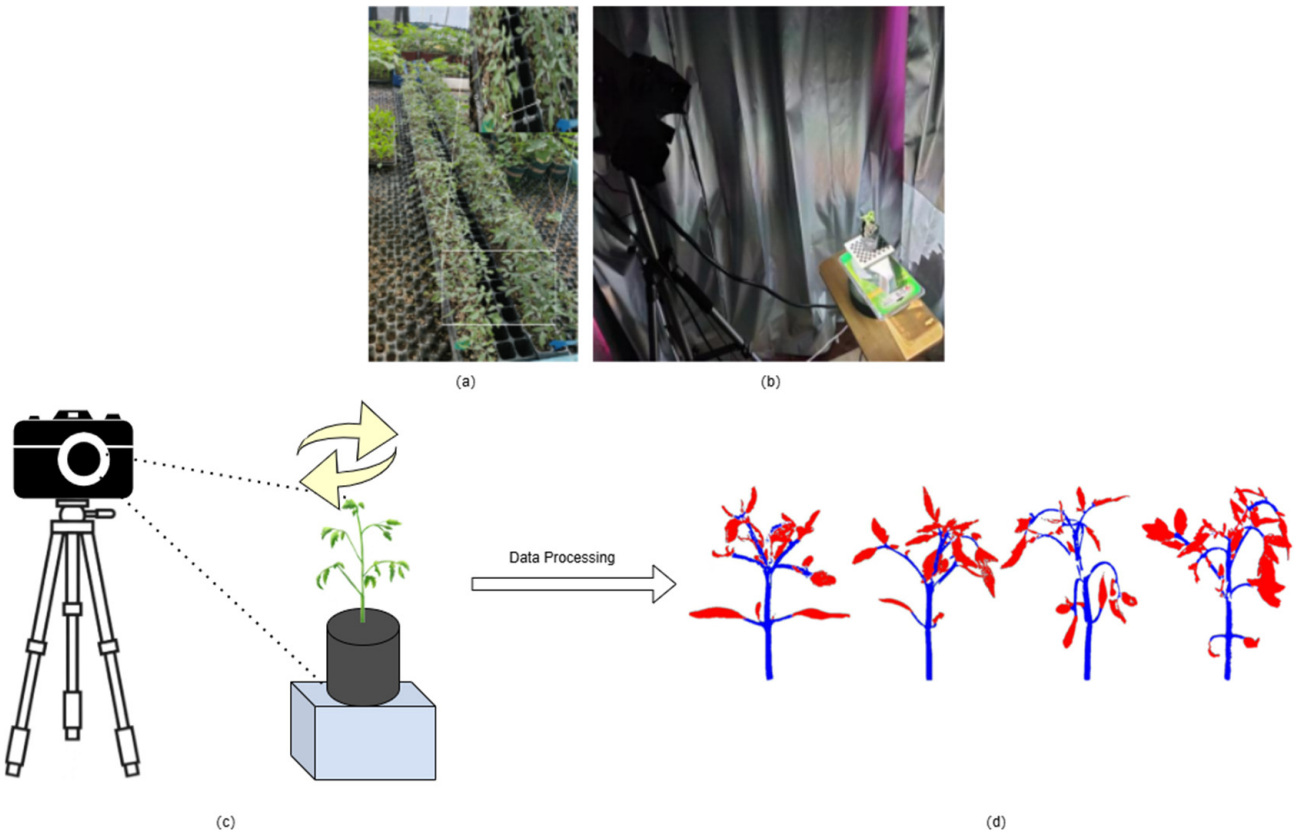


Figure 1. Image acquisition method. (**a**) Tomato plant sample Point cloud data acquisition; (**b**) Point cloud data acquisition scene; (**c**) Point cloud data acquisition; (**d**) Visualization of preprocessed image presentation.

2.2. Methodologies

2.2.1. Algorithmic Principles of the Red-Billed Blue Magpie Algorithm for Elite Strategy Optimization

The Elite Strategy-based Improved Red-billed Blue Magpie Optimization (ES-RBMO) Algorithm optimizes the red-billed blue magpie by simulating its foraging behavior in the hyper-parameter space of deep learning networks [23–25]. The elite strategy effectively avoids falling into a local optimum by retaining the current optimal solution and prioritizing the use of its information for subsequent searches.

Each individual (bird) represents a configuration of a deep learning model. In this paper, the population size was set at 50, assuming that the deep learning model has multiple hyperparameters, $\{\theta_1, \theta_2, \ldots, \theta_k\}$ the position of each individual can be represented as.

$$x_i = \left\{\theta_1^i, \theta_2^i, \ldots, \theta_k^i\right\}$$

where θ_j^i denotes the value of dimension j of the ith individual (solution) in the hyperparameter space (e.g., learning rate, number of network layers, etc.).

The fitness function is a quantitative form of the optimization objective, and we compute the fitness of an individual based on accuracy, recall, precision, F1 score, and loU. Let F_i denote the fitness value of the ith individual, we define the weighted fitness function as.

$$F_i = w_1 \cdot \text{Accuracy}_i + w_2 \cdot \text{Recall}_i + w_3 \cdot \text{Precision}_i + w_4 \cdot \text{F1}_i + w_5 \cdot \text{IoU}_i$$

where, $Accuracy_i$, $Recall_i$, $Precision_i$, $F1_i$, IoU_i denotes each performance indicator of the model corresponding to the ith individual, respectively. Individual corresponds to each performance metric of the model. w_1, w_2, w_3, w_4, w_5 is the weight of each indicator, setting the weighting ratio to 1:1:1:1:1:1 ensures that each performance indicator contributes equally to the overall evaluation, thus providing a fair approach to model evaluation. Such a weight setting is unbiased among different indicators and is suitable for scenarios in which there is no clear preference for a particular indicator. And it satisfies $w_1 + w_2 + w_3 + w_4 + w_5 = 1$.

The core of the red-billed blue magpie optimization algorithm is to search for the best solution by updating the individuals with the following update rules:

$$x_i^{t+1} = x_i^t + \alpha(x_{best}^t - x_i^t) + \beta\left(mean\left(x_{group}^t\right) - x_i^t\right)$$

x_i^{t+1} indicates the ith Individuals in position $t + 1$. x_i^t indicates the ith Individuals in position t. x_{best}^t is the best adapted elite individual in the current population. $mean\left(x_{group}^t\right)$ is the positional mean of all individuals in the current population. α and β are the step coefficients for global and local search, respectively.

