agriculture

MDPI

Article

Improved YOLOv8 Model for Phenotype Detection of Horticultural Seedling Growth Based on Digital Cousin

Yuhao Song [1], Lin Yang [1,2,*], Shuo Li [1], Xin Yang [3], Chi Ma [1], Yuan Huang [1] and Aamir Hussain [4]

[1] College of Engineering, Huazhong Agricultural University, Wuhan 430070, China; alex-songyuhao@webmail.hzau.edu.cn (Y.S.); bshq@webmail.hzau.edu.cn (S.L.); 225187@webmail.hzau.edu.cn (C.M.); huangyuan@mail.hzau.edu.cn (Y.H.)
[2] Key Laboratory of Agricultural Equipment in Mid-Lower Yangtze River, Ministry of Agriculture and Rural Affairs, Wuhan 430070, China
[3] Leibniz Centre for Agricultural Landscape Research, 15374 Müncheberg, Germany; xin.yang@zalf.de
[4] Department of Informatics, Modeling, Electronics, and Systems, University of Calabria, 87036 Rende, Italy; aamir.hussain@dimes.unical.it
* Correspondence: lin.yang@hzau.edu.cn; Tel.: +86-18186647992

Abstract: Crop phenotype detection is a precise way to understand and predict the growth of horticultural seedlings in the smart agriculture era to increase the cost-effectiveness and energy efficiency of agricultural production. Crop phenotype detection requires the consideration of plant stature and agricultural devices, like robots and autonomous vehicles, in smart greenhouse ecosystems. However, collecting the imaging dataset is a challenge facing the deep learning detection of plant phenotype given the dynamic changes among leaves and the temporospatial limits of camara sampling. To address this issue, digital cousin is an improvement on digital twins that can be used to create virtual entities of plants through the creation of dynamic 3D structures and plant attributes using RGB image datasets in a simulation environment, using the principles of the variations and interactions of plants in the physical world. Thus, this work presents a two-phase method to obtain the phenotype of horticultural seedling growth. In the first phase, 3D Gaussian splatting is selected to reconstruct and store the 3D model of the plant with 7000 and 30,000 training rounds, enabling the capture of RGB images and the detection of the phenotypes of the seedlings, overcoming temporal and spatial limitations. In the second phase, an improved YOLOv8 model is created to segment and measure the seedlings, and it is modified by adding the LADH, SPPELAN, and Focaler-ECIoU modules. Compared with the original YOLOv8, the precision of our model is 91%, and the loss metric is lower by approximately 0.24. Moreover, a case study of watermelon seedings is examined, and the results of the 3D reconstruction of the seedlings show that our model outperforms classical segmentation algorithms on the main metrics, achieving a 91.0% mAP50 (B) and a 91.3% mAP50 (M).

Keywords: smart agriculture; digital cousin; plant phenotype; 3D reconstruction; image segmentation

check for updates

Academic Editors: Chenglin Wang, Lufeng Luo, Juntao Xiong and Xiangjun Zou

Received: 22 November 2024
Revised: 23 December 2024
Accepted: 23 December 2024
Published: 26 December 2024

Citation: Song, Y.; Yang, L.; Li, S.; Yang, X.; Ma, C.; Huang, Y.; Hussain, A. Improved YOLOv8 Model for Phenotype Detection of Horticultural Seedling Growth Based on Digital Cousin. *Agriculture* **2025**, *15*, 28. https://doi.org/10.3390/agriculture15010028

Copyright: © 2024 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

Crop phenotype detection is widely used to digitally and precisely identify and predict the growth status of a crop, especial for horticultural seedlings in smart farming scenarios. As a key step in crop phenotype detection, the collection of datasets of crop images mainly relies on manual mobile shooting and fixed monitoring equipment, such as fixed-point cameras, but these methods are often affected by spatial and temporal limitations and cannot comprehensively cover the whole crop growth process [1]. For

example, in the field of factory planting, traditional methods cannot take continuous, multiangle photos of seedlings at different growth stages and under different environmental conditions, thus making it difficult to obtain comprehensive data [2]. In actual production, phenotypic detection is an important indicator of crop growth and health [3]. In the process of phenotypic detection, in order to obtain detailed textural and structural information on plants from all angles and to improve the efficiency of plant phenotype measurements, researchers have often used 3D reconstruction detection [4]. And, technology for plant detection and segmentation was successfully developed for crop seeding reconstruction via machine learning [5]. A large number of images are often used in this process, which puts high demands on the size of the datasets. Meanwhile, in order to facilitate analysis, researchers have tended to perform the semantic segmentation of images to extract the required seedling images from the complex background. While the current mainstream semantic segmentation techniques tend to focus only on the segmentation features at specific growth stages of the crop (e.g., seedling or maturity stage, usually 2–3 days), the features (e.g., shape, color, and size) of the target objects (e.g., seedlings and branches) remain relatively stable over a short period of time. However, the morphology of a crop changes extremely rapidly during the early stages of growth, for example, in the case of watermelon seedlings, whose germination stage lasts 6–11 days, and its different parts (e.g., leaves and shoots) undergo significant changes in shape, size, count, and texture as displayed by the 3D cloud points with the application of Mask-CNN [6]. These rapid changes make it difficult to accurately classify specific targets, and frequent model iterations are required if the entire process of crop growth is to be analyzed. This not only requires a huge number of datasets but also places extreme demands on the training speed of the models used for semantic segmentation.

In order to solve the problem caused by insufficient datasets, simulation–real hybrid data integration consisting of physical and virtual data is a new trend [7]. With the help of digital cousin technology [8], the 3D reconstruction of crops can be achieved by simulating different lighting conditions and physical environments in virtual environments, which further enriches the dataset and improves the accuracy of the analysis. Traditional 3D reconstruction algorithms can be categorized into active and passive methods according to whether the light source actively illuminates the object or not to be captured by a sensor [9]. Active reconstruction methods mainly include the structured light method [10], time-of-flight (TOF) method [11], and triangulation method [12]. Although the structured light method and triangulation method have high accuracy in 3D reconstruction at close range, they are susceptible to interference by ambient light and are not effective in long-distance reconstruction. The time-of-flight (TOF) method maintains high accuracy at long distances, but its high cost and low resolution limit its application range. Passive reconstruction methods are divided into multiocular visual reconstruction [13] and monocular visual reconstruction [14]. The multiocular visual reconstruction method uses multiple cameras, which can improve the accuracy of reconstruction to a certain extent, but it also increases its computation amount and cost. Although monocular vision is the least expensive and requires the smallest amount of computation, in the actual reconstruction process, it often requires a large number of multiangle images to complete the reconstruction of a scene, which is more demanding on the dataset [15]. Moreover, as an efficient method of 3D reconstruction and rendering for such sim–real hybrid imaging datasets, 3D Gaussian splatting achieves the real-time rendering of the radiation field and enables SOTA-level visualization in a short period of time [16]. This opens up new possibilities for the production of mixed reality and virtual reality datasets. By taking dozens of crop photos, the operator can obtain very-high-quality 3D images in about 30 min. Once a high-fidelity 3D model of the plant is obtained, the dataset can be extended by introducing simulated environmental factors

(e.g., natural light, greenhouse conditions, and other external variations) to improve the accuracy of the phenotypic measurements.

In terms of semantic segmentation of images, traditional image segmentation methods based on computer vision have been widely used to obtain phenotypic seedling data, but they often require a large number of cumbersome parameter adjustments in the recognition and segmentation processes. In addition, the segmentation effect may be greatly reduced when dealing with more complex scenes or in high-light environments [17]. In contrast, the emergence of deep learning techniques has revolutionized image segmentation and has attracted the attention of an increasing number of researchers. Deep learning methods can automatically learn features from large amounts of data through end-to-end learning and automatically adjust parameters to accurately and quickly recognize images [18]. In addition, these techniques outperform traditional computer vision techniques in terms of segmentation accuracy and adaptive capability. After the earliest end-to-end deep learning segmentation networks, PointNet [19] and PointNet++ [20], were proposed, a large number of deep-learning-based segmentation methods have emerged, having increasing segmentation effectiveness and accuracy. Majeed et al. developed a method for segmenting tree trunks and branches using the Kinect V2 sensor and deep-learning-based semantic segmentation techniques. The potential of deep-learning-based semantic segmentation was demonstrated in an orchard environment [21]. Chen et al. achieved the fast and accurate detection of color changes in melons based on YOLOv8 [22]. Han et al. used a deep-learning-based point cloud processing method for seedling segmentation and occluded leaf recovery [23].

In addition, to the best of our knowledge from reviews of the literature, models for optimizing horticultural seedling recognition and segmentation based on digital cousin are lacking. There are two major contributions in this paper. Firstly, we introduced the 3D Gaussian splatting technique for the reconstruction of horticultural seedlings, like watermelon seedlings, and obtained sufficient seedling phenotypic data by simulating morning, midday, and evening environmental and lighting conditions, as well as simulating different lighting effects in greenhouse cultivation environments. We ultimately sliced the three-bit watermelon seedling models from different angles and orientations. Secondly, to analyze the whole process of crop growth, an improved on YOLOv8 was proposed to identify and segment seedlings. To better adapt to the characteristics of the seedlings, we added the SPPELAN module to YOLOv8 by replacing the original detection head with the LADH detection head and improved the loss function to Focaler-ECIoU.

2. Materials and Methods

2.1. Overall Process from Creation of Mixed Real–Virtual Datasets to Semantic Segmentation

The overall workflow of our method can be described in three phases of real–virtual–real. In the first phase, we converted the real dataset into a 3D model in a virtual scene by means of monocular visual reconstruction, as shown in Figure 1a. In all deep learning training processes, using more datasets often means higher accuracy and generalization ability. So, to further supplement the dataset, in the second phase, we changed the environmental conditions of the 3D-reconstructed model by using different light sources to simulate the plant model under different environmental parameters. By slicing the new model, a set of sliced images of the model under different lighting conditions was generated, as shown in Figure 1b. The images obtained with this method can not only be used to supplement the dataset for YOLO training as well as improve the accuracy and generalization of the segmentation model but also provide a priori conditions for measuring the phenotypes of plants under different light conditions. Finally, we input all the images into the deep learning algorithm model to complete the training, as shown in Figure 1c. COLMAP is

a SOTA structure-from-motion (SfM) and MVS pipeline that converts multiple input 2D images into 3D point cloud models [24]. In this experiment, we chose the 3D Gaussian splatting method to construct the 3D model and used the improved YOLOv8 model for the semantic segmentation of images. In order to test the modeling quality of 3D Gaussian splatting under different lighting conditions, we did not use the photos formed by slicing after virtual lighting as the dataset during the actual experiment, but used the model directly on the photos collected of the seedlings in the real world as the dataset. The PSNR (peak signal-to-noise ratio) was used as a metric to test the construction of all models. Finally, we selected 450 images of seedlings under three light intensities in a 1:1:1 ratio as the YOLO dataset and input them into our improved YOLOv8 model to test the ability of the method.

