agronomy

MDPI

Article

Comparative Analysis of Base-Width-Based Annotation Box Ratios for Vine Trunk and Support Post Detection Performance in Agricultural Autonomous Navigation Environments

Hong-Kun Lyu [1,*], Sanghun Yun [1] and Seung Park [2]

[1] Division of ABB, ICT Research Institute, Daegu Gyeongbuk Institute of Science and Technology (DGIST), 333, Techno Jungang Daero, Hyeonpung-Eup, Dalseong_Gun, Daegu 42988, Republic of Korea; shyun@dgist.ac.kr

[2] AMR LABS Inc., Hyundai Silicon Alley Dongtan(B-1535), 150, Dongtanyeongcheon-ro, Hwaseong-si 18462, Republic of Korea; dean.park@amrlabs.co.kr

* Correspondence: hklyu@dgist.ac.kr

Abstract

AI-driven agricultural automation increasingly demands efficient data generation methods for training deep learning models in autonomous robotic systems. Traditional bounding box annotation methods for agricultural objects present significant challenges including subjective boundary determination, inconsistent labeling across annotators, and physical strain from extensive mouse movements required for elongated objects. This study proposes a novel base-width standardized annotation method that utilizes the base width of a vine trunk and a support post as a reference parameter for automated bounding box generation. The method requires annotators to specify only the left and right endpoints of object bases, from which the system automatically generates standardized bounding boxes with predefined aspect ratios. Performance assessment utilized Precision, Recall, F1-score, and Average Precision metrics across vine trunks and support posts. The study reveals that vertically elongated rectangular bounding boxes outperform square configurations for agricultural object detection. The proposed method is expected to reduce time consumption from subjective boundary determination and minimize physical strain during bounding box annotation for AI-based autonomous navigation models in agricultural environments. This will ultimately enhance dataset consistency and improve the efficiency of artificial intelligence learning.

Keywords: agricultural robotics; deep learning; object detection; bounding box annotation; vine trunk detection; autonomous navigation; vineyard automation

check for updates

Academic Editor: Xiongkui He

Received: 25 July 2025
Revised: 25 August 2025
Accepted: 27 August 2025
Published: 31 August 2025

Citation: Lyu, H.-K.; Yun, S.; Park, S. Comparative Analysis of Base-Width-Based Annotation Box Ratios for Vine Trunk and Support Post Detection Performance in Agricultural Autonomous Navigation Environments. *Agronomy* **2025**, *15*, 2107. https://doi.org/10.3390/agronomy15092107

1. Introduction

AI-driven agricultural automation has become increasingly important for addressing global food security issues and agricultural labor shortages. Developing autonomous agricultural robots requires advanced technologies similar to those applied in autonomous vehicle systems. However, introducing these technologies to agricultural environments faces significant challenges due to the difficulty of collecting vast amounts of big data across diverse agricultural conditions and the enormous investment required for technology development [1,2].

Agricultural autonomous navigation has evolved through several technological approaches to address the unique challenges of unstructured agricultural environments.

Computer vision techniques such as edge detection, Hough transform, and color-based segmentation have been utilized for crop row detection and traversable space identification [3,4]. While these traditional vision methods were effective in controlled environments, they were vulnerable to lighting changes, shadows, and irregular crop patterns. To address the limitations of rule-based vision, machine learning techniques including SVM, decision trees, and clustering algorithms were introduced for agricultural feature classification and free space detection [5,6]. Although robustness improved in vineyards and orchards, these approaches required extensive feature engineering and manual feature selection, making expansion to diverse agricultural environments challenging.

