


Article

YOLOv8-Pearpollen: Method for the Lightweight Identification of Pollen Germination Vigor in Pear Trees

Weili Sun ^{1,†}, Cairong Chen ^{2,†}, Tengfei Liu ¹, Haoyu Jiang ¹, Luxu Tian ², Xiuqing Fu ² , Mingxu Niu ³, Shihao Huang ² and Fei Hu ^{1,*}

- ¹ College of Artificial Intelligence, Nanjing Agricultural University, Nanjing 210031, China; 2023819031@stu.njau.edu.cn (W.S.); 2023119012@stu.njau.edu.cn (T.L.); 2022819036@stu.njau.edu.cn (H.J.)
² College of Engineering, Nanjing Agricultural University, Nanjing 210031, China; crchen@njau.edu.cn (C.C.); 2022812055@stu.njau.edu.cn (L.T.); fuxiuqing@njau.edu.cn (X.F.); 9223010725@stu.njau.edu.cn (S.H.)
³ College of Horticulture, Nanjing Agricultural University, Nanjing 210031, China; mingxu@njau.edu.cn
* Correspondence: hufei@njau.edu.cn
† These authors contributed equally to this work.

Abstract: Pear trees must be artificially pollinated to ensure yield, and the efficiency of pollination and the quality of pollen germination affect the size, shape, taste, and nutritional value of the fruit. Detecting the pollen germination vigor of pear trees is important to improve the efficiency of artificial pollination and consequently the fruiting rate of pear trees. To overcome the limitations of traditional manual detection methods, such as low efficiency and accuracy and high cost, and to meet the requirements of screening high-quality pollen to promote the yield and production of fruit trees, we proposed a detection method for pear pollen germination vigor named YOLOv8-Pearpollen, an improved version of YOLOv8-n. A pear pollen germination dataset was constructed, and the image was enhanced using Blend Alpha to improve the robustness of the data. A combination of knowledge distillation and model pruning was used to reduce the complexity of the model and the difficulty of deployment in hardware facilities while ensuring that the model achieved or approached the detection effect of a large-volume model that can adapt to the actual requirements of agricultural production. Various ablation tests on knowledge distillation and model pruning were conducted to obtain a high-quality lightweighting method suitable for this model. Test results showed that the mAP of YOLOv8-Pearpollen reached 96.7%. The Params, FLOPs, and weights were only 1.5 M, 4.0 G, and 3.1 MB, respectively, and the detection speed was 147.1 FPS. A high degree of lightweighting and superior detection accuracy were simultaneously achieved.

Keywords: pear pollen; sprouting vigor; lightweight; object detection; YOLOv8



Citation: Sun, W.; Chen, C.; Liu, T.; Jiang, H.; Tian, L.; Fu, X.; Niu, M.; Huang, S.; Hu, F. YOLOv8-Pearpollen: Method for the Lightweight Identification of Pollen Germination Vigor in Pear Trees. *Agriculture* **2024**, *14*, 1348. <https://doi.org/10.3390/agriculture14081348>

Academic Editor: Takashi Okayasu

Received: 25 June 2024

Revised: 3 August 2024

Accepted: 7 August 2024

Published: 12 August 2024



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

Pollen is the male reproductive cell of seed plants and is an important agent of sexual reproduction, which is closely related to the improvement of plant varieties and the selection of high-quality genotypes. Good-quality male pollen can greatly improve pollination success, and pollen germination vigor is one of the indicators of male pollen quality [1]. Therefore, measuring pollen germination vigor is crucial for plant science and agricultural production to achieve a high and excellent yield of plants and facilitate low-cost and efficient pollination. Most existing pollen viability assays employ staining and in vitro culture methods [2]. However, staining cannot accurately indicate pollen viability and is not suitable for studying the effect of a treatment on pollen viability. Meanwhile, in vitro culture methods require cumbersome experimental manipulations and high economic and labor costs. Therefore, developing an efficient and simple method for the determination of pollen viability is important [3].

Pear is a native fruit tree of China. Its original species can be traced back to the tertiary period (66 million years ago to 2.6 million years ago) or earlier in the mountainous areas

of western or southwestern China, and it has a long history of more than 3000 years of cultivation in China. Pears are popular because of their sweet flavor and high nutritional value. China's pear production reaches up to 19 million tons (2023), accounting for 3/4 of the world's total pear production [4]. Given that pear is a typical self-fertilizing nonaffinity fruit tree, only the same variety of pear trees can blossom and they cannot pollinate each other. Therefore, the production must be reasonably configured using pollinating trees or artificial pollination to obtain a normal yield. In addition, the pear flowering period is only a short 4–5 days. Inappropriate pollination tree configuration, such as poor flowering, could easily cause yield reduction. Therefore, pears need efficient pollination methods and male pollen with excellent germination vigor. In different varieties of pear pollen there are small differences in the rate of germination potential, but the consistent trend shows that while in pollen tube germination length in the early stage of germination there are individual differences, the germination characteristics are obvious; both pollen grains and pollen tubes are consistent in shape, and easy to observe during study.

Machine learning has been increasingly utilized in the field of palynology to classify and predict pollen concentrations. Punyasena et al., developed a layered machine learning classification system that discriminates variations in pollen shape, size, and texture, demonstrating the capability of machine learning systems to solve challenging palynological classification problems [5]. By combining pollen observations with meteorological and land surface variables, Liu et al., used machine learning to estimate atmospheric ambrosia pollen concentrations in Tulsa, OK [6]. Sobol et al., investigated the use of supervised machine learning for biome classification using pollen datasets, assigning modern pollen samples to biome classes [7]. In addition, Sobol et al., applied machine learning to reconstruct past biome states using modern pollen assemblages in Southern Africa [8]. Huete et al., utilized machine learning approaches to forecast grass pollen evolution by analyzing satellite-based landscape information and phenology [1]. Zewdie et al., presented a method for estimating airborne pollen concentrations using deep neural networks and ensemble machine learning methods, testing the performance of machine learning models on a dataset from 2012 to 2017 [9]. Furthermore, Zewdie et al., applied machine learning methods to NEXRAD (Next Generation Weather Radar) weather radar data to estimate daily Ambrosia pollen concentrations over a region [9]. Cordero et al., used supervised machine learning algorithms to predict daily *Olea* pollen concentrations in central Spain, showcasing the potential of machine learning in understanding and forecasting pollen risk levels [10]. Yamazaki et al., introduced a simple method for measuring pollen germination rates using machine learning; specifically, the YOLOv5 package for transfer learning [11]. This method could detect germinated and non-germinated pollen, allowing for the estimation of pollen germination rates across different plant species.

The above research results show that machine learning and target detection have a wide range of applications and play a good role in the field of sporology, meeting the needs of life and production and conforming to the trend of combining practical agricultural production with machine learning target detection [12]. In practice, a large amount of image data must be provided to enrich the model and cope with the effects of different lighting conditions and changes in pollen size, shape, and attitude on the model performance; however, this process requires substantial time and effort. Data augmentation is particularly important and helps the model to generalize unseen data by introducing additional variations in the data [13]. Therefore, the model can perform well on the training set and maintain high accuracy on new and different data, reducing the risk of overfitting, improving the robustness of the model to small perturbations or noise in the input data, and enriching the application scenarios of the model [14]. To further adapt to the needs of agricultural production fields in environments with limited computational resources (e.g., mobile devices and embedded systems) and to reduce the cost of applications, we constructed a model named YOLOv8-Pearpollen based on the YOLOv8 target detection model for the pollen emergence viability detection of pear pollen. With this model, we designed two deep lightweight improvements, namely, model pruning and knowledge

distillation, to reduce unnecessary parameters and connections in the neural network, minimize the Params and FLOPs, optimize the efficiency and performance of the model, maximize the use of limited data resources, and improve the model's pervasiveness in the deployment of real hardware.

2. Materials and Methods

2.1. Microphenotypic Trait Observation System

The microphenotypic trait observation system was equipped with a model NSZ818M stereoscopic microscope (NOVEL, Nanjing, China) with a total magnification range of 7.5–135 \times . A magnification of 100 was chosen in this experiment to help improve the accuracy of the pollen grain classification and identification tasks. Installed on the stereoscopic microscope was a Sony industrial microscope digital camera E3ISPM20000KPA (Hangzhou, China), which was connected to the stereoscopic microscope through the C-mount interface and had an image acquisition resolution of 5440 \times 3648 pixels. Clear and detailed images are helpful for machine learning model training and data analysis, and the model comprehensively learned and understood the sample's features [12]. The system was equipped with a four-type light source (coaxial light, ring light, adjustable angled light, and bottom fill light) that enhanced the contrast of the samples and made the fine structure visible by changing the angle and intensity of the illumination. The use of different illumination modes such as transmitted light and reflected light resulted in a special imaging effect, which was conducive to the observation of special microstructures or microscopic phenotypes [15]. Specialized data acquisition software was used to adjust the parameters, such as the exposure and contrast of the captured image data, and to capture and store the images. Finally, the processed dataset was labeled to train the detection model of pollen germination status. Figure 1 illustrates the structure of the microphenotypic trait observation system used in this experiment.