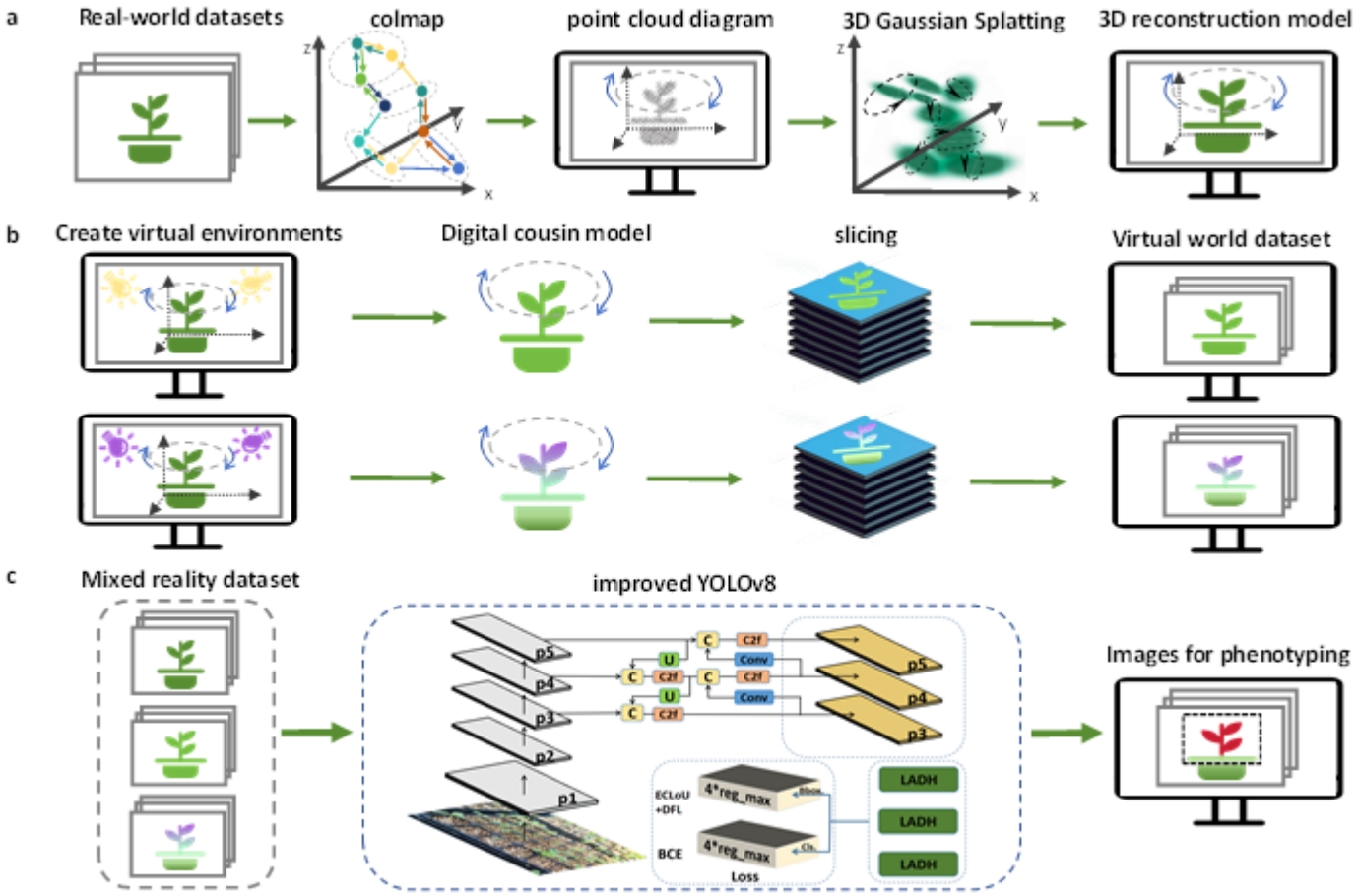


Figure 1. Overall flow chart. (**a**) Real datasets used to generate seedling digital twin models; (**b**) constructing virtual environments, modeling seedling digital cousins, and generating virtual datasets; (**c**) inputting mixed real–virtual datasets into the modified YOLOv8 model to produce phenotype detection images.

2.2. Data Preparation

2.2.1. Image Acquisition

The experimental data were collected at the Central China Sub-center of the National Vegetable Improvement Base (NVIB), where watermelon seedlings were photographed from 6 May 2024 to 13 May 2024 and from 15 May 2024 to 22 May 2024. Twelve groups of watermelon seedling beds with growth durations ranging from 2 to 10 days were collected. The watermelon seedling observation units were 70 cm long, 35 cm wide, and 3 cm high, and each observation unit contained a total of 50 watermelon seedling culture compartments, where single watermelon seedlings were placed in culture compartments

under typical soil cultivation methods. In the construction of the 3D Gaussian splatting dataset, in order to simulate the real working conditions of plant phenotyping and to ensure the reliability of the experimental data, all the images we collected were captured by the mobile device, Redmi K50, described in Table 1 via random surround shooting. For the construction of the YOLO dataset, we used different shooting techniques to capture images by controlling the distance (0.2–0.5 m), tilt angle (30–90°), and lighting conditions (strong light, night light, normal light). A total of 450 images were captured during the seedling emergence phase.

Table 1. Mobile device parameters.

Parameter	Value
Resolution	4624 × 3472
Flash bulb	none
Storage space	256 GB
Weight	210 g
Battery capacity	4700 mAh

2.2.2. Generation of Point Cloud Datasets for 3D Gaussian Splatting

(1) Retrieval Matching

After obtaining the dataset, COLMAP expresses each image in the dataset as $\mathcal{I} = \{I_i | i = 1 \ldots N_I\}$ and outputs $\mathcal{F}_i = \{(x_j, f_j), | j = 1 \ldots N\}$, where x_j denotes the position of the feature in the image, and f_j denotes the corresponding descriptor of the local features. Next, COLMAP starts to look for images with the same scene among all the images, outputs the possible matching image pairs, and expresses them as $\mathcal{C} = \{\{I_a, I_b\} | I_a, I_b \in \mathcal{I}, a < b\}$; the result of the feature matching is expressed as $\mathcal{M}_{ab} \in \mathcal{F}_a \times \mathcal{F}_b$. The matching image pairs obtained in the previous IoU step are then examined using geometric methods using the singular response matrix homograph H, which describes the variation between matched pairs of points in a plane. The essential matrix E and fundamental matrix F are used to describe the relationship between matched points using pairwise polar geometry. RANSAC robust estimation method is applied to reduce the effect of outliers. Finally, the validated photo pairs $\overline{\mathcal{C}}$, the passed matching results $\overline{\mathcal{M}}_{ab}$, and the correlation of the matching results G_{ab} are output.

(2) Incremental Reconstruction

In this process as shown in Figure 2, COLMAP first searches for a pair of images with the most geometrically consistent matches to obtain two initial keyframes for constructing a high-quality initialization. Then, it solves the PnP problem to obtain the interframe motion and triangulates the reconstructed points to extend the scene step by step. Finally, a nonlinear bundle adjustment method is used to optimize the camera position P_c and waypoint X_k to generate the final sparse point cloud. The optimized reprojection error is shown in Equation (1):

$$E = \sum_j \rho_j \left(\left| \left| \pi(P_c, X_k) - x_j \right| \right|_2^2 \right) \quad (1)$$

The objective function is minimized as shown in Equation (2):

$$J = \sum_{i,j} w_{ij} \parallel p_j - \pi(P_i, X_j) \parallel^2 \quad (2)$$

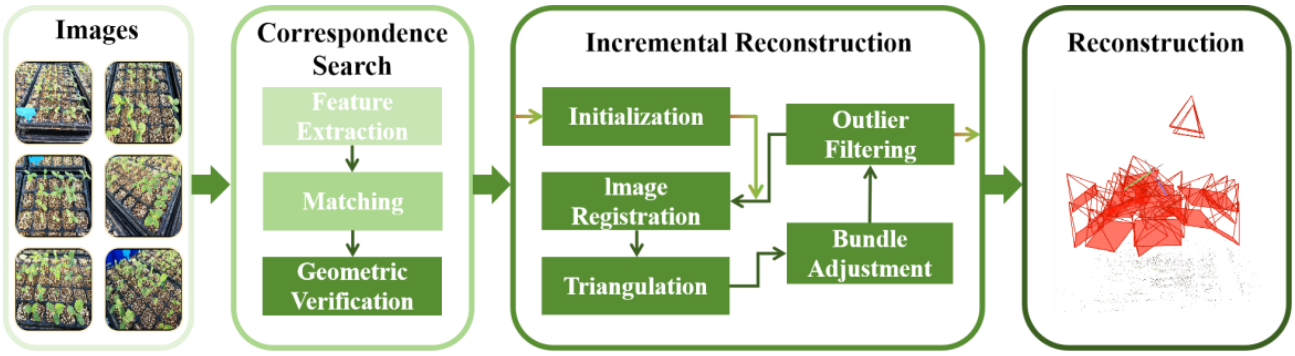


Figure 2. Converting seedling images into sparse 3D point clouds using COLMAP.