Recently, deep learning techniques using CNN and U-Net semantic segmentation models have achieved success in crop row detection and path planning, while deep reinforcement learning has enabled position-agnostic traversal in vineyards, and fully convolutional networks have significantly improved free space detection accuracy [7–10]. Research on autonomous navigation in vineyard and orchard environments has also progressed in the direction of utilizing YOLO algorithms. A method has been proposed to improve automatic driving accuracy by combining machine vision and YOLOv4 model to detect the relative position between orchard robots and orchard rows in the special environmental conditions of orchards [11]. Meanwhile, an "Improved Hybrid Model" was proposed by combining YOLOv7 and Robot Operating System (ROS) to enhance autonomous driving and obstacle avoidance accuracy. In particular, the combination of YOLOv7 and RRT (Rapidly-exploring Random Tree) algorithms improved navigation performance in complex orchard environments [12]. These studies have demonstrated that YOLO-series algorithms can be effectively utilized for the development of autonomous navigation systems in agricultural environments. However, these studies still have not solved the challenges related to efficient annotation data generation methods. Large-scale annotated datasets are essential, resulting in significant time and cost burdens for data collection and annotation. Despite technological advances, particularly the annotation stage for deep learning training datasets remains a bottleneck, and annotation work in agricultural environments presents unique challenges different from general computer vision applications.

Computer vision is a core technology enabling autonomous operation of agricultural robots. Particularly, the ability to identify free space for safe traversal is essential. Recent studies in computer vision have shown that consistent bounding box aspect ratios improve the stability of convolutional neural network training and enhance the feature extraction process [13,14]. Standardization of annotation methods has been shown to reduce inter-annotator variability and improve overall dataset quality [15]. Additionally, ergonomic research has demonstrated that increased cognitive load adversely affects work efficiency and consistency, suggesting that simplification of decision-making processes in repetitive tasks can contribute to improved work performance [16]. The performance of object detection algorithms is heavily dependent on the quality and consistency of training data, particularly the accuracy of annotation methods used in dataset generation [17,18].

In vineyards, autonomous mobile robots must move through the traversable space between vine rows while avoiding vine trunks and support posts. As illustrated in Figure 1, semi-structured vineyard environments present typical scenarios where robots must navigate between regularly spaced vine rows, whether in greenhouse vineyard facilities or open-field installations. Trunk positions serve as important reference points for determining the free space that robots can traverse through. The vineyard layout shown in Figure 1 demonstrates the characteristic corridor-like pathways formed between vine rows, where accurate detection of vine trunk and support posts is crucial for safe autonomous traversal. Conventional annotation methods typically involve manually enclosing the entire trunk from base to upper branches with bounding boxes, as exemplified in the annotation samples presented in Figure 1 [19]. These traditional bounding box annotations encompass the complete vertical extent of vine trunks, requiring annotators to determine both the upper and lower boundaries of elongated agricultural objects in both controlled greenhouse environments and variable outdoor field conditions.

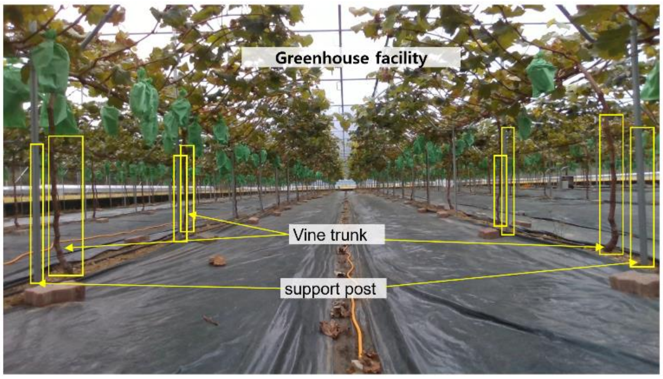

Figure 1. Representative training dataset images and bounding box annotation examples from semi-structured vineyard environments for autonomous agricultural robot development, greenhouse vineyard cultivation facility.

Agricultural objects present more ambiguous boundaries than road traffic objects, leading to several problems [20]. First, annotators must continuously determine the bottom width, top width, and height of bounding boxes, with inefficiency increasing as more data is required, and different standards among annotators reduce dataset consistency, leading to degraded deep learning performance [21,22]. Second, continuous annotation work leads to accumulated fatigue and increased processing time, while vertically elongated trunks have ambiguous top and bottom boundary determination, and large vertical mouse movements cause strain on wrists and shoulders [23–25]. Third, while trunk base position is more important than overall shape for robot traversal, conventional annotation includes unnecessary information beyond requirements, causing noise [26].