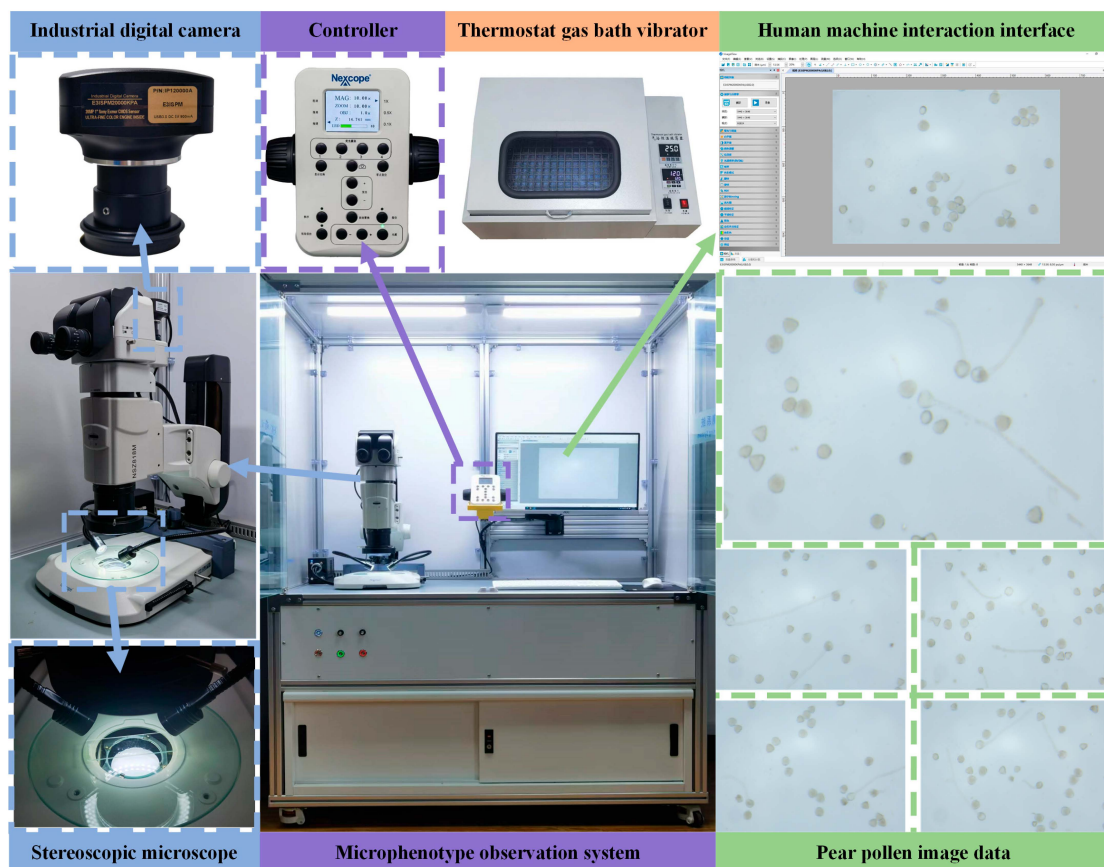


Figure 1. Structure of the microphenotypic trait observation system.

2.2. Germination Image Acquisition and Dataset Construction

The pear pollen used in this experiment was the variety “yellow flower” obtained from the Baima Teaching and Research Base of Nanjing Agricultural University, Nanjing, Jiangsu Province, China. During the macrobud stage, a large number of flower buds were collected, placed inside a room to dry, and rubbed and sieved to obtain a number of pear anthers. The anthers were then packaged in sulfuric acid paper bags, placed in dry containers with color-changing silica gel, and then frozen and sealed at a low temperature for preservation. Prior to use, the anthers were first placed in a 4 °C environment to wake up for 8–12 h and resealed after each use as shown in Figure 2a,b. After the pollen culture solution was configured (10% sucrose + 0.01% boric acid + 0.03% calcium gluconate + 0.04% xanthan gum), a pipette gun was used to transfer 1.5 mL of the culture solution to a 2 mL centrifuge tube. A trace amount of pear pollen (approximately 0.2 mL) was then dispersed and suspended in the culture solution [16]. After the centrifuge tube was fixed, it was placed in a ZD-85 (Changzhou, China) thermostatic gas-bath vibrator (light-proof environment, 25 °C, 120 rpm speed, reciprocating mode amplitude of 25 mm) for 2 h so that the pollen was uniformly suspended and distributed in the culture medium and sprouted pollen tubes. The cultured pollen suspension was evenly spread on a glass slide, which was then placed under a microscope to observe and take images of the pollen germination condition. Seven trials were repeated. After screening, 1500 images saved in the JPEG format were collected throughout the experimental period; the experimental process is shown in Figure 2c.

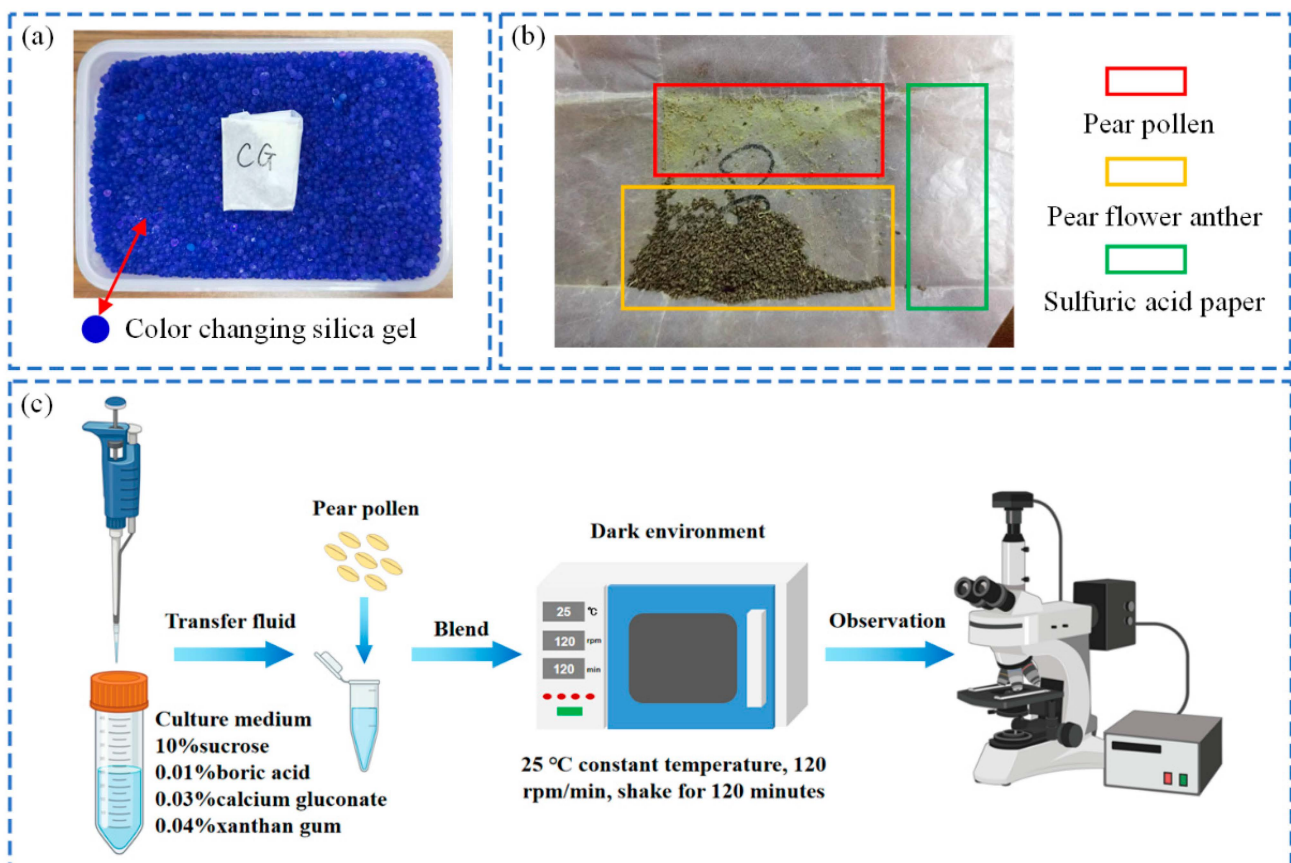


Figure 2. Data acquisition: (a) Pollen storage conditions. (b) “Yellow flower” pear pollen material. (c) Experimental procedure.

Labelling was used to annotate the images of pollen germination status [17]. Whether the length of pollen tube was greater than or equal to the length of pollen grain diameter was used as the criterion to judge if the pear pollen germinated or not, and the results were

stored as an .xml file. The labeled xml file was batch converted to a txt coordinate label file to obtain the normalized length, width, and coordinates of the center point of the labeled box and the target categories of “sprout” and “not sprout” [18]. “Sprout” represents pear pollen that meets the conditions for sprouting, and “not sprout” represents pear pollen that has not sprouted or does not meet the conditions for sprouting. The collected images of pear pollen sprouting conditions are shown in Figure 3a–c.