2.3. Building 3D Scenes with 3D Gaussian Splatting

Three-dimensional Gaussian splatting first initializes the SfM point cloud obtained from COLMAP to create 3D Gaussian spheres; then, using camera externals, the points are projected onto the image plane, followed by microscopic rasterization and rendering to generate the image. After obtaining the rendered image, it is compared with the ground truth image for loss and backpropagated along the blue arrow. In this case, the rendering optimization involves both under-reconstruction and reconstruction transitions. For under-reconstructed Gaussian points, the cloning of Gaussian points is achieved by creating copies of the same size and moving them in the direction of the position gradient; for reconstructed transition Gaussian points, the density control of the point cloud is accomplished by replacing the over-reconstructed Gaussian points with two new Gaussian values and initializing the position with the original Gaussian points. The up-pointing blue arrows indicate the update of the parameters in the 3D Gaussian, and down-pointing arrows indicate feeding into the adaptive density control to update the point cloud. Two trained 3D scenes are obtained after 7000 and 30,000 rounds of training. The overall flow is shown in Figure 3.

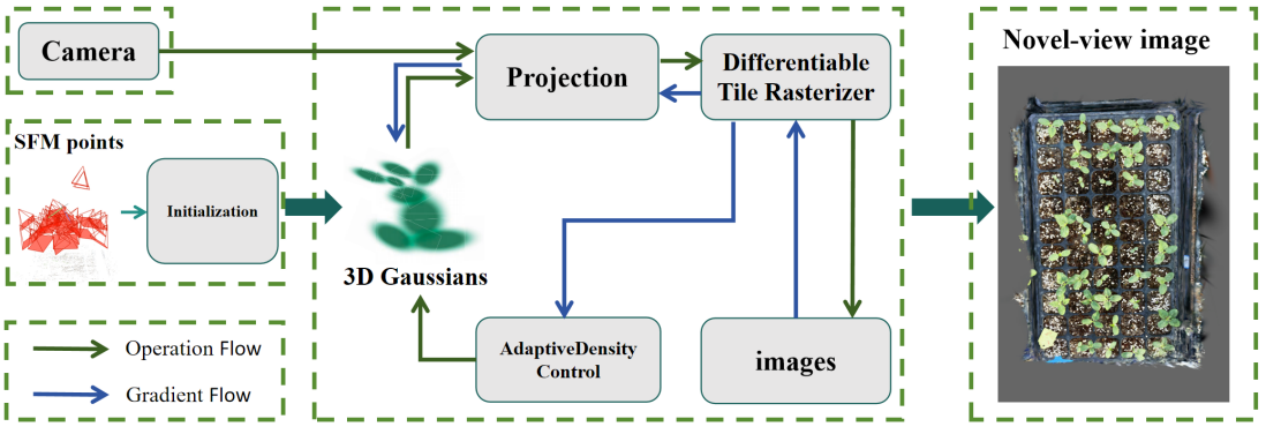


Figure 3. Rendering of sparse 3D point clouds generated by COLMAP using 3D Gaussian Splatting to obtain high-quality seedling models.

The 3D Gaussian mathematical expression is shown in Equation (3), and each 3D Gaussian point presents an ellipsoid shape, with the shape determined by the covariance matrix Σ. Its main parameters are the position, covariance matrix, opacity α, and spherical harmonic function. The position is the location and coordinates of the point, as well as the mean value of the 3D Gaussian; the covariance matrix determines the shape and orientation of the Gaussian point; the opacity α is used for real-time rendering in case of splatting; and the spherical harmonic function is used to fit the appearance related to the viewing angle.

Each of these parameters can be updated during iterative optimization and can be used to project onto the image using splatting very quickly conduct opacity fusion.

$$\begin{aligned} p(x) &= \frac{|A|}{(2\Pi)^{\frac{3}{2}}} \exp\left(-\frac{1}{2}(x-\mu)^T A^T A(x-\mu)\right) \\ &= \frac{|A|}{|\sum|^{\frac{1}{2}}(2\Pi)^{\frac{3}{2}}} \exp\left(-\frac{1}{2}(x-\mu)^T \sum^{t-1}(x-\mu)\right) \end{aligned} \tag{3}$$

The position of a given pixel x and the depth data of the Gaussian can be computed by looking at the transformation matrix W to form a sorted list of Gaussians $\mathcal{N}$. Then, alpha synthesis is used to calculate the final color of that pixel, as shown in Equation (4):

$$\mathbf{C} = \sum_{i \in \mathcal{N}} c_i a'_i \prod_{j=1}^{i-1} (1 - a'_j) \tag{4}$$

where c_i is the learned color, and the final opacity a'_i is the result of multiplication of the learned opacity $\mathbf{a_i}$ and Gaussian value, as shown in Equation (5):

$$\mathbf{a'_i} = \mathbf{a_i} * \exp\left(-\frac{1}{2}(\mathbf{x'} - \mathbf{u'_i})^{\mathbf{T}} \sum^{\mathbf{t-1}} (\mathbf{x'} - \mathbf{u'_i})\right) \tag{5}$$

where $\mathbf{x'}$ and $\mathbf{u'_i}$ are the coordinates of the three Gaussian points in the projection space. The rendering optimization is mainly adaptive density control for 3D GS with both under-reconstruction and reconstruction transitions. For the under-reconstructed Gaussian points, cloning of Gaussian points is achieved by creating copies of the same size and moving them in the direction of the position gradient; for the reconstructed transition Gaussian points, the over-reconstructed Gaussian points are replaced with two new Gaussian values, and the position is initialized by the original Gaussian points, thus completing the density control of the point cloud.

2.4. Implementation of YOLOv8 Improvements

2.4.1. Improving the Detection Model of YOLOv8

In the original YOLOv8 algorithm, for detecting small targets such as watermelon seedlings, the coupling head uses the same convolutional layers for classification and regression. This leads to different foci for classification and regression, which results in a decrease in the detection accuracy of the YOLOv8 model. Moreover, in order to continuously detect seedling growth, different segmentation models need to be used at different stages of seedling growth, which puts high demands on the speed of the model training process. In order to improve the training speed and reduce the computation amount, we introduced SPPELAN, which is a combination of SPP (spatial pyramid pooling) and ELAN, making full use of SPP's spatial pyramid pooling ability and ELAN's efficient feature aggregation ability. This algorithm reduces the number of channels, which reduces the amount of computation and increases the inference speed. This further improves the performance of the model. In addition, to improve the prediction frame adjustment and to speed up the frame regression rate, we introduce the efficient Focaler CIoU loss function. This method focuses on different regression samples through linear interval mapping, resulting in faster convergence and better localization for watermelon seedling detection. Overall, we improved the YOLOv8 detection model by introducing the LADH detection head, the SPPELAN deep learning network structure, and the efficient Focaler CIoU loss function.

2.4.2. LADH Detection Head

The original YOLOv8 detector uses a traditional symmetrical multistage compression structure, i.e., each detector head has the same compression ratio. However, in practice, there are differences in complexity between different object classes, e.g., the identification of watermelon seedlings is affected by the size and shape of the seedlings as well as by weather and light conditions. Therefore, we proposed an asymmetric multistage compression method, i.e., different compression ratios are used for different categories of detection heads. This can better accommodate different classes of feature representations and target size distributions as well as improve the generalization ability of the detector. To achieve asymmetric multilevel compression, we designed the LADH detector head, which is a lightweight structure consisting of depth-separable convolutions. LADH consists of two main components: the asymmetric head and the dual head. The asymmetric head is responsible for the asymmetric compression of different classes of features to accommodate the complexity of different classes. The dual head combines the outputs of the two asymmetric heads to produce the final object detection result. By using the LADH detection head, we can classify and recognize seedlings under different conditions, greatly improving the recognition performance and increasing the recognition accuracy. This is achieved by removing one Conv module and adding two layers of DWConv modules between the Conv module and conv2d module of the detector head branch of the original YOLOv8 network structure, after which the original structure is moved backward. The Conv module of the lower branch is replaced with the conv2d module in order to form an asymmetric head and a double head, as shown in Figure 4.

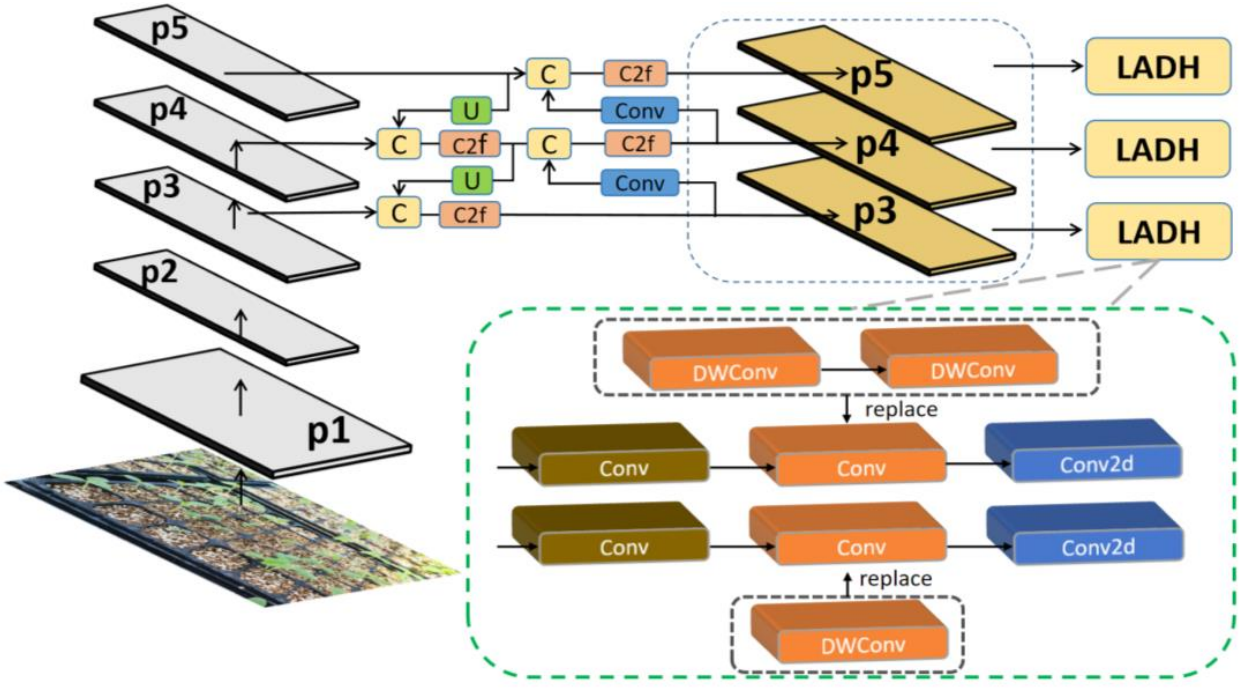


Figure 4. LADH structure in the improved YOLOv8 network structure.