Elite strategy by selecting the best adapted N_{elite} individuals, preserves its solution and passes it on directly to the next generation. This mechanism helps the algorithm to keep the search in the neighborhood of the optimal solution and avoid falling into local optima.

While local search performs a meticulous search in the vicinity of the current solution to optimize the solution quality through minor adjustments, jump search boldly explores a larger search space by randomly adjusting the position of individuals in order to find global optimal solutions that may have been overlooked [26,27]. With this strategy, the Red-billed Blue Magpie algorithm is not only able to accurately perform local optimization when dealing with point cloud data, but also maintains the vitality of the global search, ensuring that the algorithm performs well in point cloud alignment, shape recognition, and other complex tasks [28]. By retaining the elite individuals with the highest fitness, the algorithm ensures that the optimal solution is always retained during the iteration process, which improves the overall search efficiency and the quality of the solution [29].

Local search optimizes the solution by making small adjustments based on the fitness of the current individual position. For example, better solutions are explored within a neighborhood through mutation operations [30]:

$$x_i^{t+1} = x_i^t + \delta(x_{best}^t - x_i^t)$$

δ is a smaller constant that controls the trim step.

Jump search [31] appears to be more aggressively applied in the Red-billed Blue Magpie algorithm, which explores domains in the solution space far from the current population location by increasing the search step size, thus potentially discovering a better solution [32]. In each generation of the algorithm, we update the position of the population based on the fitness value of each individual and generate new individuals. In this process, we use an elite strategy, i.e., we keep the individuals with the best fitness as the elite, and these elite individuals will not be eliminated to ensure that the algorithm can maintain the inheritance of the optimal solution during the iteration process [33]. By applying this elite strategy-optimized red-billed blue magpie algorithm to point cloud data processing, the solution space can be explored and utilized more efficiently, and the efficiency and accuracy of point cloud alignment and shape optimization can be improved [34–38].

The termination conditions of the algorithm include the following three aspects: firstly, when the algorithm reaches the predetermined maximum number of iterations, the search process will be stopped automatically; secondly, if the change of the fitness of the optimal solution is less than a certain set threshold in a number of consecutive generations, it

indicates that the algorithm has converged, and the iteration will be terminated at this time as well [39–42]; lastly, if the value of the fitness of the optimal solution exceeds the predetermined threshold, it means that the algorithm has found a satisfactory solution and therefore will stop early [43,44]. With this optimization strategy, we apply the red-billed blue magpie algorithm to point cloud data processing to achieve more efficient and accurate search and matching.

2.2.2. Algorithmic Principles of 3DCNN

The 3DCNN incorporates the geometric and curvature properties of point cloud data to construct a four-layer deep learning architecture by encapsulating customized convolutional layers [45]. Encapsulate the convolutional layer CustomConv3D by combining curvature features and geometric features of the point cloud. The first convolutional layer of this network, conv1, uses a $3 \times 3 \times 3$ convolutional kernel for feature extraction from single-channel input data, which achieves the mapping from raw data to 64-dimensional feature space, and accelerates convergence and stabilizes the training through a batch normalization operation in the bn1 layer [46]. conv2 with the bn2 layer extends the feature dimensions to 128 dimensions, which enhances the abstract representation of the features. conv3 with the bn3 layers map features to 256 dimensions to capture higher-level structural information. The optional conv4 & bn4 layers achieve deeper feature extraction, enhancing 256-dimensional features to 512 dimensions, which significantly improves the model's ability to model and recognize complex spatial features [47], as shown in Figure 2. This method effectively extracts and abstracts rich geometric and structural information from the point cloud data by increasing the feature dimensions layer by layer, which significantly improves the model's modeling and recognition ability of complex 3D shapes; meanwhile, combined with batch normalization, it optimizes the training process, accelerates the convergence speed, and enhances the model's generalization performance.

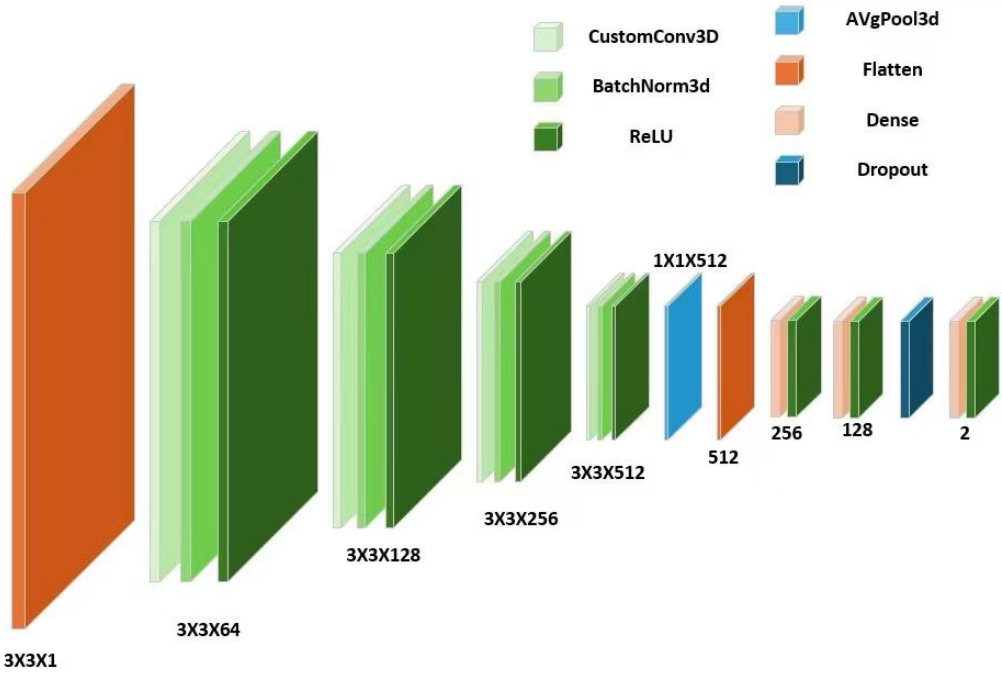


Figure 2. 3DCNN model hierarchical flowchart.

3. Results

3.1. Comparison Experiment

The experiment uses accuracy, recall, precision, loss rate, F1-Score, and Intersection over Union (IoU) to evaluate the training effectiveness of the model. The formulas are as follows.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{F1 Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$\text{IoU} = \frac{\text{A} \cap \text{B}}{\text{A} \cup \text{B}}$$

TP stands for True Positive, which refers to the correct classification of positive predictions; FP stands for False Positive, which means incorrectly classifying negative predictions as positive; FN stands for False Negative, which means incorrectly classifying positive predictions as negative; TN stands for True Negative, which refers to the correct classification of negative predictions; A represents the actual true value region; B represents the region predicted by the model.