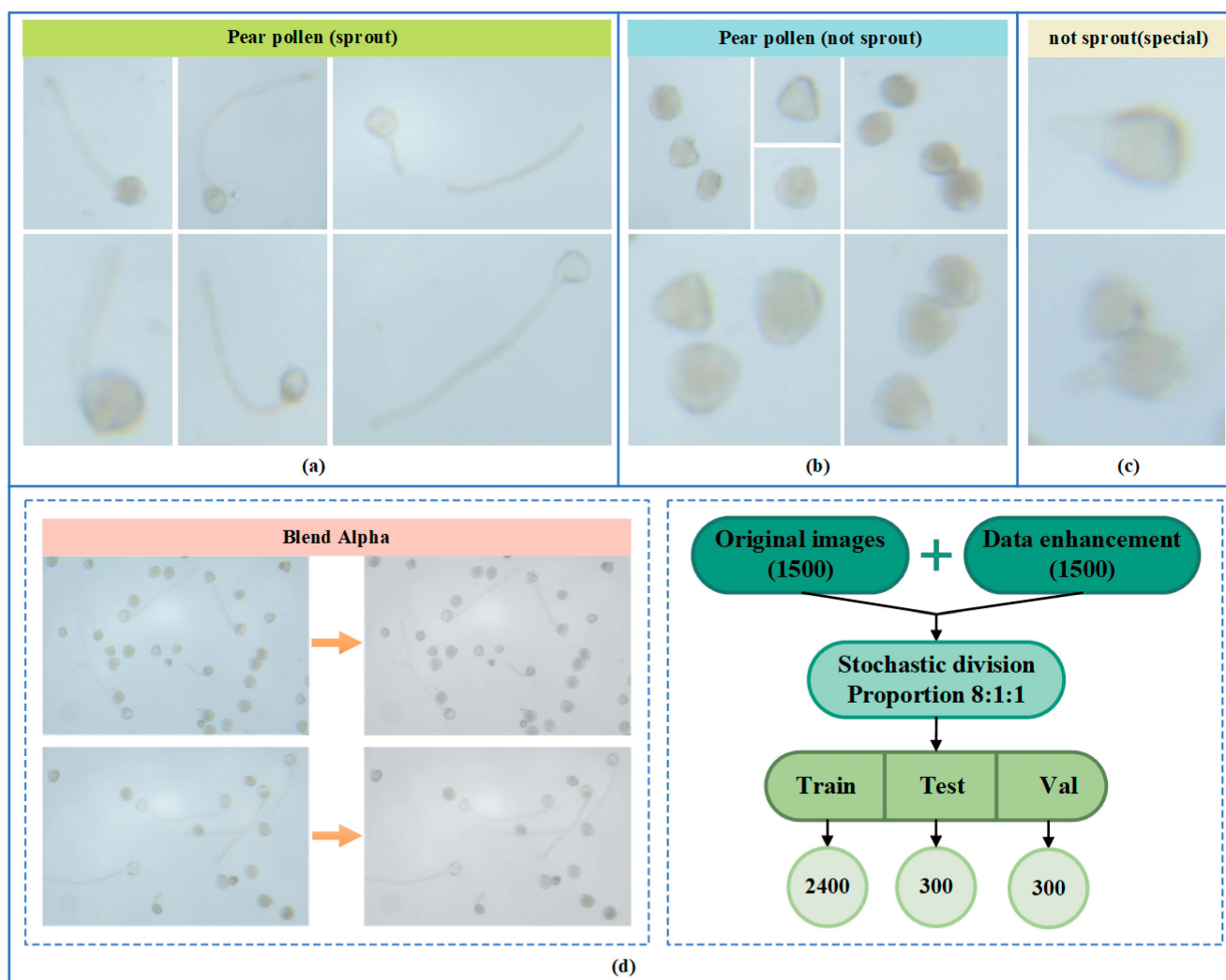


Figure 3. Dataset construction: (a) Schematic of germinated pollen. (b) Schematic of ungerminated pollen. (c) Schematic of ungerminated pollen that does not satisfy the judgment conditions for germination. (d) Data enhancement and dataset construction.

The main factors affecting the accuracy of pollen germination viability detection in pear trees are pollen color, detection background color, and pollen growth attitude. To avoid unnecessary labor and time costs, increase the robustness of the detection method, and effectively expand the dataset [19], we chose the Blend Alpha method to enhance the obtained image data. The graph contains three color channels, red, green, and blue (RGB), and an Alpha channel for recording the transparency of each pixel in the image, with values ranging from 0 (completely transparent) to 1 (completely opaque). Blend Alpha processes the image through transparency mixing and achieves the synthesis and enhancement of the image by adjusting the image Alpha value. The formula for calculating the synthesized pixel value is shown in Equation (1), where C_{result} is the synthesized pixel color value; $C_{\text{foreground}}$ and $C_{\text{background}}$ are the pixel color values of the foreground and background, respectively; and α is the transparency value of the foreground pixel. The result from Alpha

was blended with 50% of the original image, resulting in the removal of 50% of all colors to create a certain grayscale effect. The image result formula is shown in Equation (2), which is linearly interpolated between the foreground branch (FG) and background branch (BG) according to the overlay factor (0.5 in this experiment). The obtained final dataset had 3000 images, including 1500 original images and 1500 images enhanced by data, which were randomly divided into training, validation, and test sets according to the ratio of 8:1:1, as shown in Figure 3d.

$$C_{result} = C_{foreground} * \alpha + C_{background} * (1 - \alpha) \tag{1}$$

$$FinalPic = factor * FG + (1 - factor) * BG \tag{2}$$

2.3. YOLOv8-Pearpollen Design

To improve the detection accuracy of the model and reduce its computation, complexity, and cost for hardware deployment [11], we designed YOLOv8-Pearpollen based on YOLOv8n. Its detailed structure is shown in Figure 4. Relative to YOLOv8, the following optimizations were made for YOLOv8-Pearpollen: (1) The lightweight design of knowledge distillation was added, offline distillation was chosen for migrating the knowledge of the pretrained teacher model to the student network [20], and the distillation method that combines feature and logical distillation was adopted to distill the feature layer in the model and the probability distribution of the model output so that the model’s performance would be close to or reach the performance of a model with small volume, improving its performance with small volumes. The performance of a large-volume model was used as the basis to improve the efficiency of the proposed model. (2) The lightweight design of model pruning was incorporated, and the method of sparse group Lasso was used through the efficient coordinate descent method algorithm to achieve rapid convergence even when dealing with large datasets [21]. The iterative optimization of the coefficients within each cluster ensured model accuracy while maintaining computational efficiency. The features of the model were automatically selected to effectively cut out unimportant feature coefficients, thus pruning the model to improve its generalization performance and enhance its interpretability, which is invaluable for practical applications.

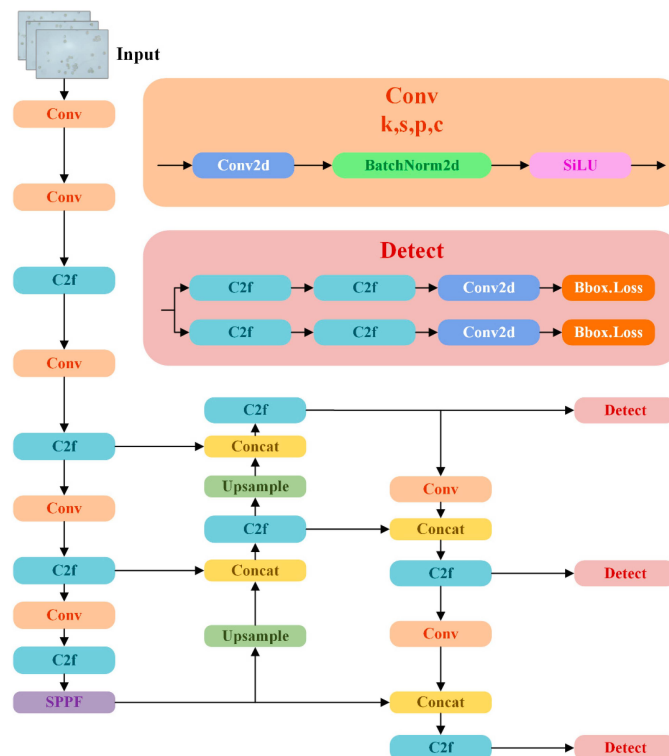


Figure 4. Structure of YOLOv8-Pearpollen.

2.3.1. Knowledge Distillation

A complex detection structure is usually required for the accurate and precise detection of small targets such as pear tree pollen. This structure must be specially designed to capture the fine features of small targets, leading to an increase in the complexity and volume of the model. To address these difficulties, this study introduced knowledge distillation for improving the small target detection performance of YOLOv8-Pearpollen. Knowledge distillation is a deep learning technique that migrates the knowledge from a large and complex model (“teacher” model) to a simple and small model (“student” model) [22]. We trained the dataset for 400 rounds using the YOLOv8l large-volume model to obtain a voluminous but accurate teacher model. YOLOv8n was then used as the basis for the student model and trained it by knowledge distillation to learn the detection effect of the teacher model. In terms of distillation methods, knowledge distillation is divided into logic or feature distillation. The soft labels of the teacher model and the manually labeled real labels were used to cotrain the student model. The soft labels contain information about the relationships between the different categories, not just the hard labels (the final classification results). Meanwhile, the soft output of the teacher model served as a regularization to help generalize the student model to unseen data and allow it to learn the subtle patterns that the teacher model learned on the training data.