The asymmetric head is responsible for the asymmetric compression of different classes of features, and the dual head combines the outputs of the two asymmetric heads and simultaneously compresses two depth-separated convolutional features along the channel dimensions to generate the final target detection results. The regression part uses convolution to extend the perceptual field and the number of parameters for the watermelon seedling detection task. At the same time, both features of the two depth-separable convolutions are compressed along the channel dimension. This approach

not only effectively alleviates the training difficulties associated with the watermelon seedling scoring task and improves the model performance but also significantly reduces the parameters and GFLOPs of the decoupling head module, which significantly improve the inference speed. Takin the watermelon seedling detection results as an example, we employed a breakthrough lightweight asymmetric detection head that reduces the computational complexity and significantly improves the inference speed of YOLOv8. This innovative utilization marks a significant milestone in the development of YOLOv8, resulting in improved efficiency and accelerated performance.

2.4.3. SPPELAN

SPP (spatial pyramid pooling) is a pooling strategy introduced in YOLOv3 [25], which is able to capture spatial information at different scales and enhance the robustness of the model. However, SPP is computationally intensive and is obviously inefficient in the task of watermelon seedling detection using a large data volume, which affects the inference speed of the model. To address this problem, we introduced an innovative technique called SPPELAN (spatial pyramid pooling enhanced with ELAN), which is a combination of SPP and ELAN, to enhance watermelon seedling detection by combining the strengths of both. By combining SPP with ELAN, we can make full use of the spatial pyramid pooling capability of SPP and the efficient feature aggregation capability of ELAN to further improve the performance of the model. We achieve this by replacing Conv, the first and sixth layer convolutional blocks of SPPF in the original YOLOv8, with the more efficient CBS, adjusting the number of channels per scale feature map and changing the number of channels in the pooling layer to reduce the number of channels from 1024 to 512, as shown in Figure 5.

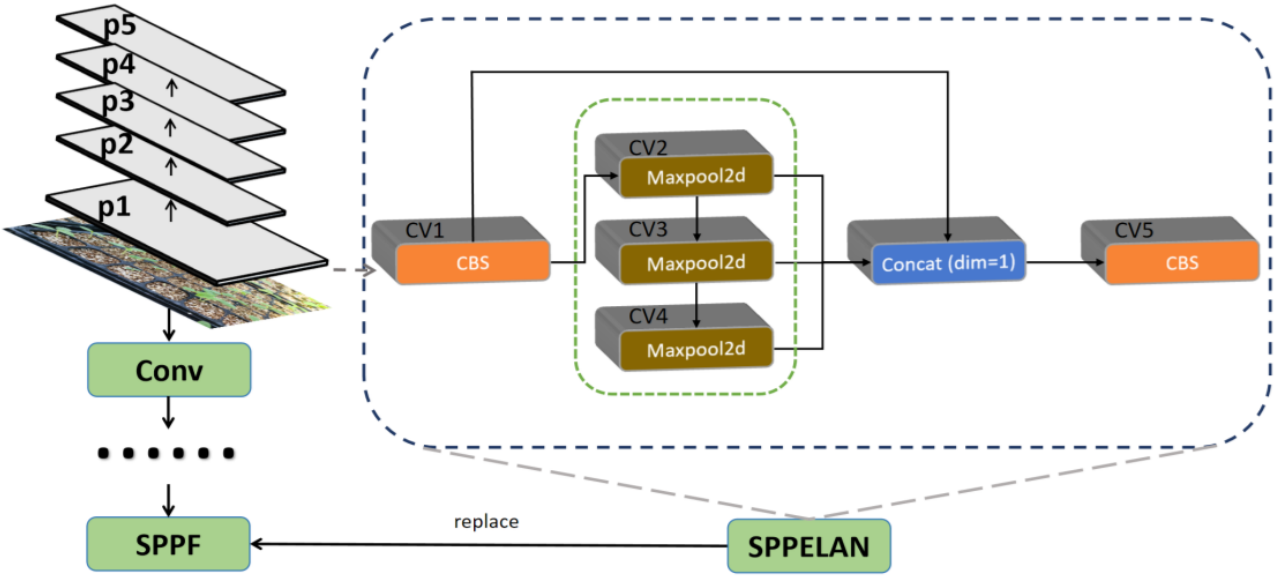


Figure 5. SPPELAN structure in the improved YOLOv8 network structure.

Compared with the previous version of IoU, we reduced the number of channels by optimizing the algorithm, which reduced the amount of computation and improved the inference speed. In addition, for the watermelon seedling dataset, which is complex, diverse, contains different sizes, and is difficult to identify, SPPELAN can reduce the number of parameters and improve the model performance and inference speed. The introduction of SPPELAN is expected to further improve the robustness and generalization of the YOLOv8 model while maintaining high accuracy. In addition, due to the lightweight

nature of ELAN, SPPELAN helps to reduce the computational effort of the model and increase the inference speed.

2.4.4. Focaler-ECIoU

Target detection can generally be divided into two parts, localization and detection, where the accuracy of localization is mainly dependent on the regression loss function. By choosing appropriate positive and negative samples, the intersection over union (IoU) is the most popular metric in bounding box regression, which is used to measure the similarity between predicted and real frames. To further obtain the optimal IoU metric, the IoU loss function is proposed to improve the IoU metric. However, the IoU loss function does not work when the predicted frames and the IoU loss function do not overlap. In order to solve these problems, many different evaluation systems based on the IoU have been derived, which address the defects in the original IoU loss function from different aspects and greatly enhance its robustness. The most representative methods are the generalized intersection-to-parallel (GIoU), distance-intersection-to-parallel (DIoU), and complete-intersection-to-parallel (CIoU) loss functions, which have played a fundamental role in the great progress of target detection, but there is still a lot of room for optimization. Among the above methods, CIoU is currently the best-performing boundary regression loss function [26], which takes into account three important geometric factors: overlap region, center-of-mass distance and aspect ratio, as shown in Equation (6):

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2\left(b^{gt}, b\right)}{c^2} + \alpha v \tag{6}$$

where $\rho^2\left(b^{gt}, b\right)$ denotes the square of the distance between the true and predicted frames, c^2 denotes the normalization factor, αv denotes the aspect ratio influence factor, $\mathcal{L}_{CIoU}$ denotes the improved central intersection and merging ratio loss, and IoU denotes the conventional intersection and merging ratio.

However, the width and height of the predicted frame cannot be simultaneously increased or decreased during the regression process. Therefore, once the model converges to the line-to-line ratio between the width and height of the predicted and actual frames, it sometimes prevents the model from optimizing the similarity effectively. So, to address the above problem, for watermelon seedling detection, we propose a new augmented loss function Focaler-ECIoU, which increases the adjustment of the predicted frames and speeds up the frame regression rate. The EIoU loss function [27] solves the CIoU problem by partitioning the aspect ratio influence factor αv according to CIoU and calculating the aspect ratios of the predicted frames and the actual frames, as shown in Equation (7):

$$ECIoU = 1 - IoU + \alpha v + \frac{\rho^2\left(b^{gt}, b\right)}{c^2} + \frac{\rho^2\left(h^{gt}, h\right)}{c_h^2} + \frac{\rho^2\left(w^{gt}, w\right)}{c_w^2} \tag{7}$$

where ECIoU denotes the improved intersection and concurrency ratio, IoU denotes the traditional intersection and concurrency ratio, αv denotes the aspect ratio influence factor, $\rho^2\left(b^{gt}, b\right)$ denotes the square of the distance between the real frame and the predicted frame, $\rho^2\left(h^{gt}, h\right)$ denotes the square of the distance between the true frame and the height, $\rho^2\left(w^{gt}, w\right)$ denotes the square of the distance between the true frame and the width, c^2 denotes the normalization factor, c_h^2 denotes the normalization factor related to the height, and c_w^2 denotes the normalization factor related to the width.

By the method of linear interval mapping, we can focus on different regression samples in different detection tasks to enhance the model's performance We use the linear interval

mapping method to reconstruct the IoU loss to improve the edge regression, as shown in Equation (8):

$$ECIoU^{focaler} = \begin{cases} 0, & ECIoU < d \\ \frac{ECIoU-d}{u-d}, & d \ll ECIoU \ll u \\ 1, & ECIoU > u \end{cases} \quad (8)$$

where $ECIoU^{focaler}$ denotes the improved intersection and merger ratio with focal adjustment, ECIoU denotes the improved intersection and merger ratio, d denotes the lower threshold, u denotes the upper threshold, and $\frac{ECIoU-d}{u-d}$ denotes the linear adjustment within the threshold, and is also denotes the ratio of the loss between d and u.

Here, $ECIoU^{focaler}$ denotes the reconstructed Focaler-ECIoU. By adjusting the values of d and u, $IoU^{focaler}$ focuses on different regression samples [28]. Its loss is defined as shown in Equation (9):

$$L_{Focaler-IoU} = 1 - IoU^{focaler} \quad (9)$$

where $L_{Focaler-IoU}$ denotes the loss of the focus-adjusted cross-parallel ratio, and $IoU^{focaler}$ denotes the focus-adjusted cross-parallel ratio.

3. Results

3.1. Experimental Setup

In this study, all of our images were captured with the same high-resolution mobile device, ensuring experimental rigor. The model training software environment was as follows: operating system, Ubuntu 20.04; deep learning framework, Pytorch1.10.0 using Python3.8 was used as the programming language to achieve model training. The hardware environment was as follows: graphics card, NVIDIA RTX4090 with 24 GB of video memory, CPU14-core Intel(R) Xeon(R) Gold 6330 CPU @ 2.00 GHz, and 80 GB of RAM. For 3D Gaussian splatting training, the COLMAP input image pixels were 4624 × 3472. The YOLO model compressed the image of 4624 × 3472 into an image input of 640 × 640 pixels in size. The number of samples in each batch was 32, starting from an initial learning rate of 0.01. We used an Adam optimization algorithm with a learning rate momentum of 0.937 and a weight decay coefficient of 0.0005. The model was trained for up to 500 cycles. Patience was set to 50, and the training process was stopped if no significant improvement was observed after 50 consecutive cycles. These hyperparameters were carefully chosen to encourage faster convergence, mitigate overfitting, and avoid the model falling into a local optimum.