To address these limitations, this study proposes a base-width-based annotation method that standardizes bounding box aspect ratios based on trunk base width. Rather than allowing annotators to arbitrarily determine box sizes, once the base width is determined, fixed width:height ratios are automatically applied. This is based on the insight that trunk base position is more important for robot traversal. The primary objective of this study is to systematically evaluate eight aspect ratio combinations to find the optimal annotation configuration, combining width multipliers (a = 1.0, 1.5, 2.0, 2.5) and height multipliers (b = 1 × a, 2 × a). The configurations T1010, T1515, T2020, T2525, T1020, T1530, T2040, T2550 were evaluated to find the optimal balance between contextual information content and computational efficiency. Each configuration was evaluated using Precision, Recall, F1-score, and Average Precision (AP).

This study presents the following differentiated contributions in the field of dataset construction for autonomous agricultural robots. First, unlike existing annotation methods that encompass the entire trunk area, this study proposes for the first time a standardized bounding box generation method based solely on the base width of vine trunks and support posts, designed to minimize subjective judgment by annotators and reduce mouse pointer movement distance required for generating bounding box boundaries. Second, through systematic evaluation of eight aspect ratio combinations (T1010~T2550) for agricultural environment object detection, this study confirms that vertically elongated rectangular bounding boxes demonstrate relatively superior performance compared to square configurations for elongated agricultural objects. Third, computer program experiments validated that the proposed base-width-based annotation method shows applicability for generating traversal paths required for autonomous navigation of actual agricultural robots. These research achievements can contribute substantially to the advancement of AI-driven automation technology in the agricultural robotics field.

2. Materials and Methods

2.1. Greenhouse Vineyard Environment

The experiment was conducted in a semi-structured greenhouse vineyard environment located in Sangju-si, Gyeongsangbuk-do, Republic of Korea, in September 2020, to enable controlled data collection. We used an image data collection cart system equipped with an Intel® RealSense™ Depth Camera D435i mounted on a commercial 1/5 scale radio-controlled cart to collect driving path images of the greenhouse vineyard. For this study, we stored RGB images while driving the radio-controlled cart. The images used for this experiment were acquired through the following process. The image collection cart system was controlled at 3.6 Km/h, and camera images were collected at 30 fps. Among the collected consecutive images, every 6th image was selected for the experiment, resulting in an effective image collection rate of 5 fps, and the experiment was designed so that the distance difference between images would be approximately 20 cm. Greenhouse vineyards provide advantages for agricultural robot research including stable power supply, controlled environmental conditions, and ease of equipment installation for repetitive development testing.

As shown in Figures 1 and 2, productivity-oriented vineyards typically consist of grapevines planted in regular rows at regular intervals and support posts installed at consistent spacing. These support structures are connected by steel wires to which the grapevines are attached, forming what is known as a "tree wall" or "fruit wall". The space between vine rows creates corridors that serve as pathways for both human workers and autonomous operation vehicles. This structured arrangement defines a semi-structured agricultural environment suitable for robotic applications [19].

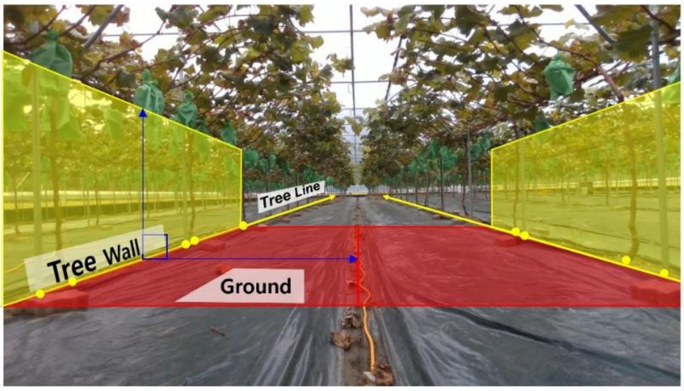

Figure 2. Semi-structured vineyards designed for productivity feature grapevines and support posts positioned at consistent intervals, creating a distinctive "tree wall" or "fruit wall" formation.

The experimental vineyard, as shown in Figures 1 and 2, features grapevines planted at regular intervals with support posts arranged to maintain consistent row spacing. The bases of vine trunks and support posts serve as important reference points for determining traversable free space, as they represent the primary obstacles that autonomous robots must avoid while moving through vineyard environments.