Logic distillation utilizes the outputs prior to the last fully connected layer of the model network as a carrier of knowledge, allowing the student model to learn the nuances of the teacher model’s recognition of categories [23]. Rather than just the final probabilistic outputs or categorical labels, the output distributions of both models are also matched. The complexity of the model is controlled, and overfitting is prevented by introducing L1 regularization. The total training loss of the logic distillation process ($Loss_{logical}$) is shown in Equation (3), where $Loss_{distill}$ is the base loss function of the distilled model, and λ is a nonnegative regularization hyperparameter. As the value of λ increases, the impact of the regularization term on the total loss also increases, thus making the model sparser. Conversely, the smaller the value of λ , the smaller the regularization term’s contribution to the total loss, and the complexity of the model increases accordingly. The weight vector is w . Meanwhile, L1 regularization limits the size of the model parameters by adding the absolute values of the weight vectors to the loss function, thus inducing sparse feature learning in the model [24].

$$Loss_{logical} = Loss_{distill} + \lambda \sum_{i=1}^n |w_i| \quad (3)$$

Different from logic distillation that focuses on knowledge transfer in the output layer of the model, feature distillation concentrates on the transfer of feature representations in the internal layers (e.g., convolutional layers). By minimizing the differences between the feature representations of the student and teacher models on these internal layers, the student model can learn the rich representations and the high-level abstraction capabilities of the teacher model. The principle of feature distillation is illustrated in Figure 5. Channel-wise distillation [25] focuses on the channel level of the model’s internal features. For the feature maps of the student and teacher models, we used the softmax function to obtain the normalized probability distributions. Equation (4) calculates the distillation loss in the channel direction, where T and S are the teacher and student models, respectively, and y^T and y^S are their corresponding activation maps. Converting the activation values to probability distributions aims to make the feature maps of the student model as close as possible to those of the teacher model in the channel dimension and to realize a fine-grained and efficient knowledge distillation process as shown in Equation (5), where $c = 1, 2, \dots, C$ is the channel, and i is the spatial location of the channel. The temperature parameter \mathcal{T} is introduced to regulate the smoothness of the softmax function and control the “softness”

of the output distribution. A high \mathcal{T} leads to a flat distribution, and a low value leads to a sharp distribution [26].

$$L_{CWD} = \varphi(\phi(y^T), \phi(y^S)) = \varphi(\phi(y_c^T), \phi(y_c^S)) \tag{4}$$

$$\phi(y_c) = \frac{\exp\left(\frac{y_{c,i}}{\mathcal{T}}\right)}{\sum_{i=1}^{W \cdot H} \exp\left(\frac{y_{c,i}}{\mathcal{T}}\right)} \tag{5}$$

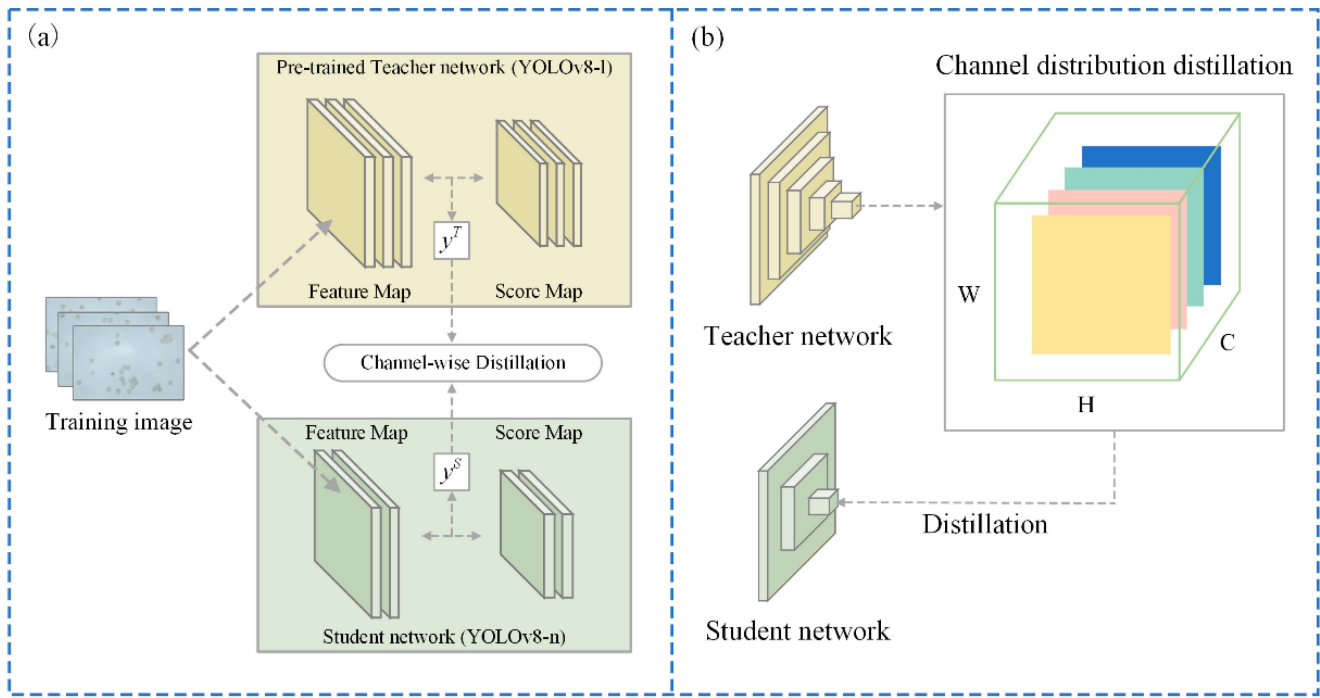


Figure 5. (a) Structural map of feature distillation. (b) Aligning each channel of the student feature map with the channels of the teacher network by minimizing KL divergence.

Knowledge distillation can be applied by determining the Kullback–Leibler (KL) divergence between the feature maps of the teacher and student models [27]. The difference between the outputs of the student and teacher networks was quantified using the KL scatter, which guides the student network to adjust its parameters to mimic the behavior of the teacher network, and was calculated as shown in Equation (6), where C denotes the total number of channels, and W and H denote the width and height of the feature map, respectively. Scatter calculation was performed for each channel and the scatter values from all spatial locations in each channel were summed to ensure that every spatial location within each channel is taken into account. This process helps accurately align the feature responses of the teacher and student networks at the spatial locations and ensures that the student network learns as comprehensively as possible from the teacher network’s knowledge [28].

$$\varphi(y^T, y^S) = \frac{\mathcal{T}^2}{C} \sum_{c=1}^C \sum_{i=1}^{W \cdot H} \phi(y_{c,i}^T) \cdot \log \left[\frac{\phi(y_{c,i}^T)}{\phi(y_{c,i}^S)} \right] \tag{6}$$

The total training loss of the student model of the knowledge distillation process ($Loss_{Distillation}$) is shown in Equation (7) and consists of two components, $Loss_{feature}$ and

L_{CWD} . $Loss_{feature}$ represents the loss of the student model when performing the instance segmentation task and α is an adjustable hyperparameter.

$$Loss_{Distillation} = Loss_{feature} + \alpha L_{CWD} \quad (7)$$

In this experiment, we used a combination of feature and logical distillation to enable the student model to simultaneously acquire knowledge from the internal representation of the teacher's model and the output decision-making level, realizing a comprehensive knowledge transfer. The student model not only learns the final predictions of the teacher model but also understands the abstract feature processing behind arriving at those predictions. Hence, it can further understand the essential features of the input data, improve generalization ability, and achieve or approach the performance of the teacher model while maintaining a small model size. For model deployment under limited resources, such as in mobile devices and embedded systems, the model size and computational requirements can be drastically reduced while maintaining predictive performance, providing great flexibility for a wide range of tasks and model architectures.

2.3.2. Model Pruning

The distilled YOLOv8-Pearpollen model was structurally pruned to eliminate the nonessential parts of the model and reduce its complexity while keeping it as accurate as possible.

The sparse group-lasso [29] achieves sparsity at the group and individual feature levels by combining the penalty terms of lasso (L1 regularization) [30] and group-lasso (L2 regularization) [31]. This approach selects important feature clusters and performs feature selection within each selected cluster, further reducing model complexity and improving model interpretability [32]. As shown in Equation (8), X_ℓ and β_ℓ denote the model matrix and coefficient vector of the l th cluster, respectively, and λ_1 and λ_2 are the regularization parameters that control the strength of the group-lasso and lasso penalties. Equation (9) is the objective function to minimize the sum of squares of the differences between the model-predicted and actual values [33]. Equation (10) is the group-lasso penalty, which applies the L2 paradigm to the coefficient vector β_ℓ of each group and adjusts the effect of group size by $\sqrt{p_l}$ to achieve group-level sparsity, i.e., the entire group can be selected in or excluded. Equation (11) is the lasso penalty, which applies the L1 paradigm to all coefficients to achieve sparsity at the individual feature level, i.e., only a fraction of the important predictor variables are included in the model. λ_1 and λ_2 are the regularization parameters that control the strength of the group-lasso and lasso penalties. A balance between model complexity and sparsity can be achieved by tuning these parameters. The generation of sparsity was changed to the group and individual feature levels by group-lasso to accurately select predictors between and within groups.

$$\min_{\beta \in R^p} \left(\left\| y - \sum_{\ell=1}^L X_\ell \beta_\ell \right\|_2^2 + \lambda_1 \sum_{\ell=1}^L \sqrt{p_l} \left\| \beta_\ell \right\|_2 + \lambda_2 \left\| \beta \right\|_1 \right) \quad (8)$$

$$\left\| y - \sum_{\ell=1}^L X_\ell \beta_\ell \right\|_2^2 \quad (9)$$

$$\lambda_1 \sum_{\ell=1}^L \sqrt{p_l} \left\| \beta_\ell \right\|_2 \quad (10)$$

$$\lambda_2 \left\| \beta \right\|_1 \quad (11)$$

The sparse group-lasso can be calculated from the coordinate descent by choosing an initial coefficient vector β_0 , which is usually a zero vector or any estimate based on a priori knowledge. For each group l , the residuals are shown in Equation (12), i.e., the target value minus the current predicted values for all other clusters excluding group l . The predictors were denoted as $X_\ell = Z = (Z_1, Z_2, \dots, Z_k)$, and the coefficients were denoted as $\beta_\ell = \theta = (\theta_1, \theta_2, \dots, \theta_k)$. For each feature j within the group, $w_j = (w_1, w_2, \dots, w_N) = r - \sum_{k \neq j} Z_k \theta_k$ is

the updated residual, excluding the contribution of the current feature j . If $|Z_j^T w_j|$ is less than λ_2 , then it will be directly set to 0, otherwise, θ_j will be set to the minimized value of Equation (13). If the whole coefficient vector is zero is not satisfied, then each coefficient must be optimized and the steps must be repeated until the coefficients of all groups reach the convergence condition [34]. Only one coefficient or one group of coefficients must be updated at a time, and the global optimal solution can be obtained through stepwise approximation. The key advantage of the coordinate descent method is its high efficiency, especially when dealing with high-dimensional data with a group structure. It enables fast and accurate feature selection, thus enhancing the explanatory and predictive performance of the model.