The ES-RMBO algorithm proposed in this paper shows significant superiority in stem-and-leaf segmentation tasks. In comparison with other mainstream algorithms, ES-RMBO achieves the optimal level in key performance metrics such as precision (0.965), recall (0.965), F1 score (0.965), IoU (0.965), and ACC (0.933), demonstrating higher segmentation accuracy and stability (Table 1). In contrast, although the UNet algorithm [48], AC-UNet algorithm [49], PointNet++ algorithm [50], PCNN algorithm [51], and DeepLabV3 algorithm [52] have also demonstrated good performance in segmentation tasks, they slightly underperform RMBO in terms of accuracy (Figure 3a), recall (Figure 3b), F1 score (Figure 3c), IoU (Figure 3d), and ACC (Figure 3e). Tomato plants exhibit unique growth characteristics, with upright and regular stems and dense, diverse leaves featuring relatively complex structures. The RMBO algorithm effectively integrates the geometric information of point cloud data, enabling it to accurately identify and distinguish these complex structural features. By striking a better balance between precision and recall, the RMBO algorithm significantly reduces the occurrence of misclassification, thereby enhancing the reliability and accuracy of segmentation results.

Table 1. ES-RMBO Comparison Experiments.

	P	**R**	**F1**	**IoU**	**ACC**
AC-UNet	0.916	0.903	0.915	0.892	0.886
UNet	0.912	0.901	0.912	0.894	0.864
PointNet++	0.927	0.911	0.896	0.881	0.873
PCNN	0.906	0.890	0.904	0.909	0.898
DeepLabV3	0.877	0.904	0.885	0.821	0.853
ES-RMBO	0.965	0.965	0.965	0.965	0.933

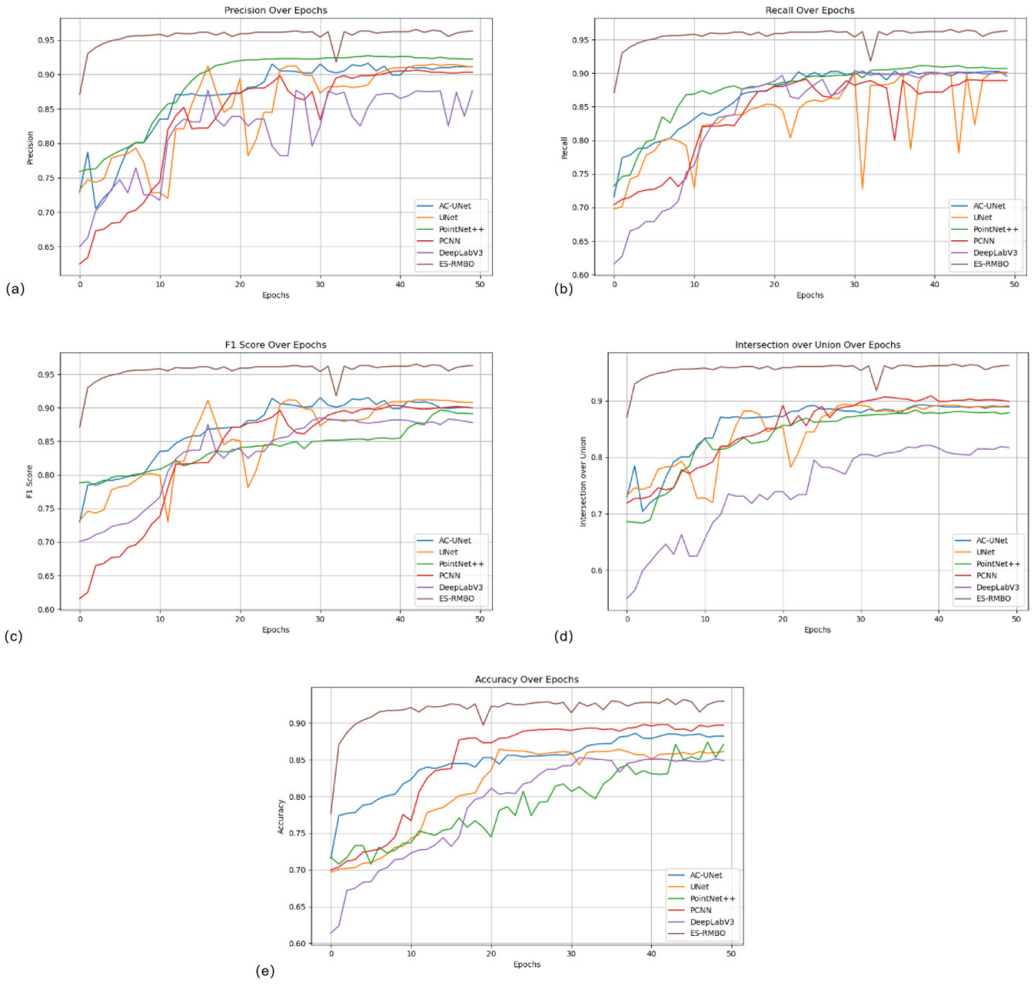


Figure 3. Comparison of each data of ES-RMBO comparison test. (**a**) Accuracy; (**b**) Recall rate; (**c**) F1 score; (**d**) IoU; (**e**) ACC.

In order to comprehensively evaluate the computational efficiency and resource usage of the ES-RBMO algorithm, we compare it with the models in Table 1, and the results, as shown in Table 2, show that ES-RBMO has a significant advantage in a number of metrics. The training time of ES-RBMO is 13.1 h, which is about 7% and 45% less than that of UNet, PointNet++ and DeepLabV3, respectively. The training time of ES-RBMO is 13.1 h, which is about 7%, 15% and 45% less than that of UNet, PointNet+ and DeepLabV3, respectively. The model parameter size is only 5.8 million, which is about 14% of DeepLabV3, and the segmentation accuracy (IoU of 0.965) is still high while reducing the computational complexity. In addition, ES-RBMO consumes 5.9 GB of video memory, which is significantly lower than 8.2 GB of PointNet++ and 10.4 GB of DeepLabV3, and is more suitable for resource-constrained scenarios. The computational efficiency of ES-RBMO is 17.04, which is significantly higher than 9.56 of UNet, 10.29 of PointNet++, and 5.20 of DeepLabV3. These results fully demonstrate the superiority of ES-RBMO in terms of runtime, resource usage and overall computational efficiency.