For the 3D Gaussian splatting dataset, we classified the dataset into three types: strong light, D1; medium light, D2; and night-time violet light, D3, which simulated the various periods of light of the plant phenotypes by combining the proposed model with a physical world simulating engine, like Unity3D. Firstly, we tested the quality of the Gaussian splatting 3D scene reconstruction with fewer image inputs for testing the different lighting conditions, and we compared it with other currently dominant 3D reconstruction methods to understand their abilities to extract plant geometric models from these datasets. Second, we compared the improved algorithm with YOLOv8 itself and other mainstream algorithms from multiple perspectives to validate the superiority of our model. Ablation experiments were also conducted to demonstrate the contribution of each module.

3.2. 3D Models Generated with 3D Gaussian Splatting

3.2.1. RGB Imaging Datasets in Real World

All images collected in this study were taken at the Central China Branch of the National Vegetable Improvement Centre (NVIC) at Huazhong Agricultural University in Wuhan, Hubei Province, China. For 3D model reconstruction, we selected 12 groups of

watermelon seedlings with growth durations ranging from 2 to 10 days for focused image acquisition. As shown in Figure 6, The dataset was classified based on the lighting conditions during the actual growth of the seedlings in the greenhouse, which were categorized into three groups: strong light (D1), medium light (D2), and nighttime violet light (D3). D1 represents the growth scene of watermelon seedlings under strong illumination, such as mid-day on a sunny day, where brightness varies significantly due to shadow masking, posing a challenge for 3D reconstruction. D2 corresponds to the growing environment in early morning, dusk, or cloudy days, where light intensity is lower and brightness variation due to occlusion is minimal, making it an ideal environment for measuring seedling phenotypes. D3 represents the environment under violet light at night, where the luminance variations caused by occlusion are larger, and the violet light further complicates the 3D reconstruction process.

(**a**) (**b**) (**c**)

Figure 6. Examples from our dataset under different light conditions. (**a**) Representative image of the model under strong light conditions; (**b**) representative image of the model under medium light conditions; (**c**) representative image of the model under violet light conditions.

3.2.2. Example of 2D Imaging in Virtual World

This section shows 3D Gaussian splatting training the model 7000 and 30,000 times. In order to test the reconstruction effect of 3D Gaussian splatting with a smaller dataset, we set the number of image inputs to 30 for each set, and all the input imaging was captured with the mobile device mentioned in Section 2.2.1.

The peak signal-to-noise ratio (PSNR) is an engineering term that expresses the ratio of the maximum possible power of a signal to the power of the destructive noise that affects the accuracy of its representation, which has been widely recognized as an important metric for quantitatively assessing image quality. It is mainly used in the field of image compression and reconstruction, and its application has been further extended to emerging fields such as computer vision and neurographics [29]. Two $\boldsymbol{M} \times \boldsymbol{N}$ monochrome images I and K, where one is a noise approximation of the other, have their mean square error defined as shown in Equation (10):

$$\mathbf{MSE} = \frac{\mathbf{1}}{\boldsymbol{M} \times \boldsymbol{N}} \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{M}} \sum_{\boldsymbol{j}=\mathbf{1}}^{\boldsymbol{N}} [\boldsymbol{I}_{\text{reference}}(\boldsymbol{i},\boldsymbol{j}) - \boldsymbol{I}_{\text{examined}}(\boldsymbol{i},\boldsymbol{j})]^{\mathbf{2}} \tag{10}$$

The PSNR is defined as shown in Equation (11):

$$\mathbf{PSNR} = \mathbf{10} \times \mathbf{log}_{\mathbf{10}} \left(\frac{\boldsymbol{MAX}_{\boldsymbol{I}}^{\mathbf{2}}}{\mathbf{MSE}} \right) \tag{11}$$

where MAX_I denotes the maximum feasible pixel intensity of the image. Figures 7–9 show 3D reconstructions of images under the three different lighting conditions.

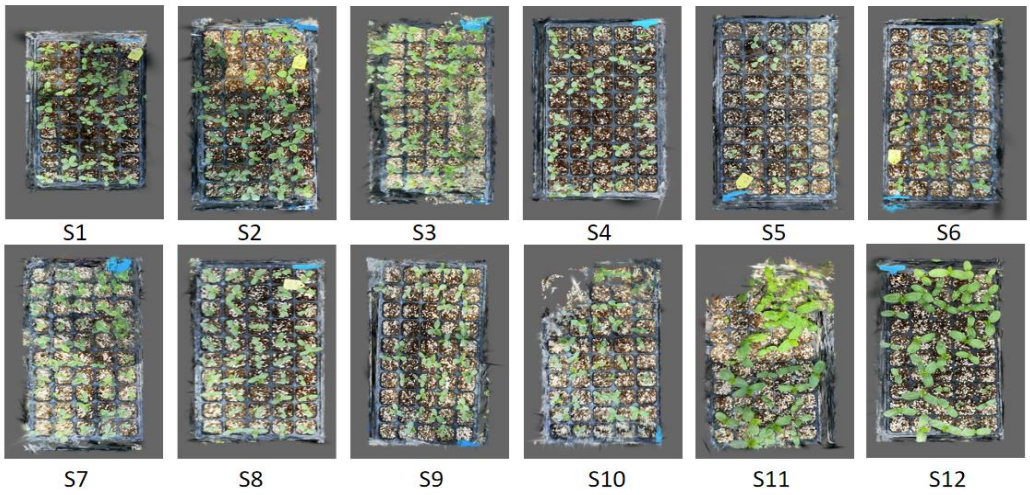


Figure 7. Novel images rendered from 3D Gaussian splatting—D1, strong light in different samples from S1 to S12.

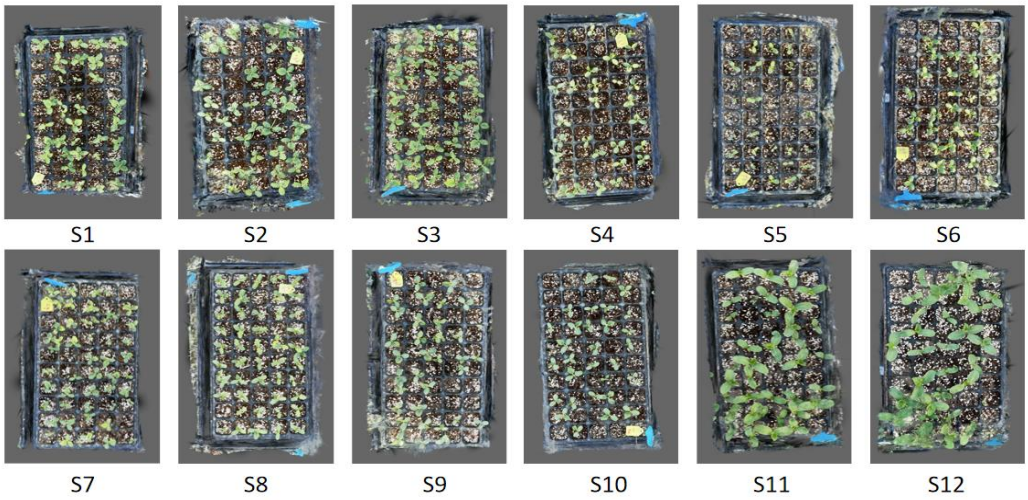


Figure 8. Novel images rendered from 3D Gaussian splatting—D2, medium light in different samples from S1 to S12.

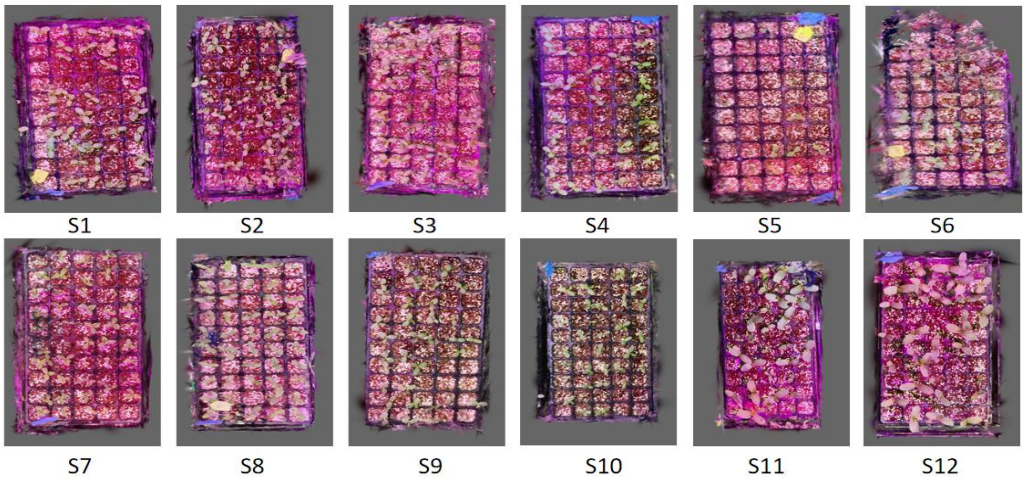


Figure 9. Novel images rendered from 3D Gaussian splatting—D3, violet light in different samples from S1 to S12.

3.2.3. Example of 3D Geometry Extraction in Virtual World

In this section, we detail the experimental results for the three datasets with a total of 36 objects. We list the parameters such as training time and PSNR value for each object reconstructed using 3D Gaussian splatting and compare them with the current mainstream training method, Instant NGP [30], as shown in Tables 2–4. The visual waterfall diagram is shown in Figure 10. In addition, the training times and rounds are key factors in the trade-off between the performance of 3D reconstruction and the size of models. We selected of 7000 and 30,000 as the feature parameters in the experiments and case studies in the early and original reconstruction model of 3D Gaussin splatting [16]. In order to visually compare the gap between 3D Gaussian splatting and InstantNGP in terms of modeling details, we also show plots comparing the models constructed with 3D Gaussian splatting with 7000 and 30,000 training iterations with the models constructed with Instant NGP in Figure 11.