$$r_\ell = y - \sum_{k \neq \ell} X_k \hat{\beta}_k \quad (12)$$

$$\frac{1}{2} \sum_{i=1}^N \left(w_i - \sum_{j=1}^k Z_{ij} \theta_j \right)^2 + \lambda_1 \|\theta\|_2 + \lambda_2 \sum_{j=1}^k |\theta_j| \quad (13)$$

3. Results and Discussion

3.1. Training Environment and Hyperparameter Settings

The hardware configuration for this experiment comprised an Intel(R) Xeon(R) Gold 6248R @ 3.00 GHz processor, an NVIDIA GeForce RTX3090 graphics card, and an operating system of Windows 11. The environment configuration for deep learning consisted of Python 3.11, Pytorch 2.1.0, CUDA 12.2, and Torchvision 0.16.0. The important hyperparameter settings during the training are shown in Table 1. No pretraining weights were used in any part of this experiment to ensure the fairness and effectiveness of the detection model for pear pollen germination vigor.

Table 1. Model hyperparameter setting.

Parameters	Setup
Epoch	150
Batch size	32
Optimizer	SGD
Close_mosaic	20
Cache	True
Image size	640 × 640
Initial learning rate	1×10^{-2}
Final learning rate	1×10^{-4}
Momentum	0.937
Weight-decay	5×10^{-4}
Warmup-epochs	3

3.2. Ablation Tests

To test the feasibility and effectiveness of the detection model for pear pollen germination vigor, we designed several sets of ablation experiments, evaluated the degree of contribution of each module to the model performance, and excluded and removed the modules that had a small or negative effect on the final results to make the model concise and efficient.

First, in the design of knowledge distillation, three types of distillation (feature, logical, and hybrid) were set. For feature distillation, the following three types were chosen: channel-wise distillation (CWD), masked generative distillation (MDG), and mimic. For logical distillation, the following three types were selected: L1, L2, and BCKD. The corresponding loss rates λ and α were set. The loss rate quantified the difference between the predictions of the student and teacher models. It must be changed to adjust the weight between the lost items. Adjustments must be made to the similarity between the outputs of the intermediate layers of the student and teacher models in terms of shape, direction, or distribution and the difference between the probability distribution of the outputs of the student and teacher models to ensure that the student model can mimic the teacher

model to deal with internal mechanism data and thus learn an efficient and sophisticated representation of the data. After the above parameters were changed and the accuracy and efficiency of the model were adjusted, the results of the ablation test for knowledge distillation were compared as shown in Table 2. In terms of feature distillation, CWD and MDG were generally better than the other types. Meanwhile, the results of the mimic method for several data were different from those of the above two methods. Even if the loss rate was adjusted, the results were still not satisfactory. In terms of logical distillation, a general improvement was observed over feature distillation. The $AP_{\text{not sprout}}$, $mAP@0.5$, precision, and recall were comparable with those of feature distillation, but the AP_{sprout} was relatively lower. When feature and logical distillation were combined, the data became stable and improved compared with using feature or logical distillation alone. Logical distillation l1 type (loss rate 0.8) and characteristic distillation CWD (loss rate 0.8) stood out among the 17 compared methods, showing the highest values for $mAP@0.5$, AP_{sprout} , and $AP_{\text{not sprout}}$ at 97.6%, 97.6%, and 97.7%, respectively, relative to the baseline models. YOLOv8-n (without distillation) showed improvement in all data, verifying the excellent effect of knowledge distillation on the small-volume model's performance and accuracy improvement.

Table 2. Comparison of test results for knowledge distillation ablation.

Models	$mAP@0.5$ (%)	AP_{sprout} (%)	$AP_{\text{not sprout}}$ (%)	Precision (%)	Recall (%)	FPS
YOLOv8-n	97.0	97.0	97.1	91.4	94.2	149.3
feature cwd loss1.0	97.2	97.6	96.8	94.0	92.6	142.9
feature cwd loss0.8	97.3	97.1	97.4	93.9	93.6	138.9
feature mdg loss1.0	97.2	97.0	97.4	93.4	94.5	147.1
feature mimic loss1.0	96.8	96.7	96.9	92.7	92.7	117.6
feature mimic loss0.8	96.6	96.2	97.1	93.6	92.6	128.2
logical l1 loss1.0	97.1	97.1	97.2	93.7	94.5	140.8
logical l1 loss0.8	96.8	96.3	97.2	93.1	93.5	151.1
logical l2 loss1.0	97.1	96.8	97.5	93.5	93.8	151.1
logical l2 loss0.8	97.6	97.5	97.6	93.9	94.2	144.9
logical BCKD loss1.0	97.1	96.6	97.5	92.6	94.5	142.9
logical BCKD loss0.8	97.5	97.5	97.5	92.4	95.3	153.8
all l2 loss1.0 cwd loss1.0	97.2	97.3	97.1	93.9	94.4	137.0
all l2 loss0.8 cwd loss0.8	97.4	97.5	97.4	94.3	94.5	153.8
all l1 loss1.0 cwd loss1.0	97.2	96.9	97.5	93.7	94.3	151.5
all l1 loss0.9 cwd loss0.9	97.3	97.4	97.2	94.2	94.2	125.0
all l1 loss0.8 cwd loss0.8	97.6	97.6	97.7	94.0	94.7	151.5

To intuitively reflect the superiority of our choice of knowledge distillation method, we drew a hexagonal radar chart using the normalization method to compare the results of the six parameters as shown in Figure 6 below. Through min–max normalization, the original data were linearly transformed so that the results of the processed data were between [0, 1]. The closer the value to 1, the more effective the model; the closer the value to 0, the worse the effect. The transformation formula is shown in Equation (14), where X_{max} is the maximum value of the data, X_{min} is the minimum value of the data, and R_A is the converted ratio of the difference between the maximum and minimum values of the current indicator. The larger the true value, the higher the normalized score on this item.

$$R_A = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \quad (14)$$

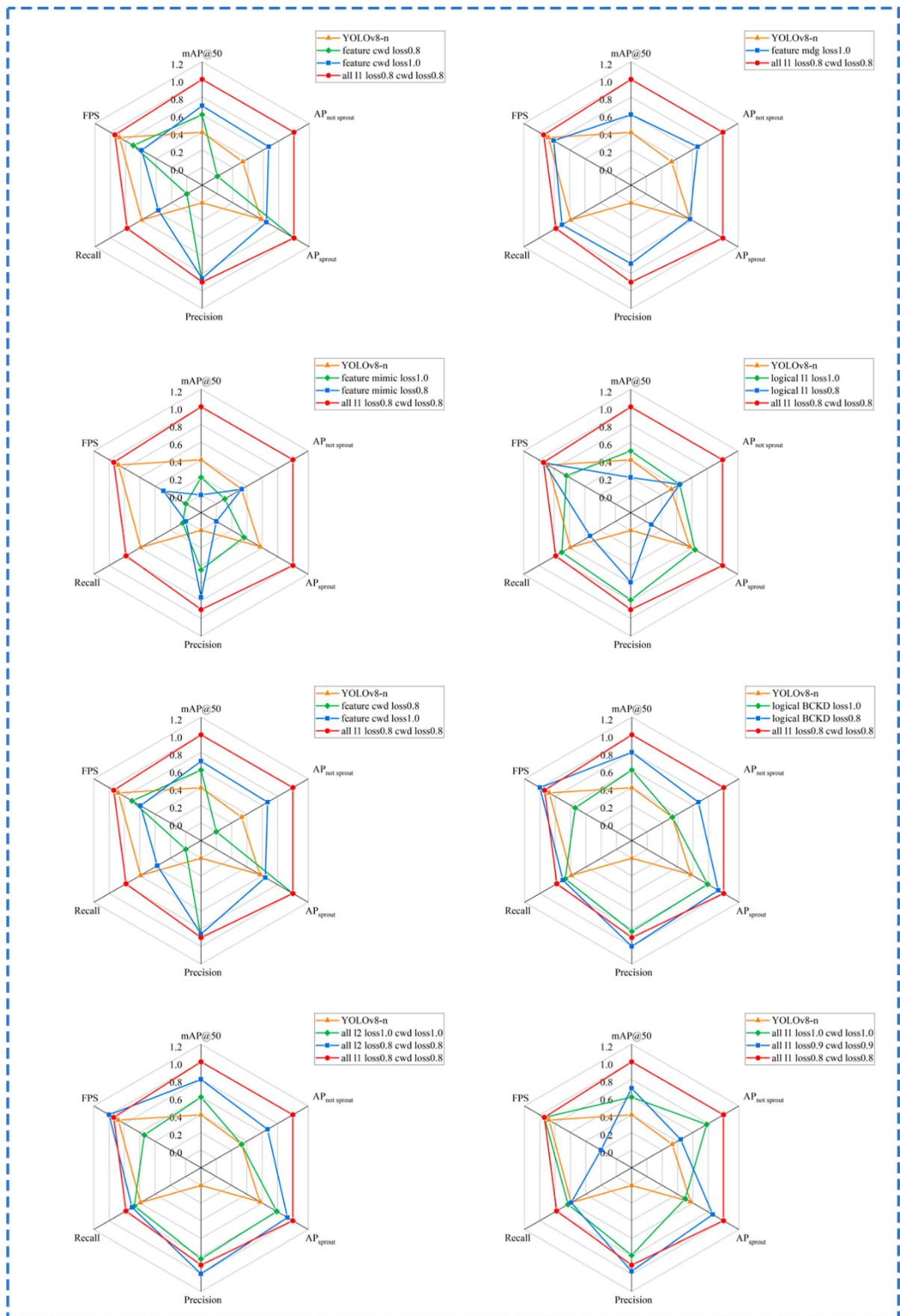


Figure 6. Normalized analysis of multiple indicators of knowledge distillation.