Table 2. Comparison of runtime and resource usage.

	Training Time (Hours)	Parameter Size (Millions)	Video Memory Usage (GB)	Computational Efficiency (E)
AC-UNet	14.0	22.8	6.7	9.84
UNet	14.2	23.4	6.5	9.56
PointNet++	15.5	4.5	8.2	10.29
PCNN	15.0	15.2	7.8	11.96
DeepLabV3	23.8	40.6	10.4	5.20
ES-RMBO	13.1	5.8	5.9	17.04

3.2. Ablation Experiment

In order to verify the effectiveness of ES-RMBO algorithm in tomato stem and leaf segmentation task, we conducted ablation experiments to compare ES-RMBO with other unoptimized algorithms, including the original 3DCNN and the addition of the unoptimized RMBO algorithm. Table 3 demonstrates the comparison of the performance of different algorithms on the tomato stem and leaf segmentation task. As can be seen from the table, the ES-RMBO algorithm achieves optimal results in all key performance metrics of precision (0.965), recall (0.965), F1 score (0.965), IoU (0.965), and ACC (0.933), which proves its superiority in the tomato stem-and-leaf segmentation task. The results of the ablation experiments show that several values of precision (Figure 4a), recall (Figure 4b), F1 value (Figure 4c), IoU (Figure 4d) and ACC (Figure 4e) of the unoptimized algorithm are not as good as the ES-RMBO algorithm in the tomato stem and leaf segmentation task, which suggests that the ES-RMBO algorithm is able to efficiently process the tomato point cloud data, improve segmentation precision and stability, providing an efficient and accurate solution for tomato growth monitoring, variety identification and phenotypic analysis.

Table 3. ES-RMBO ablation experiments.

	P	R	F1	IoU	ACC
3DCNN	0.876	0.877	0.868	0.871	0.793
RMBO	0.913	0.915	0.924	0.928	0.818
ES-RMBO	0.965	0.965	0.965	0.965	0.933

From the tabular data, it can be seen that each of the three models has its own characteristics in terms of training time, parameter size, video memory occupation and computational efficiency. As shown in Table 4, 3DCNN has a parameter size of 10.8 million, a memory usage of 6.7 GB, and a computational efficiency of 9.37, which is a more basic performance. RMBO has a training time of 10.2 h, a memory usage of 6.5 GB, a computational efficiency of 11.36, and a parameter size of 11.4 million, which is slightly larger than that of 3DCNN and shows a certain degree of performance improvement. ES-RMBO has the most outstanding performance, despite the longest training time (13.1 h), it achieves the highest computational efficiency (17.04) with the smallest parameter size (5.8 million) and video memory usage (5.9 GB), which indicates that it has a significant advantage in balancing the use of resources and performance, and is the model with the best overall performance.

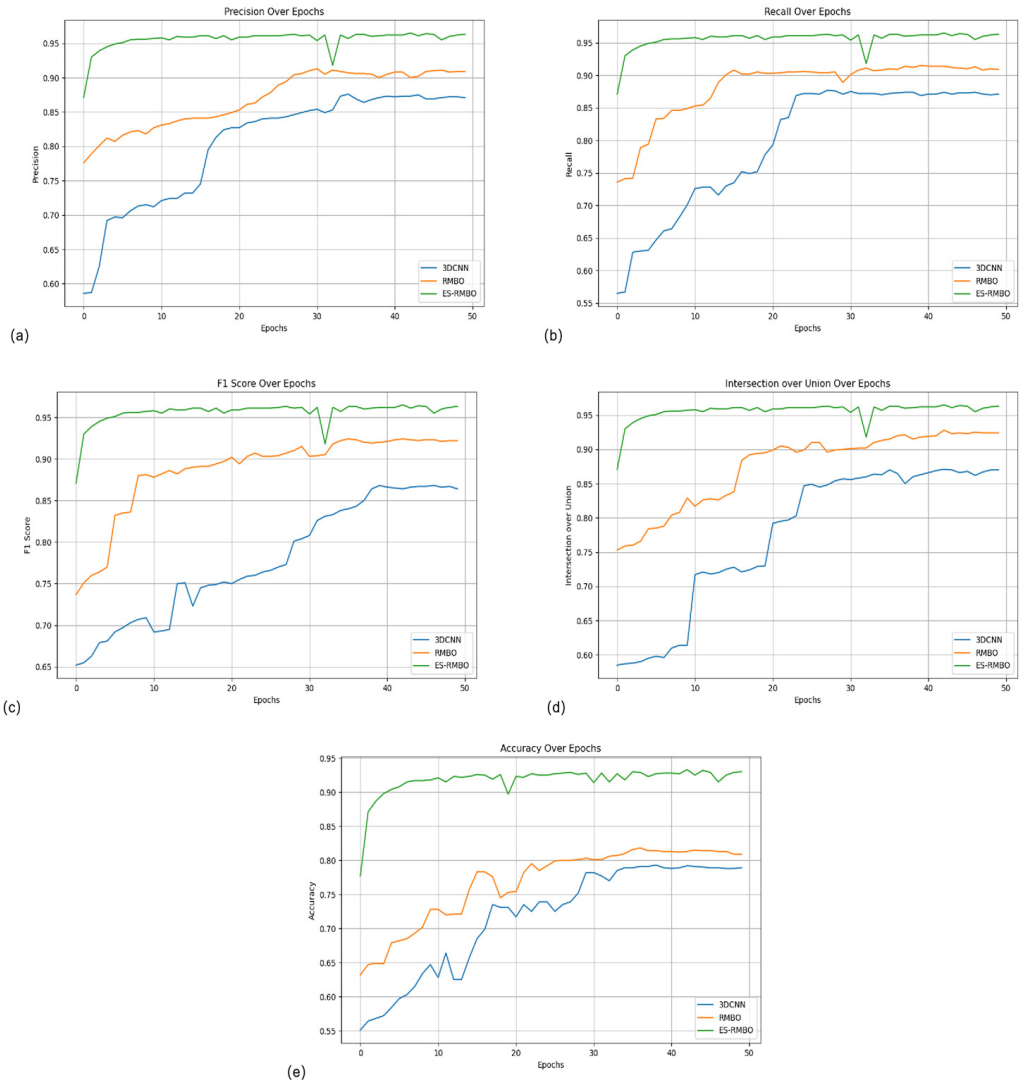


Figure 4. Comparison of each data of ES-RMBO ablation experiments. (**a**) Accuracy; (**b**) Recall rate; (**c**) F1 score; (**d**) IoU; (**e**) ACC.