Table 2. D1 group training results (strong light).

Dataset	**3D Gaussian Splatting (Strong Light)**				**InstantNGP (Strong Light)**	
	Time 7000 (min)	PSNR-7000	Time 30,000 (min)	PSNR-30,000	Time (min)	PSNR
S1	5.38	25.22	30.03	32.31	3.05	21.28
S2	5.87	26.02	27.65	32.09	3.13	21.65
S3	6.00	25.55	30.85	39.17	3.82	20.77
S4	5.45	24.64	31.52	33.67	3.18	20.25
S5	6.15	26.11	29.38	35.61	3.65	21.02
S6	5.77	30.34	29.85	39.01	3.45	21.84
S7	5.82	27.40	27.98	35.11	3.58	20.34
S8	6.03	27.49	29.62	36.37	3.98	20.67
S9	5.35	27.40	27.55	35.11	3.25	20.14
S10	5.67	16.51	22.87	22.31	3.33	13.27
S11	6.35	18.20	24.65	23.56	4.11	14.54
S12	5.82	24.63	28.45	31.68	3.87	19.51

Table 3. D2 group training results (medium light).

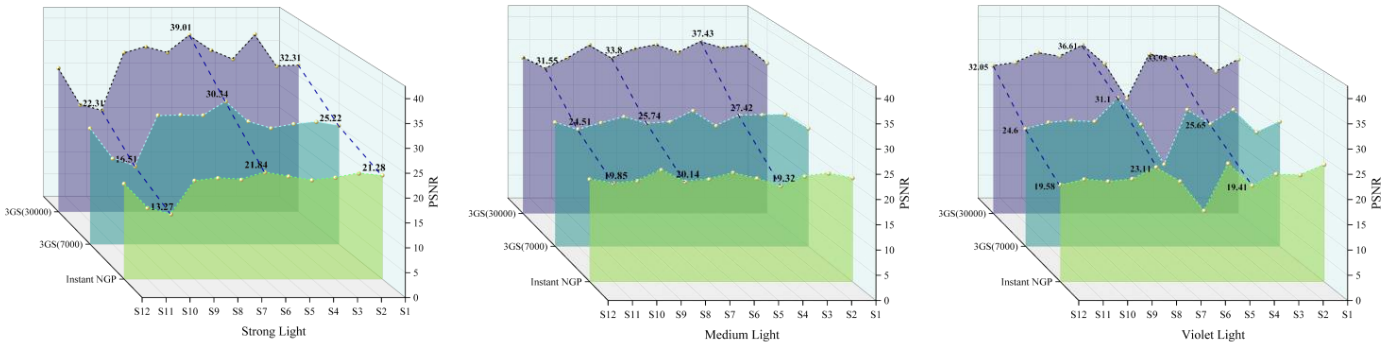
Dataset	**3D Gaussian Splatting (Medium Light)**				**InstantNGP (Medium Light)**	
	Time 7000 (min)	PSNR-7000	Time 30,000 (min)	PSNR-30,000	Time (min)	PSNR
S1	6.77	24.72	31.55	32.64	3.87	20.96
S2	7.02	27.65	33.25	36.57	4.30	21.88
S3	5.87	27.52	30.62	35.97	3.55	21.24
S4	6.55	27.42	32.45	37.43	3.67	19.32
S5	6.52	25.23	34.21	34.97	3.82	20.85
S6	6.32	28.37	33.30	36.65	3.58	21.98
S7	6.45	26.07	33.75	35.78	3.45	20.65

Table 3. *Cont.*

Dataset	3D Gaussian Splatting (Medium Light)				InstantNGP (Medium Light)	
S8	6.77	25.74	32.25	33.80	3.13	20.14
S9	5.98	27.12	29.82	36.64	3.85	22.54
S10	6.77	25.84	34.52	33.75	4.03	20.32
S11	6.21	24.51	30.85	31.55	4.48	19.85
S12	6.13	25.98	29.77	33.87	3.70	20.66

Table 4. D3 group training results (violet light).

Dataset	3D Gaussian Splatting (Violet Light)				InstantNGP (Violet Light)	
	Time 7000 (min)	PSNR-7000	Time 30,000 (min)	PSNR-30,000	Time (min)	PSNR
S1	6.18	26.16	29.85	33.43	4.14	23.66
S2	5.93	23.94	30.03	30.85	3.35	21.54
S3	6.10	28.60	23.58	34.47	3.77	21.72
S4	6.21	25.65	32.65	33.95	3.68	19.41
S5	6.33	28.61	23.68	34.62	4.32	23.85
S6	6.17	17.15	23.45	24.90	3.70	14.32
S7	5.85	25.53	28.70	32.55	3.82	20.25
S8	6.11	31.10	23.48	36.61	3.97	23.11
S9	5.87	26.16	28.63	34.14	3.35	20.74
S10	5.97	26.34	23.77	34.93	3.55	20.25
S11	6.30	25.96	26.97	32.76	3.82	20.66
S12	6.18	24.60	27.80	32.05	3.80	19.58



(**a**) (**b**) (**c**)

Figure 10. Visualized waterfall plots of 3D Gaussian splatting and InstantNGP in reconstructing model parameters in three lighting scenarios. (**a**) Comparison of InstantNGP, 3D Gaussian splatting trained 7000 times and 30,000 times under strong light; (**b**) comparison of InstantNGP, 3D Gaussian splatting trained 7000 times and 30,000 times under medium light; (**c**) comparison of Instant NGP, 3D Gaussian splatting trained 7000 times and 30,000 times under violet light.

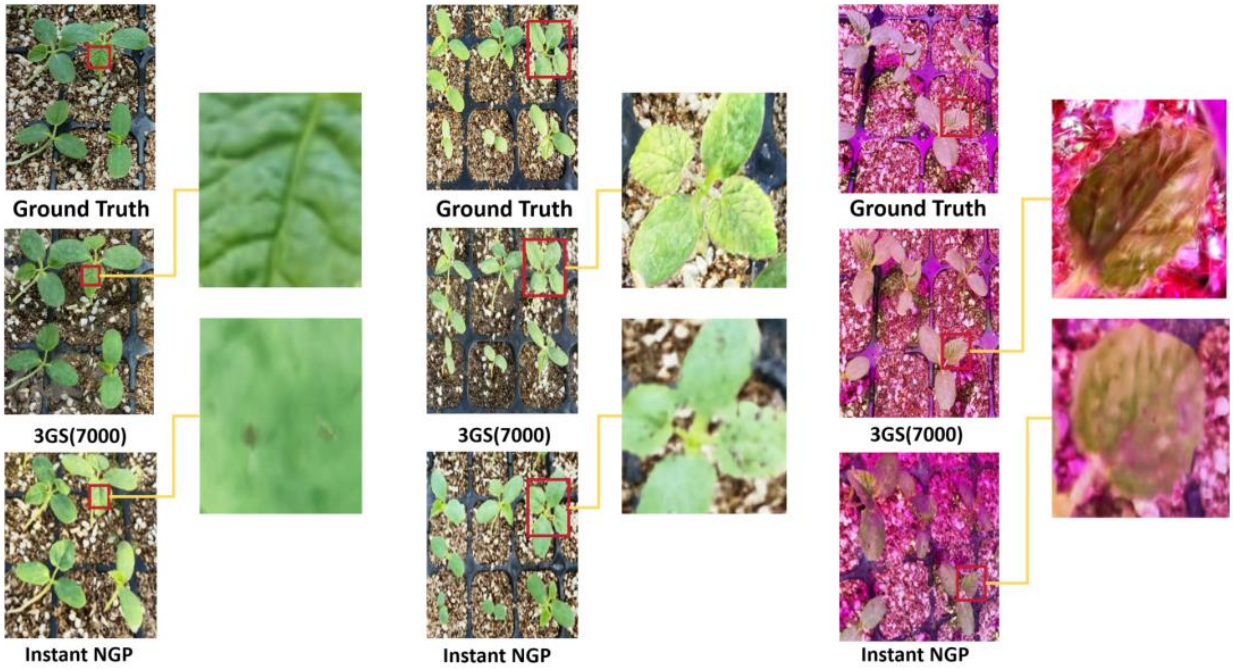


Figure 11. Detailed comparison of 3D Gaussian splatting (7000 trainings) and InstantNGP on the model surface.

3.3. Performance of Improved YOLOv8

3.3.1. Comparative Analysis of the Performance of Different Models

In order to verify the performance of the improved YOLOv8 model, we conducted comparison experiments with several image segmentation models. The results of the comparison experiments are shown in Table 5.

Table 5. Comparison with other methods.

Model	BOX-MAP50	MASK-MAP50	Loss	Epochs
YOLOv5n-SEG	0.895	0.909	0.855	201
YOLOv8n-SEG	0.907	0.909	0.765	370
YOLOv11n-SEG	0.901	0.909	0.807	180
OURS	0.910	0.913	0.616	254

As can be seen from Table 6, although the number of convergence iterations (epochs) of the improved YOLOv8 model increased compared to YOLOv5n-SEG and YOLOv11n-SEG (254 vs. 201/180), the average detection accuracy and average segmentation accuracy on key metrics improved for all models (0.910 vs. 0.895/0.907/0.901, 0.913 vs. 0.909/0.909/0.909). Moreover, the training loss (loss) is lower than that of YOLOv5n-SEG, YOLOv8n-SEG and YOLOv11n-SEG (0.616 vs. 0.855,0.616 vs. 0.765, 0.616 vs. 0.807), and the improved YOLOv8 model showed excellent performance on the specific watermelon seedling detection task. The loss of the proposed method increased by 0.24. This reflects the expertise and adaptability of the improved YOLOv8 model in specific scenarios. The data in the table overall reflect the improved performance of the YOLOv8 model.

Table 6. Ablation experiments when LADH, Focaler-ECIoU, and SPPELAN are applied to baseline.