2.2. Annotation Methodology

Traditional bounding box annotation requires annotators to manually determine the top-left and bottom-right corners of each object. Agricultural object recognition targets for autonomous operation differ from road autonomous driving recognition targets in that agricultural objects have ambiguous boundary characteristics unlike the clear boundary features of road objects. Therefore, Agricultural environments present unique challenges for AI training data generation compared to conventional computer vision applications. Plant-based objects in agricultural settings exhibit ambiguous boundaries and irregular forms, making bounding box annotation significantly more time-intensive than annotation work for artificial objects with clear geometric boundaries. This increased complexity leads to greater inter-annotator variability and reduced dataset consistency, which directly impacts the quality of training data for agricultural AI systems. Annotation work for vertically elongated objects such as vine trunks and support posts become particularly difficult due to two main problems. First, accurately determining the boundaries of the upper and lower portions of trunks is subjective and ambiguous. Second, creating vertically elongated bounding boxes requires large vertical mouse movements, causing physical strain on the annotator's wrists and shoulders.

To address these limitations, we propose a novel two-point annotation method focusing on the lower portion of vine trunks and support posts. This method was developed based on the insight that the position of the trunk and post base is more important than the overall object shape for autonomous operation movement.

As shown in Figure 3, the existing annotation method (a) requires annotators to set diagonal boundaries from the top-left to bottom-right that encompass the entire trunk, necessitating large mouse movements and subjective boundary decisions. In contrast, the proposed method (b) requires only specifying a diagonal line from the left endpoint of the vine trunk base or post base as the starting point to an arbitrary point on the y-axis while maintaining the x-axis position of the right endpoint in the x-y coordinate system, significantly improving annotation work efficiency and consistency through reduced mouse movement distance and clear boundary judgment.

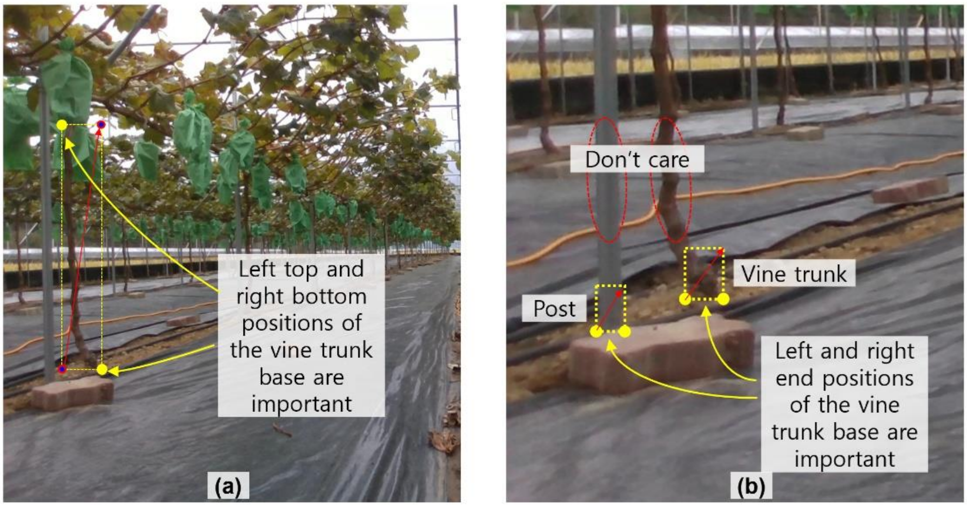
Left top and right bottom positions of the vine trunk base are important
(a)
Don't care
Vine trunk
Post
Left and right end positions of the vine trunk base are important
(b)

Figure 3. Annotation method comparison: (**a**) traditional full-trunk bounding box annotation with extensive mouse movements, and (**b**) proposed base-width-based method focusing on vine trunk base endpoints for improved efficiency and consistency.

The Base-Width-Based Annotation proposed in our research consists of two phases: Pre-annotation Phase and Standardized Bounding Box Generation Phase.

The Pre-annotation Phase represents the initial stage of the proposed base-width-based annotation methodology for agricultural object detection. Figure 4 shows the workflow of the Pre-annotation Phase. The steps of the Detailed Pre-annotation Phase Process are as follows.