To further explore the effect of feature layer selection in feature distillation, we took the model with the best results and designed an ablation test using this model to determine the effect of knowledge distillation by changing the number of distillation layers under the same parameters. The results are shown in Table 3. When using the feature vectors of the 15th, 18th, and 21st layers as pseudo labels for distillation, the mAP@0.5, AP_{sprout}, and AP_{not sprout} were improved to 97.6%, 97.6%, and 97.7%, respectively, and the precision and recall were improved to 94.0% and 94.7%, respectively, relative to those in the baseline model, YOLOv8-n (which does not use distillation). When the number of distillation layers was increased to six (layers 6, 8, 12, 15, 18, and 21), the AP_{not sprout} of the model improved to 97.7% compared with the use of three layers (layers 15, 18, and 21). Meanwhile, the mAP@0.5 and AP_{sprout} decreased to 97.3% and 96.9%, respectively. Although the recall slightly improved to 95.2%, the precision decreased by one percentage point to 92.9%. When the number of distillation layers was further raised to eight, the AP_{not sprout} of the model did not change, and the mAP@0.5 and AP_{sprout} continued to drop to 97.0% and 96.3%, respectively. These findings demonstrate that choosing the appropriate number of feature layers is the key to optimizing feature distillation.

Table 3. Comparison of ablation test results with the same parameters and changing the number of distillation layers.

Number of Distillation Layers	mAP@0.5 (%)	AP _{sprout} (%)	AP _{not Sprout} (%)	Precision (%)	Recall (%)
none	97.0	97.0	97.1	91.4	94.2
15, 18, 21	97.6	97.6	97.7	94.0	94.7
6, 8, 12, 15, 18, 21	97.3	96.9	97.7	92.9	95.2
2, 4, 6, 8, 12, 15, 18, 21	97.0	96.3	97.7	94.1	93.9

Second, in terms of model pruning design, different pruning methods were adopted. The effect of global pruning and local pruning on precision, Params, and FLOPs was examined, and the results of the ablation test for model pruning were compared as shown in Tables 4 and 5. Four model pruning methods were selected for comparison; namely, lamp [35], groupnorm [36], grouptaylor [37], and groupsilm [38]. Although the lamp method greatly reduced the Params and weights, its mAP@0.5, AP_{sprout}, and AP_{not sprout} were slightly inferior to those of grouptaylor and groupsilm. The mAP@0.5, AP_{sprout}, and AP_{not sprout} of groupnorm were lower than those of the other methods. Especially in the detection of sprouting pollen, its effect was poor. Grouptaylor had a clear advantage in detection accuracy, showing the highest mAP@0.5, AP_{sprout}, and AP_{not sprout}. However, its frame per second (FPS) was significantly reduced, and the model inference speed was slow. Meanwhile, the Params, FLOPs, and weights of groupsilm showed decreases of 50%, 50%, and 48%, respectively, and its accuracy exhibited minimal reduction. Groupsilm was effective in the detection of sprouted pollen, but its accuracy for detecting unsprouted pollen was low. For global pruning, which calculates the proportion of the model's pruning according to all the parameters, we found that the model's pruning was more effective than that of groupsilm. Global pruning, which uses all parameters to calculate the proportion of model clippings, showed a slight improvement in accuracy over local pruning by calculating clippings according to the parameters of each layer. However, its lightweighting effect was significantly inferior to that of local pruning. With the increase in the pruning rate, the Params and FLOPs of the model decreased and its detection performance decreased significantly. Model pruning can effectively and simultaneously reduce the model size and computational cost, ensure the detection accuracy of the model and effectively optimize its detection efficiency, fully utilize the computational resources, reduce the model complexity in the hardware deployment of the threshold, and enrich the application of the model scene.

Table 4. Analysis of ablation test results of model pruning methods.

Pruning Method	mAP@0.5 (%)	APsprout (%)	APnot Sprout (%)	Params (M)	FLOPs (G)	Weights (MB)	FPS
none	97.6	97.5	97.6	3.0	8.1	6.0	151.5
lamp	96.3	95.8	96.8	0.8	4.0	1.7	147.1
groupnorm	92.4	89.2	95.6	1.2	4.0	2.6	156.3
grouptaylor	97.3	97.7	96.8	1.6	4.0	3.2	122.0
groupsilm	96.7	96.4	97.1	1.5	4.0	3.1	147.1

Table 5. Analysis of the ablation test results of different model pruning rates with the same methodology.

Pruning Rate	mAP@0.5 (%)	APsprout (%)	APnot Sprout (%)	Params (M)	FLOPs (G)	Weights (MB)	FPS
1.5 (global)	97.6	97.3	97.8	2.7	5.4	5.3	144.9
2.0 (global)	96.9	96.7	97.2	2.4	4.0	4.8	142.9
2.0 (nonglobal)	96.7	96.4	97.1	1.5	4.0	3.1	147.1
2.5 (global)	95.1	94.5	95.7	2.1	3.2	4.3	144.9
3.0 (global)	93.7	91.7	95.6	1.7	2.7	3.4	128.2

The designed detection model for pear pollen germination vigor was compared with the YOLOv8-n benchmark detection model, and the experimental results are demonstrated in Table 6. After the model was subjected to knowledge distillation, its mAP@0.5, APsprout, and APnot sprout improved to 97.6%, 97.5%, and 97.6%, respectively. After the model was pruned, its Params, FLOPs, and weights were further reduced by 50%, 50.6%, and 48.3% respectively, relative to their original values. Meanwhile, its FPS reached 147.1, which was a reduction of 4.4 compared with that of the unpruned one. In terms of accuracy, the APnot sprout only decreased by 0.6%, and the mAP@0.5 and APsprout decreased by only 0.9% and 1.2%, respectively. The ablation test demonstrated the effectiveness of the detection model for pear pollen germination viability in terms of lightweighting and the superiority of its accuracy, further proving the feasibility of this experimental design for pear pollen detection.

Table 6. Feature distillation and structured pruning validity analysis.

Models	mAP@0.5 (%)	APsprout (%)	APnot Sprout (%)	Params (M)	FLOPs (G)	Weights (MB)	FPS
YOLOv8-n	97.0	97.0	97.1	3.0	8.1	6.0	149.3
+distill	97.6	97.6	97.7	3.0	8.1	6.0	151.5
+distill +prune	96.7	96.4	97.1	1.5	4.0	3.1	147.1

3.3. Comparative Tests

YOLOv8-Pearpollen was compared with multigeneration models from YOLOv3 to YOLOv8 to further prove its superiority for the detection of pear pollen germination vigor. The following outcome indicators were evaluated: mAP50 (which includes the accuracy of the single identification of pollen germination and pollen nongermination results), Params, FLOPs, and weights. The results are shown in Table 7. Compared with those of the other models, the Params, FLOPs, and weights of YOLOv8-Pearpollen were significantly lower at 1.5 M, 4.0 G, and 3.1 MB, respectively. For YOLOv3, which has a high accuracy, its Params, FLOPs, and weights decreased by 98.6%, 98.6%, and 98.4%, respectively, relative to those of the model in agriculture and by 75.0%, 69.2%, and 73.5%, respectively, compared with those of YOLOv7-tiny, a commonly used target detection model in agriculture. In terms of detection accuracy, except for the large-volume YOLOv3, YOLOv8-Pearpollen had the best AP_{not sprout}, which was 0.3%, 1.0%, and 0.3% higher than that of YOLOv5-n, YOLOv6-n, and YOLOv7-tiny, respectively, and its mAP@0.5 and AP_{sprout} were also higher than those

of YOLOv5-n and YOLOv7-tiny. The comparison test proved that the YOLOv8-Pearpollen model designed in this experiment has outstanding performance and can maintain superior detection accuracy while reducing the computational complexity. It has advantages over other models, is suitable and easy to deploy in hardware devices (such as embedded devices and mobile devices) to satisfy the actual needs of agricultural production, and can accurately detect the pollen germination vigor of pear trees on devices in real time.

Table 7. Comparison of detection results of multiple models.