Table 4. Comparison of runtime and resource usage.

	Training Time (Hours)	Parameter Size (Millions)	Video Memory Usage (GB)	Computational Efficiency (E)
3DCNN	13.0	10.8	6.7	9.37
RMBO	10.2	11.4	6.5	11.36
ES-RMBO	13.1	5.8	5.9	17.04

3.3. Phenotypic Parameter Measurement Results and Analysis

The prediction of phenotypic parameters on the segmented point cloud data is conducted using classic algorithms, including key parameters such as plant height, stem diameter, leaf area, and leaf inclination angle. Specific methods involve using the RANSAC algorithm for linear fitting of the main stem to measure plant height and stem diameter, employing a region growing algorithm to estimate leaf area and leaf inclination angle, and further optimizing the calculation of phenotypic parameters through the greedy projection triangulation algorithm. Measurement results and analysis were taken from 100 tomato plants, comparing the manual measurements with the extracted values of 3D point cloud parameters, including plant height R^2 = 0.932, RMSE = 0.511, MAPE = 1.970, (Figure 5a);

stem thickness R^2 = 0.741, RMSE = 0.135, MAPE = 4.299, (Figure 5b); leaf area R^2= 0.938, RMSE = 0.989, MAPE = 4.365, (Figure 5c); leaf inclination R^2 = 0.935, RMSE = 3.628, MAPE = 5.531, (Figure 5d). The experimental data showed that the results indicated a high correlation between the measured results and the real data for the measured leaf data. The low R^2 for stem thickness prediction stems from the model's lack of ability in extracting features related to stem thickness. In addition, measurement noise is one of the influencing factors. The high RMSE for blade tilt reflects the difficulty of the model in capturing the complexity of blade angle changes, especially in the face of high variability in the data.

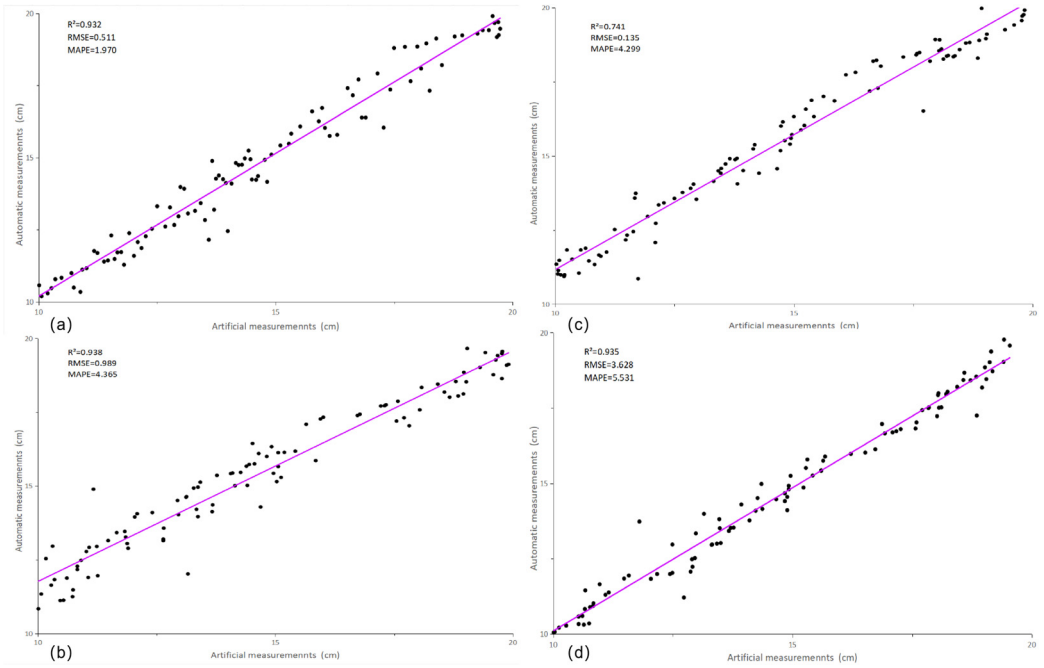


Figure 5. Measurements of phenotypic parameters. (**a**) Plant height; (**b**) Stem thickness; (**c**) Leaf area; (**d**) Leaf inclination angle.

4. Discussion

The ES-RBMO algorithm proposed in this paper demonstrates excellent adaptability and robustness to the point cloud characteristics exhibited by tomato plants at different growth and development stages and under diverse environmental conditions. By extracting the geometric and curvature features of the point cloud in a hierarchical manner, the algorithm accelerates the convergence speed and the quality of the solution, and improves the accuracy and stability of the segmentation. The experimental results show that the algorithm can efficiently perform stem and leaf segmentation on tomato point cloud data processing, and achieves the optimal level of key performance metrics, such as precision, recall, F1 score, IoU, and ACC, demonstrating higher segmentation precision and stability (Figure 6a).