Method	LADH	Focaler-ECIoU	SPPLAN	mAP50 (B)	mAP50 (M)	Epoch	Loss	GFLOPS	F1 Score
Baseline	×	×	×	0.907	0.909	370	0.765	12.0	0.855
LADH	√	×	×	0.906	0.908	201	0.854	11.6	0.855
LADH + Focaler-ECIoU	√	√	×	0.907	0.913	181	0.667	11.6	0.854
LADH + Focaler-ECIoU + SPPELAN	√	√	√	0.910	0.913	254	0.616	11.1	0.856

3.3.2. Improved YOLOv8 Detection Model Ablation Test

In order to verify the performance of the improved YOLOv8 model, we conducted comparison experiments with several image segmentation models. The results of the comparison experiments are shown in Table 6.

The introduction of the LADH detection head reduced the training duration and parameter count by 52.81% and 3.33%, respectively, but the mAP50 (B) and mAP50 (M) both decreased by 0.1%. This decrease could be attributed to LADH, which improved the convergence speed and reduced the parameter count but at the expense of some detail, resulting in slight decreases in mAP50 (B) and mAP50 (M). By retaining LADH and adding Focaler-ECIoU, mAP50 (B) and mAP50 (M) improved to 90.7% and 91.3%, respectively. In addition, we observed improvements in epochs (181), F1 score (0.856), and loss (0.667). And, the performance of improved model increased 0.001% compared with the baseline. The addition of SPPELAN further enhanced mAP50 (B) to 91.0%. Thanks to its efficient feature aggregation capability, the loss (0.616) and GFLOPS (11.1) also reduced, improving the lightness of the model. From the above experiments, we found that compared with the baseline, the improved YOLOv8 model shows significant improvements in average detection accuracy, average segmentation accuracy, number of converged iterations (epochs), loss, and GFLOPS. This suggests that the combination of the LADH, SPPELAN, F1 score, and Focaler-ECIoU modules has a positive impact on the performance of the model. In conclusion, the proposed scheme is feasible.

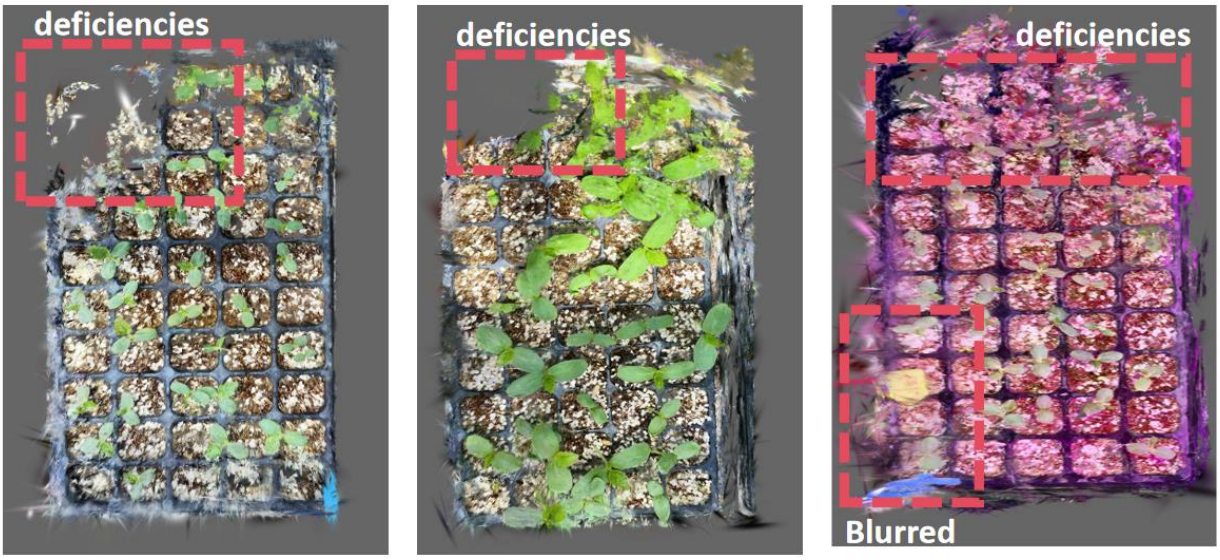
4. Discussion

4.1. RGB Imaging Dataset from Real and Virtual Worlds

In the current field of phenotype detection, the problems of insufficient datasets and the long training time of semantic segmentation models have somewhat limited the effect of phenotype detection in the whole stage of seedling growth. In this paper, we addressed the difficult problem of data acquisition through obtaining three types of mixed real and virtual datasets: real, virtual and evolutionary. Meanwhile, a more efficient seedling semantic segmentation model was constructed, which promoted the adaptability of the model to diverse plant morphologies and improved the performance of the model in practical applications. During the construction of the hybrid virtual–real dataset, we used the 3D Gaussian splatting technique as a model for constructing a digital twin scene, and we changed the virtual environment based on it to construct a digital cousin model. One of its key features is that it balances speed and quality, improving the quality of the model while ensuring faster convergence. During the construction of our 3D seedling models, after 7000 iterations of training, all models converged within 7 min. At this point, the model quality exceeded that of InstantNGP, with a PSNR of over 24, which met the requirements for segmenting watermelon seedlings from different angles and performing phenotypic

measurements in 3D scenes. After 30,000 iterations, the quality was further improved, with a PSNR above 30. When high-precision measurements of certain data are required, the model with 30,000 iterations can be used instead of the model with 7000 iterations to improve the accuracy of the measurements. This approach not only accelerates the training process but also significantly improves the accuracy of the final model, allowing for more efficient and accurate measurements in plant phenotyping research.

However, since the seedling images were collected randomly, the geometric features of the seedlings may not be fully extracted during the image acquisition process due to the incompleteness of the viewpoints captured, resulting in a local point cloud with missing data, which ultimately leads to a paralyzed model and a drastically reduced PSNR. As shown in Figure 12, S10 and S11 in group D1 and S6 in group D3 exhibit slight disabling and local blurring, which may limit the application of 3D Gaussian splatting in the 3D reconstruction process of seedlings to some extent. In the future, we can focus on constructing exclusive shooting schemes and related equipment for watermelon seedlings to completely and efficiently finish the seedling model while requiring less image acquisition. We will extend our research to seedlings of oilseed rape, chili, and other plants, as well as establish a whole set of databases of different seedlings under different light conditions with different modeling methods so as to provide more possibilities for subsequent research on plant phenology.



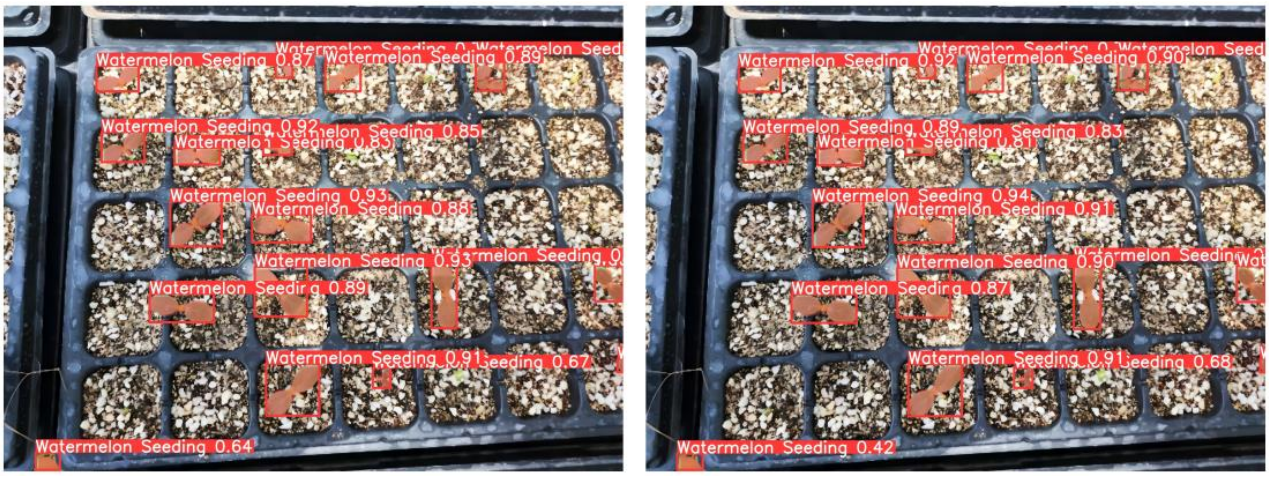
(**a**) (**b**) (**c**)

Figure 12. Defects in 3D modeling. (**a**) Missing models due to randomness of image capture in medium lighting; (**b**) missing areas due to randomness of image capture under strong lighting; (**c**) missing areas due to randomness of image capture under violet lighting.

4.2. Performance Analysis and Comparison of Improved YOLOv8 Semantic Segmentation Models

In this study, we aimed to achieve the semantic segmentation of watermelon seedlings. To this end, we integrated three modules, LADH, SPPELAN, and Focaler-ECIoU, and evaluated their impact on model performance. Our comparison and ablation experiments indicated that our model outperformed the classical segmentation algorithms on all metrics, achieving a 91.0% mAP50 (B) and a 91.3%mAP50 (M). In Figure 13, under challenging lighting conditions and increased background complexity, the LADH module utilizes dynamic detection heads to enhance contextual understanding. This enables the model to provide a deeper understanding of a wider range of perspectives and complex shapes. As a result, our model stands out from the original YOLOv8 model under difficult lighting and

background conditions, validating its robustness and efficiency for real-world applications. However, our study still has some limitations. High-quality datasets are very important in deep learning, which may directly or indirectly affect the segmentation results. Although we expanded the dataset by using multiangle slices of 3D reconstruction models, we spent considerable time and money in the process of producing high-quality seedling datasets. Obtain seedling datasets quickly and efficiently is still a difficult task. In our future work, we will continue to explore the possibility of reconstructing 3D scenes in the production of datasets and compare the performance differences between the datasets generated from virtual scenes and those produced from real scenes, so as to provide more accurate data for the phenotype detection of seedlings. We will also conduct identification and yield prediction studies on watermelon seedlings at different fertility stages, and we plan to further explore the collaborative control of the watermelon seedling growth environment and facilities, including robotic picking, harvesting, and logistic resource scheduling, to satisfy the needs for quantitative identification statistics, real-time growth assessment, and yield prediction in smart watermelon production.



(**a**) (**b**)

Figure 13. Comparison of actual detection results. (**a**) Test results for YOLOv8; (**b**) test results for our method.