Models	mAP@0.5 (%)	AP _{sprout} (%)	AP _{not sprout} (%)	Params (M)	FLOPs (G)	Weights (MB)	FPS
YOLOv3	98.3	98.0	98.6	103.7	282.2	198.1	102.0
YOLOv5-n	96.2	95.6	96.8	2.5	7.1	5.0	142.9
YOLOv6-n	94.2	92.2	96.1	4.2	11.8	8.3	158.7
YOLOv7-tiny	95.2	93.7	96.8	6.0	13.0	11.7	90.0
YOLOv8-n	97.0	97.0	97.1	3.0	8.1	6.0	149.3
YOLOv8-Pearpollen	96.7	96.4	97.1	1.5	4.0	3.1	147.1

To further demonstrate the superiority of YOLOv8-Pearpollen’s detection effect over that of the other models, we plotted its multimetric normalized histogram as shown in Figure 7. The conversion formulas for the three metrics of model parameter count, computation, and weight file size are shown in Equation (15). Different from Equation (13), R_B is the ratio of the maximum value of the current indicator minus the value of the indicator that needs to be converted to the difference between the maximum and minimum values of the current indicator. The smaller the true value, the higher the normalized score on this item, fitting the correct comparison of the model parametric quantity, computational quantity, and weight file size indicators. YOLOv8-Pearpollen reaches the highest level in four data items, namely, the number of model parameters, amount of computation, size of the weight file, and $AP_{not\ sprout}$. It is also in the forefront in terms of mAP and AP_{sprout} , and its overall performance reaches the optimal level, far ahead of the other models.

$$R_B = \frac{X_{max} - X}{X_{max} - X_{min}} \tag{15}$$

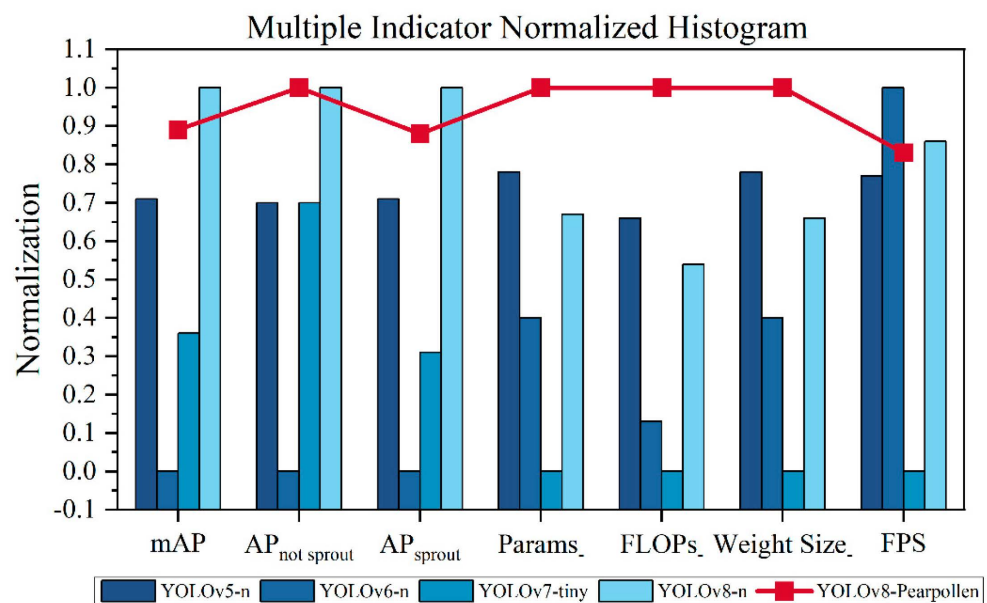


Figure 7. Normalized analysis of multiple model detection results.

3.4. Evaluation of Model Prediction Performance

We analyzed the recognition effect of YOLOv8-Pearpollen compared with three lightweight models, YOLOv5-n, YOLOv6-n, and YOLOv7-tiny, for pear tree pollen germination. Six images were randomly selected from 1800 images for the comparison test. The detection comparison results are shown in Figure 8 and Table 8. The pink box area in the figure indicates the identified sprouted pear pollen, the red box area indicates the identified not sprouted pear pollen, and the green circle, yellow circle, and blue circle indicate the missed detection, repeated detection, and wrong detection, respectively.

The results show that YOLOv5-n and YOLOv6-n often had repeated detection conditions, especially in pollen-dense or superimposed scenarios. Meanwhile, YOLOv7-tiny showed a certain degree of missed detection, which was rare in the detection results of the other models. Overall, YOLOv8-Pearpollen had almost no repeat detection, which is a huge improvement over the other four models, and accurately recognized pear pollen germination in complex recognition scenarios.



Figure 8. Comparison of the detection results of YOLOv8-Pearpollen, YOLOv5-n, and YOLOv6-n, YOLOv7-tiny, and YOLOv8-n.

Table 8. Statistical table comparing the detection results of different models.

Models	Wrong Detection	Repeated Detection	Missed Detection
YOLOv5-n	4	7	0
YOLOv6-n	3	6	1
YOLOv7-tiny	6	3	2
YOLOv8-n	2	5	0
YOLOv8-Pearpollen	4	0	0

4. Conclusions

Traditional manual pear tree pollen detection methods are cumbersome and have low efficiency, insufficient accuracy, and high labor costs. Screening high-quality male plant pollen is necessary to improve the success rate of artificial pollination for pear tree pollen. In addition, the cost of deploying the model on hardware must be minimized to further adapt to the requirements of actual agricultural production. To address the above problems, this study proposed a lightweight detection model for pear pollen germination named YOLOv8-Pearpollen. The following are the accomplishments of this work:

- (1) A pear tree pollen dataset was collected from several trials, and the original image data were enhanced using Blend Alpha to improve the robustness of the detection method, optimize its detection effect against a complex environmental background, and reduce the labor and time costs required for image collection.
- (2) Two designs of knowledge distillation and model pruning were used for the model to improve its lightness of weight, reduce its deployment cost on hardware equipment, and adapt to the actual needs of agricultural production. In terms of knowledge distillation, we chose to use a combination of logical and feature distillation so that the student model could learn from the teacher model and the internal layer and output decision-making to achieve or approach the detection effect of the large-volume teacher model. A number of ablation tests were carried out on different knowledge distillation methods to select the best method, and the detection results were compared under different numbers of distillation layers but the same method to determine effect of the number of distillation layers on the distillation effect. The structured pruning method of sparse-group lasso was used for model pruning to reduce the number of model calculations and maintain the detection performance of the model as much as possible. A variety of model pruning methods were also tested. The detection effects of different pruning rates of the same method were analyzed to obtain the best model pruning effect and ensure that the model has a high degree of lightweighting while showing good detection accuracy.
- (3) YOLOv8-Pearpollen was compared with YOLOv3, YOLOv5-n, YOLOv6-n, YOLOv7-tiny, and YOLOv8-n detection models in terms of detection accuracy and model complexity. The results were statistically analyzed using the normalization method so that the strengths and weaknesses of the different models in various indicators are intuitively derived and to prove the effectiveness and superiority of the proposed method.
- (4) The test results showed that YOLOv8-Pearpollen achieved a mAP of 96.7%. However, its number of model parameters, amount of computation, and size of the weight file were only 1.5 M, 4.0 G, and 3.1 MB, respectively, which were 50.0%, 50.6%, and 48.3% lower than those of YOLOv8, respectively. It realized a detection speed of 147.1 FPS. Compared with other models, YOLOv8-Pearpollen achieved a high degree of lightweighting while possessing superior detection accuracy, which is suitable for deployment on hardware models and meets the actual needs of agricultural production.

The proposed method still has some limitations. The detection of overlapping or stacked pear pollen is prone to erroneous detection caused by overlapping images that make the model unable to determine which pollen tube belongs to which pollen. Semantic

segmentation can be considered to annotate the image data, helping the model to detect the pollen of sprouting pear trees.

We hope that our research on the lightweight identification and detection of pollen germination vigor in pear trees can provide effective methods and valuable references for the screening of high-quality male pollen. The findings will help improve the success rate of artificial pollination and thus increase the fruiting rate of pear trees. The lightweighting degree of the model must be further enhanced so it can be deployed on hardware facilities to promote the practical agriculture production of fruit trees.

Author Contributions: Conceptualization, W.S. and C.C.; methodology, W.S. and C.C.; software, W.S. and C.C.; validation, W.S., C.C., T.L. and F.H.; formal analysis, W.S., C.C. and H.J.; investigation, W.S., C.C. and S.H.; resources, M.N.; data curation, W.S. and C.C.; writing—original draft preparation, W.S. and C.C.; writing—review and editing, F.H. and X.F.; visualization, W.S. and C.C.; supervision, H.J. and L.T.; project administration, W.S. and C.C.; funding acquisition, X.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Jiangsu Province Innovation Support Plan (Rural Industrial Revitalization) (Grant number SZ-SY20231103), Major Science and Technology Projects of Xinjiang Academy of Agricultural and Reclamation Sciences (Grant number NCG202407), the Jiangsu Agriculture Science and Technology Innovation Fund (JASTIF) (Grant number (CX(23)3619), Hainan Seed Industry Laboratory (Grant number B21HJ1005), and Jiangsu Province Seed Industry Revitalization Unveiled Project (Grant number JBGS(2021)007).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to possible further research.