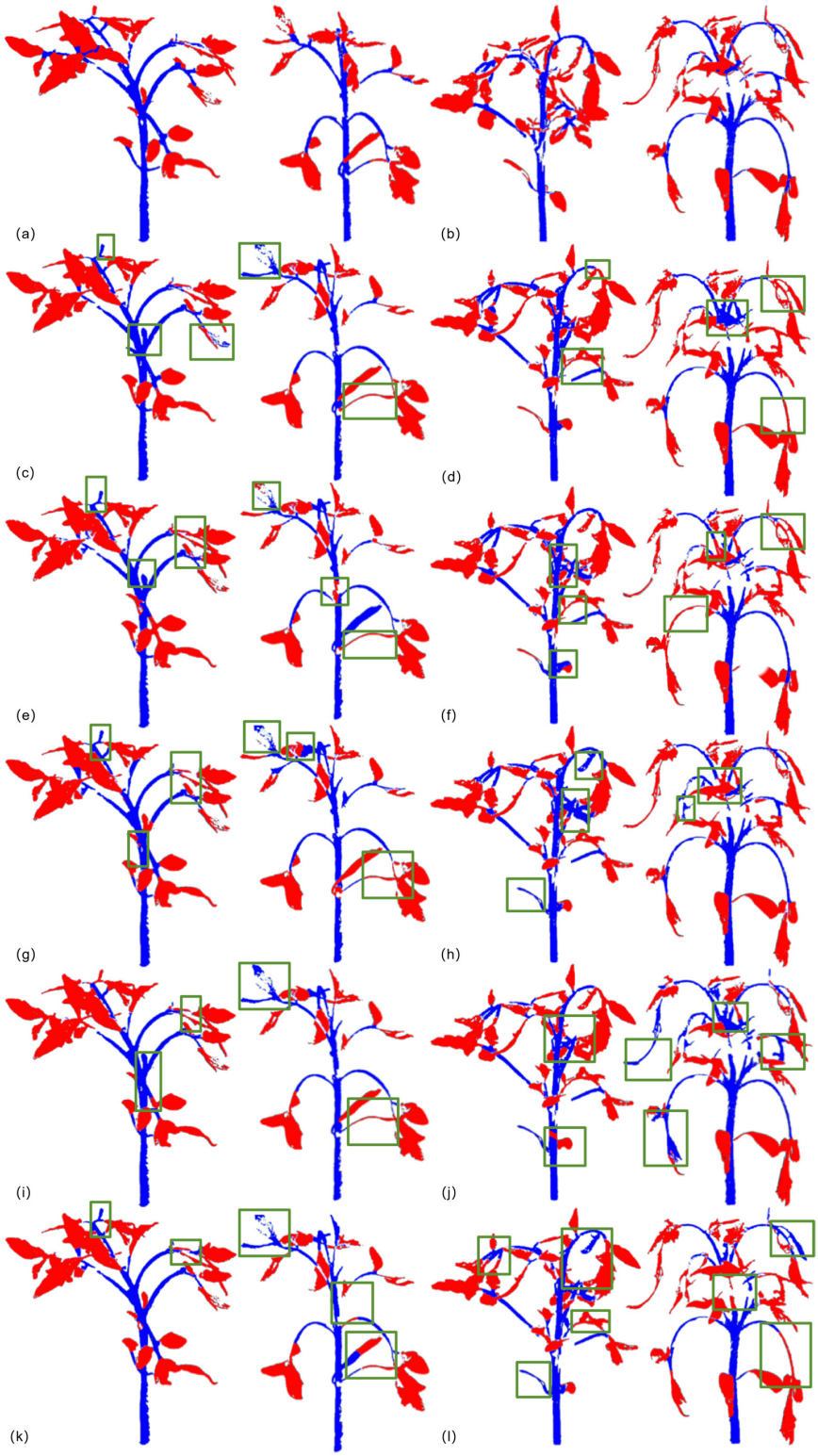


Figure 6. Point cloud results of tomato plants with different growth conditions. The green boxes in the figure indicate the undetected sites of the other models compared to ES-RMBO. (**a**) Normal tomato plants identified by ES-RMBO; (**b**) More complex tomato plants identified by ES-RMBO; (**c**) Normal tomato plants identified by AC-UNet; (**d**) More complex tomato plants identified by AC-UNet; (**e**) Normal tomato plants identified by UNet; (**f**) More complex tomato plants identified by UNet; (**g**) Normal tomato plants identified by PointNet++; (**h**) More complex tomato plants identified by PointNet++; (**i**) Normal tomato plants identified by PCNN; (**j**) More complex tomato plants identified by PCNN; (**k**) Normal tomato plants identified by DeepLabV3; (**l**) More complex tomato plants identified by DeepLabV3.

AC-UNet and UNet, as classical convolutional neural network models, have better segmentation accuracy and computational efficiency, but are prone to feature confusion or boundary blurring when dealing with complex scenes (e.g., dense targets and severe occlusion) (Figure 6c–f). PointNet++ and PCNN have a strong capability of capturing spatial-geometric features with the introduction of point-cloud processing and hierarchical feature extraction mechanisms (Figure 6g–j). geometric features, but are still insufficient in discriminating fine-grained features in occluded regions (Figure 6g–j). DeepLabV3 excels in capturing global contextual information with the help of null convolution and multi-scale feature extraction, but its large parameter sizes and high memory requirements limit its application in resource-constrained environments (Figure 6k,l). In contrast, ES-RMBO combines an efficient deep learning architecture and optimization strategy, which offers significant advantages in parameter size, video memory footprint and computational efficiency, while effectively addressing dense growth and occlusion problems. However, ES-RMBO may still suffer from insufficient detail capture in scenes with extreme occlusions and complex geometries. This dense and interlaced growth pattern leads to the difficulty of traditional edge detection algorithms in accurately recognizing leaf edges, thus affecting the accuracy of the segmentation results (Figure 6b).

5. Conclusions

In this paper, we propose a stem-and-leaf segmentation method based on the improved optimization algorithm for red-billed blue magpies, and verify its excellent performance and robustness in point cloud data of different densities. The method significantly improves the accuracy and stability of stem-and-leaf segmentation by fusing the geometric features, spatial distribution information and deep learning model of point cloud data. Through dynamic volume compression, the aggregation of the point cloud is enhanced, which effectively improves the accuracy of the feature extraction process. For different point cloud densities, the algorithm is able to accurately capture stem and leaf features, optimize local details, and significantly reduce misclassification. During the processing of tomato plants, ES-RMBO demonstrates strong stability and reliability, and is able to accurately distinguish stems from leaves. Through hierarchical processing and geometric feature computation, the method successfully copes with the unique hierarchical structure of tomato plants, further enhancing the segmentation effect. The algorithm achieves optimal results in key performance metrics such as precision, recall, F1 score, IoU, and ACC, which verifies its high adaptability in plant growth stages. The algorithm proposed in this paper can effectively deal with complex plant structures and provides solid technical support for plant growth monitoring and automated segmentation tasks in precision agriculture, and its applicability in a wide range of plant species and agricultural application scenarios can be expanded in the future by further optimizing the algorithm's generalization ability.