5. Conclusions

In this study, we introduced 3D Gaussian splatting technology as a new method for phenotype information extraction and preservation, which breaks through the spatial and temporal limitations of traditional phenological measurement methods. Moreover, for the recognition and segmentation of watermelon seedlings, we improved YOLOv8 by adding three innovative modules, namely, LADH, SPPELAN, and Focaler-ECIoU, to enhance the efficiency and performance of seedling recognition. When we identify watermelon seedlings in complex environments, different compression ratios can be adopted for LADH for different categories of detector heads. This can better adapt to the feature representation and target size distribution of the different categories of watermelon seedlings as well as improve the generalization ability of the detector. SPPELAN can extract watermelon seedlings in different weather conditions, light intensities, and from complex backgrounds separately to improve the accuracy of model recognition, greatly reducing the detection time and improving the detection efficiency. Focaler-ECIoU can increase the prediction frame adjustment and accelerate the frame regression rate during watermelon seedling detection to speed up the model inference. Compared with YOLOv8, the performance of the proposed detecting model is increased: mAP50 (B) and mAP50 (M) are improved to

90.7% and 91.3%, respectively. Once we reconstruct a particular seedling in 3D, we can extract the phenotype of that plant at the moment of reconstruction at any time, and the high-fidelity quality of the reconstruction greatly improves the accuracy of phenotypic measurements. Our three datasets captured under different light conditions correspond to seedling growth scenarios under three different light intensities, emphasizing the different situations of the actual production process. By analyzing and comparing the reconstruction models obtained from sparse random shots of watermelon seedlings, we found that the 3D models obtained with this method can basically meet the requirements of phenotype extraction, showing that the model has great potential for watermelon seedling phenology measurement. However, this work is an early exploration of object detection using a digital cousin, and the performance improvement of YOLOv8 is minor compared with that of other YOLO series, as evidenced by the F1 score only increasing 0.001%. Plus, the dynamics and evolution basically show the benefit of digital cousins, which can be studied in more complexity and temporospatial scenarios in future research.

Author Contributions: Conceptualization, Y.S. and L.Y.; data curation, Y.S. and L.Y.; investigation, S.L., X.Y., C.M. and Y.H.; methodology, Y.S. and L.Y.; resources, S.L., C.M., Y.H. and A.H.; visualization, Y.S. and S.L.; writing—original draft, Y.S. and S.L.; writing—review and editing, L.Y., X.Y. and A.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Fundamental Research Funds for the Central Universities (BC2024211), the Key R&D Program of Hubei Province (2024EIA009), and the Key R&D Program of Shannan China (SNSBJKJJHXM2023004).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request.

Acknowledgments: The authors would like to thank Zhilong Bie and the laboratory; we appreciate the reviewers who provided helpful suggestions for this manuscript.

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

References

1. Saranya, T.; Deisy, C.; Sridevi, S.; Anbananthen, K.S.M. A comparative study of deep learning and Internet of Things for precision agriculture. *Eng. Appl. Artif. Intell.* **2023**, *122*, 106034. [CrossRef]
2. Kierdorf, J.; Junker-Frohn, L.V.; Delaney, M.; Olave, M.D.; Burkart, A.; Jaenicke, H.; Muller, O.; Rascher, U.; Roscher, R. GrowliFlower: An image time-series dataset for GROWth analysis of cauLIFLOWER. *J. Field Robot.* **2023**, *40*, 173–192. [CrossRef]
3. Fan, J.; Zhang, Y.; Wen, W.; Gu, S.; Lu, X.; Guo, X. The future of Internet of Things in agriculture: Plant high-throughput phenotypic platform. *J. Clean. Prod.* **2021**, *280*, 123651. [CrossRef]
4. Strock, C.F.; Schneider, H.M.; Lynch, J.P. Anatomics: High-throughput phenotyping of plant anatomy. *Trends Plant Sci.* **2022**, *27*, 520–523. [CrossRef]
5. Xu, S.; Zhang, Y.; Dong, W.; Bie, Z.; Peng, C.; Huang, Y. Early identification and localization algorithm for weak seedlings based on phenotype detection and machine learning. *Agriculture* **2023**, *13*, 212. [CrossRef]
6. Li, L.; Bie, Z.; Zhang, Y.; Huang, Y.; Peng, C.; Han, B.; Xu, S. Nondestructive Detection of Key Phenotypes for the Canopy of the Watermelon Plug Seedlings Based on Deep Learning. *Hortic. Plant J.* 2023; *in press*. [CrossRef]
7. Dallel, M.; Havard, V.; Dupuis, Y.; Baudry, D. Digital twin of an industrial workstation: A novel method of an auto-labeled data generator using virtual reality for human action recognition in the context of human–robot collaboration. *Eng. Appl. Artif. Intell.* **2023**, *118*, 105655. [CrossRef]
8. Dai, T.; Wong, J.; Jiang, Y.; Wang, C.; Gokmen, C.; Zhang, R.; Wu, J.; Fei-Fei, L. Acdc: Automated creation of digital cousins for robust policy learning. *arXiv* **2024**, arXiv:2410.07408.
9. Samavati, T.; Soryani, M. Deep learning-based 3D reconstruction: A survey. *Artif. Intell. Rev.* **2023**, *56*, 9175–9219. [CrossRef]
10. Kim, G.; Kim, Y.; Yun, J.; Moon, S.W.; Kim, S.; Kim, J.; Park, J.; Badloe, T.; Kim, I.; Rho, J. Metasurface-driven full-space structured light for three-dimensional imaging. *Nat. Commun.* **2022**, *13*, 5920. [CrossRef]
11. Lee, Y.J.; Yoo, S.K. Design of ToF-Stereo Fusion Sensor System for 3D Spatial Scanning. *Smart Media J.* **2023**, *12*, 134–141. [CrossRef]

12. Lin, H.; Zhang, H.; Li, Y.; Huo, J.; Deng, H.; Zhang, H. Method of 3D reconstruction of underwater concrete by laser line scanning. *Opt. Lasers Eng.* **2024**, *183*, 108468. [CrossRef]
13. Li, H.; Wang, S.; Bai, Z.; Wang, H.; Li, S.; Wen, S. Research on 3D reconstruction of binocular vision based on thermal infrared. *Sensors* **2023**, *23*, 7372. [CrossRef]
14. Yu, Z.; Peng, S.; Niemeyer, M.; Sattler, T.; Geiger, A. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 25018–25032.
15. Pan, S.; Wei, H. A global generalized maximum coverage-based solution to the non-model-based view planning problem for object reconstruction. *Comput. Vis. Image Underst.* **2023**, *226*, 103585. [CrossRef]
16. Kerbl, B.; Kopanas, G.; Leimkühler, T.; Drettakis, G. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Trans. Graph.* **2023**, *42*, 139:1–139:14. [CrossRef]
17. Sodjinou, S.G.; Mohammadi, V.; Mahama AT, S.; Gouton, P. A deep semantic segmentation-based algorithm to segment crops and weeds in agronomic color images. *Inf. Process. Agric.* **2022**, *9*, 355–364. [CrossRef]
18. Sharifani, K.; Amini, M. Machine learning and deep learning: A review of methods and applications. *World Inf. Technol. Eng. J.* **2023**, *10*, 3897–3904.
19. Haznedar, B.; Bayraktar, R.; Ozturk, A.E.; Arayici, Y. Implementing PointNet for point cloud segmentation in the heritage context. *Herit. Sci.* **2023**, *11*, 2. [CrossRef]
20. Luo, J.; Zhang, D.; Luo, L.; Yi, T. PointResNet: A grape bunches point cloud semantic segmentation model based on feature enhancement and improved PointNet++. *Comput. Electron. Agric.* **2024**, *224*, 109132. [CrossRef]
21. La, Y.J.; Seo, D.; Kang, J.; Kim, M.; Yoo, T.W.; Oh, I.S. Deep Learning-Based Segmentation of Intertwined Fruit Trees for Agricultural Tasks. *Agriculture* **2023**, *13*, 2097. [CrossRef]
22. Chen, G.; Hou, Y.; Cui, T.; Li, H.; Shangguan, F.; Cao, L. YOLOv8-CML: A lightweight target detection method for Color-changing melon ripening in intelligent agriculture. *Sci. Rep.* **2024**, *14*, 14400. [CrossRef] [PubMed]
23. Han, B.; Li, Y.; Bie, Z.; Peng, C.; Huang, Y.; Xu, S. MIX-NET: Deep learning-based point cloud processing method for segmentation and occlusion leaf restoration of seedlings. *Plants* **2022**, *11*, 3342. [CrossRef] [PubMed]
24. Liu, Z.; Liu, X.; Guan, H.; Yin, J.; Duan, F.; Zhang, S.; Qv, W. A depth map fusion algorithm with improved efficiency considering pixel region prediction. *ISPRS J. Photogramm. Remote Sens.* **2023**, *202*, 356–368. [CrossRef]
25. Dewi, C.; Chen, R.C.; Yu, H.; Jiang, X. Robust detection method for improving small traffic sign recognition based on spatial pyramid pooling. *J. Ambient Intell. Humaniz. Comput.* **2023**, *14*, 8135–8152. [CrossRef]
26. Cai, H.; Lan, L.; Zhang, J.; Zhang, X.; Zhan, Y.; Luo, Z. IoUformer: Pseudo-IoU prediction with transformer for visual tracking. *Neural Netw.* **2024**, *170*, 548–563. [CrossRef]
27. Dong, C.; Duoqian, M. Control distance IoU and control distance IoU loss for better bounding box regression. *Pattern Recognit.* **2023**, *137*, 109256. [CrossRef]
28. Zhang, H.; Zhang, S. Focaler-IoU: More Focused Intersection over Union Loss. *arXiv* **2024**, arXiv:2401.10525.
29. Venu, D.N. PSNR based evalution of spatial Guassian Kernals For FCM algorithm with mean and median filtering based denoising for MRI segmentation. *IJFANS Int. J. Food Nutr. Sci.* **2023**, *12*, 928–939.
30. Lv, J.; Jiang, G.; Ding, W.; Zhao, Z. Fast Digital Orthophoto Generation: A Comparative Study of Explicit and Implicit Methods. *Remote Sens.* **2024**, *16*, 786. [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.