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

References

- Huete, A.; Tran, N.N.; Nguyen, H.; Xie, Q.; Katelaris, C. Forecasting pollen aerobiology with Modis EVI, land cover, and phenology using machine learning tools. In Proceedings of the 2019 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2019), Yokohama, Japan, 28 July–2 August 2019; pp. 5429–5432.
- Agudelo, C.G.; Sanati Nezhad, A.; Ghanbari, M.; Naghavi, M.; Packirisamy, M.; Geitmann, A. TipChip: A modular, MEMS-based platform for experimentation and phenotyping of tip-growing cells. *Plant J.* **2013**, *73*, 1057–1068. [[CrossRef](#)]
- Dechkrong, P.; Srirama, S.; Nilwananon, T.; Tongyoo, P.; de Jong, H.; Chunwongse, J. Morphological Characterization of Anther and Pollen Formation in an EMS Induced Tomato Mutant with Blossom Drop Phenotype. *Plant Biol. Crop Res.* **2020**, *1*, 1030.
- Zhang, M.; Zhao, J.; Hoshino, Y. Deep learning-based high-throughput detection of in vitro germination to assess pollen viability from microscopic images. *J. Exp. Bot.* **2023**, *74*, 6551–6562. [[CrossRef](#)] [[PubMed](#)]
- Punyasena, S.W.; Tcheng, D.K.; Wesseln, C.; Mueller, P.G. Classifying black and white spruce pollen using layered machine learning. *New Phytol.* **2012**, *196*, 937–944. [[CrossRef](#)] [[PubMed](#)]
- Liu, X.; Wu, D.; Zewdie, G.K.; Wijerante, L.; Timms, C.I.; Riley, A.; Levetin, E.; Lary, D.J. Using machine learning to estimate atmospheric Ambrosia pollen concentrations in Tulsa, OK. *Environ. Health Insights* **2017**, *11*, 1178630217699399. [[CrossRef](#)] [[PubMed](#)]
- Sobol, M.K.; Finkelstein, S.A. Predictive pollen-based biome modeling using machine learning. *PLoS ONE* **2018**, *13*, e0202214. [[CrossRef](#)] [[PubMed](#)]
- Sobol, M.K.; Scott, L.; Finkelstein, S.A. Reconstructing past biomes states using machine learning and modern pollen assemblages: A case study from Southern Africa. *Quat. Sci. Rev.* **2019**, *212*, 1–17. [[CrossRef](#)]
- Zewdie, G.K.; Lary, D.J.; Levetin, E.; Garuma, G.F. Applying deep neural networks and ensemble machine learning methods to forecast airborne ambrosia pollen. *Int. J. Environ. Res. Public Health* **2019**, *16*, 1992. [[CrossRef](#)] [[PubMed](#)]
- Cordero, J.M.; Rojo, J.; Gutiérrez-Bustillo, A.M.; Narros, A.; Borge, R. Predicting the Olea pollen concentration with a machine learning algorithm ensemble. *Int. J. Biometeorol.* **2021**, *65*, 541–554. [[CrossRef](#)] [[PubMed](#)]
- Yamazaki, A.; Takezawa, A.; Nagasaka, K.; Motoki, K.; Nishimura, K.; Nakano, R.; Nakazaki, T. A simple method for measuring pollen germination rate using machine learning. *Plant Reprod.* **2023**, *36*, 355–364. [[CrossRef](#)] [[PubMed](#)]
- Tian, G.; Li, X.; Wu, Y.; Liu, A.; Zhang, Y.; Ma, Y.; Guo, W.; Sun, X.; Fu, B.; Li, D. Recognition effect of models based on different microscope objectives. In Proceedings of the 3rd International Symposium on Artificial Intelligence for Medicine Sciences, Amsterdam, The Netherlands, 13–15 October 2022; pp. 133–141.
- Zewdie, G.K.; Lary, D.J.; Liu, X.; Wu, D.; Levetin, E. Estimating the daily pollen concentration in the atmosphere using machine learning and NEXRAD weather radar data. *Environ. Monit. Assess.* **2019**, *191*, 418. [[CrossRef](#)] [[PubMed](#)]

14. Tan, Z.; Yang, J.; Li, Q.; Su, F.; Yang, T.; Wang, W.; Aierxi, A.; Zhang, X.; Yang, W.; Kong, J.; et al. PollenDetect: An open-source pollen viability status recognition system based on deep learning neural networks. *Int. J. Mol. Sci.* **2022**, *23*, 13469. [[CrossRef](#)] [[PubMed](#)]
15. Wang, C.; Gui, C.P.; Liu, H.K.; Zhang, D.; Mosig, A. An Image Skeletonization-Based Tool for Pollen Tube Morphology Analysis and Phenotyping. *J. Integr. Plant Biol.* **2013**, *55*, 131–141. [[CrossRef](#)] [[PubMed](#)]
16. Gallardo, R.; García-Orellana, C.J.; González-Velasco, H.M.; García-Manso, A.; Tormo-Molina, R.; Macías-Macías, M.; Abengózar, E. Automated multifocus pollen detection using deep learning. *Multimed. Tools Appl.* **2024**, *83*, 72097–72112. [[CrossRef](#)]
17. Fu, X.; Bai, Y.; Zhou, J.; Zhang, H.; Xian, J. A method for obtaining field wheat freezing injury phenotype based on RGB camera and software control. *Plant Methods* **2021**, *17*, 120. [[CrossRef](#)] [[PubMed](#)]
18. Iang, H.; Hu, F.; Fu, X.; Chen, C.; Wang, C.; Tian, L.; Shi, Y. YOLOv8-Peas: A lightweight drought tolerance method for peas based on seed germination vigor. *Front. Plant Sci.* **2023**, *14*, 1257947.
19. Zewdie, G.K.; Liu, X.; Wu, D.; Lary, D.J.; Levetin, E. Applying machine learning to forecast daily Ambrosia pollen using environmental and NEXRAD parameters. *Environ. Monit. Assess.* **2019**, *191*, 261. [[CrossRef](#)]
20. Shu, C.; Liu, Y.; Gao, J.; Yan, Z.; Shen, C. Channel-wise knowledge distillation for dense prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 5311–5320.
21. Behnke, M.; Heafield, K. Pruning neural machine translation for speed using group lasso. In Proceedings of the Sixth Conference on Machine Translation, Online, 10–11 November 2021; pp. 1074–1086.
22. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531.
23. Wen, W.; Wu, C.; Wang, Y.; Chen, Y.; Li, H. Learning structured sparsity in deep neural networks. *Adv. Neural Inf. Process. Syst.* **2016**, *29*.
24. Zheng, Z.; Ye, R.; Wang, P.; Ren, D.; Zuo, W.; Hou, Q.; Cheng, M.M. Localization distillation for dense object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 9407–9416.
25. Vidaurre, D.; Bielza, C.; Larranaga, P. A survey of L1 regression. *Int. Stat. Rev.* **2013**, *81*, 361–387. [[CrossRef](#)]
26. Cai, T.T.; Zhang, A.R.; Zhou, Y. Sparse group lasso: Optimal sample complexity, convergence rate, and statistical inference. *IEEE Trans. Inf. Theory* **2022**, *68*, 5975–6002. [[CrossRef](#)]
27. Fang, G.; Ma, X.; Song, M.; Mi, M.B.; Wang, X. Structure Level Pruning of Efficient Convolutional Neural Networks with Sparse Group LASSO. *Int. J. Mach. Learn. Comput.* **2022**, *12*, 16091–16101.
28. Fang, G.; Ma, X.; Song, M.; Mi, M.B.; Wang, X. Depgraph: Towards any structural pruning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 16091–16101.
29. Wu, C.; Pang, W.; Liu, H.; Lu, S. Group pruning with group sparse regularization for deep neural network compression. In Proceedings of the 2019 IEEE 4th International Conference on Signal and Image Processing (ICSIP), Wuxi, China, 19–21 July 2019; pp. 325–329.
30. Li, X.; Chen, L.; Gao, Z.; Zhang, X.; Wang, C.; Chen, H. Lasso regression based channel pruning for efficient object detection model. In Proceedings of the 2019 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Jeju, Republic of Korea, 5–7 June 2019; pp. 16–29.
31. Oyedotun, O.; Aouada, D.; Ottersten, B. Structured compression of deep neural networks with debiased elastic group lasso. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass, CO, USA, 1–5 March 2020; pp. 2277–2286.
32. Xie, Z.; Li, P.; Li, F.; Guo, C. Pruning filters base on extending filter group lasso. *IEEE Access* **2020**, *8*, 217867–217876. [[CrossRef](#)]
33. Molchanov, P.; Mallya, A.; Tyree, S.; Frosio, I.; Kautz, J. Importance estimation for neural network pruning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 11264–11272.
34. Lee, J.; Park, S.; Mo, S.; Ahn, S.; Shin, J. Layer-adaptive sparsity for the magnitude-based pruning. *arXiv* **2020**, arXiv:2010.07611.
35. Friedman, J.; Hastie, T.; Tibshirani, R. A note on the group lasso and a sparse group lasso. *arXiv* **2010**, arXiv:1001.0736.
36. Nogueira, P.V.; Coutinho, G.; Pio, R.; da Silva, D.F.; Zambon, C.R. Establishment of growth medium and quantification of pollen grains and germination of pear tree cultivars. *Rev. Ciênc. Agron.* **2016**, *47*, 380–386. [[CrossRef](#)]
37. Quinet, M.; Jacquemart, A.L. Troubles in pear pollination: Effects of collection and storage method on pollen viability and fruit production. *Acta Oecol.* **2020**, *105*, 103558. [[CrossRef](#)]
38. Yu, L.; Xiong, J.; Fang, X.; Yang, Z.; Chen, Y.; Lin, X.; Chen, S. A litchi fruit recognition method in a natural environment using RGB-D images. *Biosyst. Eng.* **2021**, *204*, 50–63. [[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.