Author Contributions: Conceptualization, L.Z. and Z.H.; methodology, Z.Y.; software, H.Y. (Helong Yu) and L.Z.; validation, H.Y. (Helong Yu) and L.Z.; formal analysis, S.Y.; investigation, Z.H. and B.Y.; data curation, Z.H. and S.Z.; writing—original draft preparation, H.Y. and Y.L.; writing—review and editing, L.Z. and Z.H.; supervision, H.Y. (Han Yang); project administration, L.Z.; translation, Y.L.; proofreading, X.Z.; Image beautification, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by science and technology development plan project of Science and Technology Department of Jilin Province, project name: A Trusted Traceability System for Smart Agriculture Based on Blockchain Technology, project No: 20220202036NC.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: For experimental data in this paper, please contact the corresponding author for assistance.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Boogaard, F.; Henten, E.; Kootstra, G. The added value of 3D point clouds for digital plant phenotyping—A case study on internode length measurements in cucumber. *Biosyst. Eng.* **2023**, *234*, 1–12. [CrossRef]
2. Alighaleh, P.; Mesri Gundoshmian, T.; Alighaleh, S.; Rohani, A. Feasibility and reliability of agricultural crop height measurement using the laser sensor array. *Inf. Process. Agric.* **2024**, *11*, 228–236. [CrossRef]
3. Anshori, M.F.; Dirpan, A.; Sitaresmi, T.; Rossi, R.; Farid, M.; Hairmansis, A.; Purwoko, B.; Suwarno, W.B.; Nugraha, Y. An overview of image-based phenotyping as an adaptive 4. 0 technology for studying plant abiotic stress: A bibliometric and literature review. *Heliyon* **2023**, *9*, e21650. [CrossRef] [PubMed]
4. Dong, Q.; Sun, L.; Han, T.; Cai, M.; Gao, C. PestLite: A Novel YOLO-Based Deep Learning Technique for Crop Pest Detection. *Agriculture* **2024**, *14*, 228. [CrossRef]
5. Li, R.; Li, Y.; Qin, W.; Abbas, A.; Li, S.; Ji, R.; Wu, Y.; He, Y.; Yang, J. Lightweight Network for Corn Leaf Disease Identification Based on Improved YOLO v8s. *Agriculture* **2024**, *14*, 220. [CrossRef]
6. Wang, Y.; Wu, M.; Shen, Y. Identifying the Growth Status of Hydroponic Lettuce Based on YOLO-EfficientNet. *Plants* **2024**, *13*, 372. [CrossRef] [PubMed]
7. Zhao, L.; Zhao, Y.; Liu, T.; Deng, H. A Weakly Supervised Semantic Segmentation Model of Maize Seedlings and Weed Images Based on Scrawl Labels. *Sensors* **2023**, *23*, 9846. [CrossRef] [PubMed]
8. Sun, Y.; Guo, X.; Yang, H. Win-Former: Window-Based Transformer for Maize Plant Point Cloud Semantic Segmentation. *Agronomy* **2023**, *13*, 2723. [CrossRef]
9. Luo, L.; Jiang, X.; Yang, Y.; Samy, E.R.; Lefsrud, M.; Hoyos-Villegas, V.; Sun, S. Eff-3DPSeg: 3D organ-level plant shoot segmentation using annotation-efficient point clouds. *arXiv* **2022**, arXiv:2212.10263.
10. Wang, Y.; Liu, Q.; Yang, J.; Ren, G.; Wang, W.; Zhang, W.; Li, F. A Method for Tomato Plant Stem and Leaf Segmentation and Phenotypic Extraction Based on Skeleton Extraction and Supervoxel Clustering. *Agronomy* **2024**, *14*, 198. [CrossRef]
11. Morteza, G.; Kevin, W.; Corke, F.M.; Tiddeman, B.; Liu, Y.; Doonan, J.H. Deep Segmentation of Point Clouds of Wheat. *Front. Plant Sci.* **2021**, *12*, 608732.
12. Yonatan, L.; Ofri, R.; Merav, A. Dissecting the roles of supervised and unsupervised learning in perceptual discrimination judgments. *J. Neurosci.* **2020**, *41*, 757–765.
13. Yao, D.; Chuanchuan, Y.; Hao, C.; Yan, W.; Li, H. Low-complexity point cloud denoising for LiDAR by PCA-based dimension reduction. *Opt. Commun.* **2021**, *482*, 126567.
14. Alkadri, F.M.; Yuliana, Y.; Agung, C.R.M.; Rahman, M.A.; Hein, C. Enhancing preservation: Addressing humidity challenges in Indonesian heritage buildings through advanced detection methods point cloud data. *Results Eng.* **2024**, *24*, 103292. [CrossRef]
15. Jing, R.; Shao, Y.; Zeng, Q.; Liu, Y.; Wei, W.; Gan, B.; Duan, X. Multimodal feature integration network for lithology identification from point cloud data. *Comput. Geosci.* **2025**, *194*, 105775. [CrossRef]
16. Chen, T.; Ying, X. FPSMix: Data augmentation strategy for point cloud classification. *Front. Comput. Sci.* **2024**, *19*, 192701. [CrossRef]
17. Miao, T.; Zhu, C.; Xu, T.; Yang, T.; Li, N.; Zhou, Y.; Deng, H. Automatic stem-leaf segmentation of maize shoots using three-dimensional point cloud. *Comput. Electron. Agric.* **2021**, *187*, 106310. [CrossRef]
18. Shen, F.; Lu, Z.-M.; Lu, Z.; Wang, Z. Dual semantic-guided model for weakly-supervised zero-shot semantic segmentation. *Multimed. Tools Appl.* **2021**, *81*, 5443–5458. [CrossRef]
19. Wu, J.; Sun, M.; Xu, H.; Jiang, C.; Ma, W.; Zhang, Q. Class agnostic and specific consistency learning for weakly-supervised point cloud semantic segmentation. *Pattern Recognit.* **2025**, *158*, 111067. [CrossRef]
20. Samoaa, P.; Aronsson, L.; Longa, A.; Leitner, P.; Chehreghani, M.H. A unified active learning framework for annotating graph data for regression task. *Eng. Appl. Artif. Intell.* **2024**, *138*, 109383. [CrossRef]
21. Bicheng, S.; Peng, Z.; Liang, D.; Li, X. Active deep image clustering. *Knowl.-Based Syst.* **2022**, *252*, 109346.
22. Sun, R.; Guo, S.; Guo, J.; Li, W.; Zhang, X.; Guo, X.; Pan, Z. GraphMoCo: A graph momentum contrast model for large-scale binary function representation learning. *Neurocomputing* **2024**, *575*, 127273. [CrossRef]
23. Shengwei, F.; Ke, L.; Haisong, H.; Ma, C.; Fan, Q.; Zhu, Y. Red-billed blue magpie optimizer: A novel metaheuristic algorithm for 2D/3D UAV path planning and engineering design problems. *Artif. Intell. Rev.* **2024**, *57*, 134.
24. Hassen, L.; Ali, L.; Slim, B.; Kariri, E. Joint filter and channel pruning of convolutional neural networks as a bi-level optimization problem. *Memetic Comput.* **2024**, *16*, 71–90.

25. Baljon, M. A Framework for Agriculture Plant Disease Prediction using Deep Learning Classifier. *Int. J. Adv. Comput. Sci. Appl. IJACSA* **2023**, *14*, 1098–1111. [CrossRef]
26. Xiaodan, L.; Zijian, Z. A Whale Optimization Algorithm with Convergence and Exploitability Enhancement and Its Application. *Math. Probl. Eng.* **2022**, *2022*, 2904625.
27. Anderson, J.P.; Stephens, D.W.; Dunbar, S.R. Saltatory search: A theoretical analysis. *Behav. Ecol.* **1997**, *8*, 307–317. [CrossRef]
28. Fergany EA, A.; Agwa, M.A. Red-Billed Blue Magpie Optimizer for Electrical Characterization of Fuel Cells with Prioritizing Estimated Parameters. *Technologies* **2024**, *12*, 156. [CrossRef]
29. Wang, P.; Liu, Z.; Wang, Z.; Zhao, Z.; Yang, D.; Yan, W. Graph generative adversarial networks with evolutionary algorithm. *Appl. Soft Comput.* **2024**, *164*, 111981. [CrossRef]
30. Zhang, M.; Wang, H.; Cui, Z.; Chen, J. Hybrid multi-objective cuckoo search with dynamical local search. *Memetic Comput.* **2018**, *10*, 199–208. [CrossRef]
31. Li, C.; Priemer, R.; Cheng, K.H. Optimization by random search with jumps. *Int. J. Numer. Methods Eng.* **2004**, *60*, 1301–1315. [CrossRef]
32. Shan, D.; Zhang, X.; Shi, W.; Li, L. Neural Architecture Search for a Highly Efficient Network with Random Skip Connections. *Appl. Sci.* **2020**, *10*, 3712. [CrossRef]
33. Gan, W.; Li, H.; Hao, P. Many-objective optimization algorithm based on the similarity principle and multi-mechanism collaborative search. *J. Supercomput.* **2024**, *81*, 124. [CrossRef]
34. Fang, G.; Weibin, Z.; Guofu, L.; Zhang, X.; Luo, L.; Wu, Y.; Guo, P. A point cloud registration method based on multiple-local-feature matching. *Optik* **2023**, *295*, 171511.
35. Jingtao, W.; Changcai, Y.; Lifang, W.; Chen, R. CSCE-Net: Channel-Spatial Contextual Enhancement Network for Robust Point Cloud Registration. *Remote Sens.* **2022**, *14*, 5751. [CrossRef]
36. Wang, L. High-precision point cloud registration method based on volume image correlation. *Meas. Sci. Technol.* **2024**, *35*, 035024. [CrossRef]
37. Chuang, T.Y.; Jaw, J.J. Multi-feature registration of point clouds. *Remote Sens.* **2017**, *9*, 281. [CrossRef]
38. Yu, F.; Chen, Z.; Cao, J.; Jiang, M. Redundant same sequence point cloud registration. *Vis. Comput.* **2023**, *40*, 7719–7730. [CrossRef]
39. Xu, S. *An Introduction to Scientific Computing with Matlab and Python Tutorials*; CRC Press: Boca Raton, FL, USA, 2021.
40. Garcia, H.A.; Zhu, W. Building an Accessible and Flexible Multi-User Robotic Simulation Framework with Unity-MATLAB Bridge. *Computers* **2024**, *13*, 282. [CrossRef]
41. Gasmi, K.; Hasnaoui, S. Dataflow-based automatic parallelization of MATLAB/Simulink models for fitting modern multicore architectures. *Clust. Comput.* **2024**, *27*, 6579–6590. [CrossRef]
42. Yang, D. An improved particle swarm optimization algorithm for parameter optimization. *Comput. Informatiz. Mech. Syst.* **2022**, *5*, 35–38.
43. Mohan, B.G.; Kumar, P.R.; Elakkiya, R. Enhancing pre-trained models for text summarization: A multi-objective genetic algorithm optimization approach. *Multimed. Tools Appl.* **2024**, 1–17. [CrossRef]
44. Chaudhury, A. Multilevel Optimization for Registration of Deformable Point Clouds. *IEEE Trans. Image Process.* **2020**, *29*, 8735–8746. [CrossRef]
45. Tianyuan, L.; Jiacheng, W.; Xiaodi, H.; Lu, Y.; Bao, J. 3DSMDA-Net: An improved 3DCNN with separable structure and multi-dimensional attention for welding status recognition. *J. Manuf. Syst.* **2021**, *62*, 811–822.
46. Peyman, A.; Keyhan, G.; Atieh, A.; Deb, P.; Moradkhani, H. Bayesian Multi-modeling of Deep Neural Nets for Probabilistic Crop Yield Prediction. *Agric. For. Meteorol.* **2022**, *314*, 108773.
47. Zhu, G.; Zhang, L.; Shen, P.; Song, J.; Shah, S.A.; Bennamoun, M. Continuous Gesture Segmentation and Recognition Using 3DCNN and Convolutional LSTM. *IEEE Trans. Multimed.* **2019**, *21*, 1011–1021. [CrossRef]
48. Zhihua, D.; Peiliang, G.; Baohua, Z.; Zhang, D.; Yan, J.; He, Z.; Zhao, S.; Zhao, C. Maize crop row recognition algorithm based on improved UNet network. *Comput. Electron. Agric.* **2023**, *210*, 107940.
49. Yi, X.; Wang, J.; Wu, P.; Wang, G.; Mo, L.; Lou, X.; Liang, H.; Huang, H.; Lin, E.; Maponde, B.T. AC-UNet: An improved UNet-based method for stem and leaf segmentation in Betula luminifera. *Front. Plant Sci.* **2023**, *14*, 1268098. [CrossRef] [PubMed]
50. Liu, B.; Chen, S.; Huang, H.; Tian, X. Tree Species Classification of Backpack Laser Scanning Data Using the PointNet++ Point Cloud Deep Learning Method. *Remote Sens.* **2022**, *14*, 3809. [CrossRef]
51. Xiang, R. Image segmentation for whole tomato plant recognition at night. *Comput. Electron. Agric.* **2018**, *154*, 434–442. [CrossRef]
52. Zeng, W.; He, M. Rice disease segmentation method based on CBAM-CARAFE-DeepLabv3+. *Crop Prot.* **2024**, *180*, 106665. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.