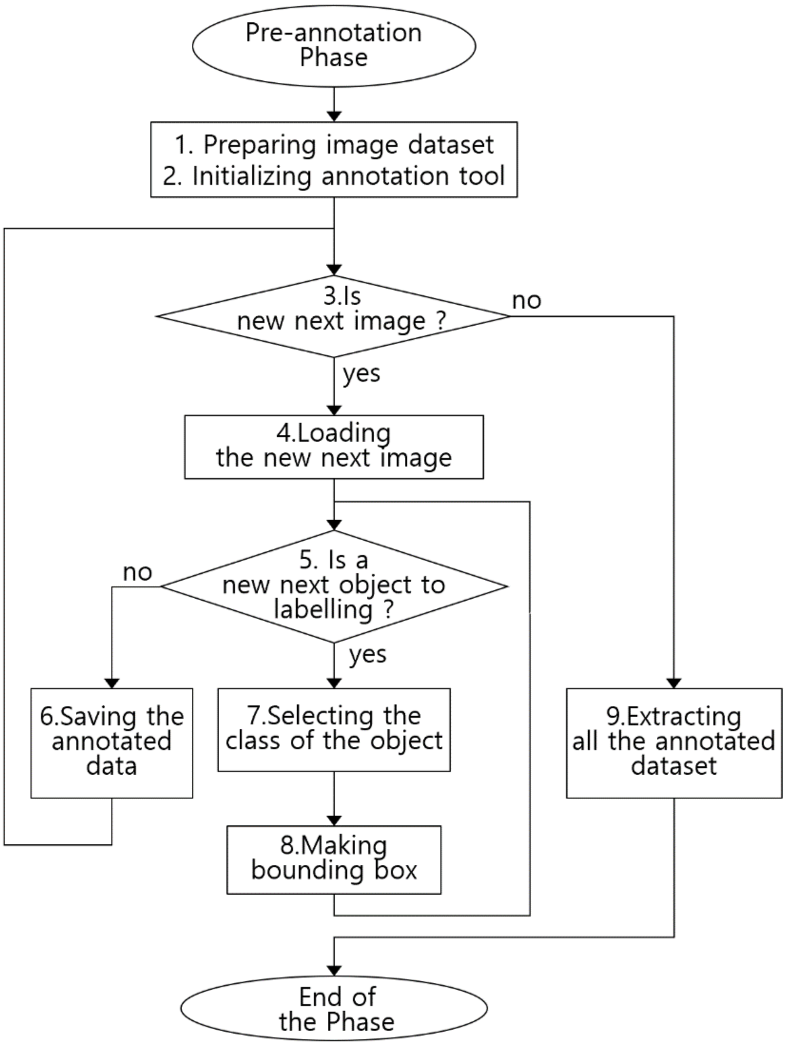
Pre-annotation Phase
1. Preparing image dataset
2. Initializing annotation tool
3.Is new next image ?
no
yes
4.Loading the new next image
5. Is a new next object to labelling ?
no
yes
6.Saving the annotated data
7.Selecting the class of the object
8.Making bounding box
9.Extracting all the annotated dataset
End of the Phase

Figure 4. The workflow of the Pre-annotation Phase.

Step 1. Image Dataset Preparation: In this initial step, the image dataset for AI training is prepared and organized. This involves collecting and arranging vineyard environment images containing vine trunks and support posts into appropriate folders or datasets. This preparatory step requires all images to be carefully reviewed and prepared for the annotation process.

Step 2. Initializing Annotation Tool: This step involves selecting and launching an appropriate annotation program or tool that enables bounding box creation. The annotation tool is not restricted to any specific software; any tool supporting bounding box annotation can be utilized. During this stage, the image dataset prepared in the first step is loaded into the annotation environment.

Step 3. Verification of New Next Image: The system determines whether there are images requiring annotation in the loaded dataset. If unprocessed images exist, the workflow proceeds to the fourth step; otherwise, it moves directly to the ninth step for data extraction.

Step 4. Loading the New Next Image: This step involves loading a new image from the dataset into the annotation tool's workspace according to a predetermined sequence. The image is displayed in the annotation workspace for analysis and processing.

Step 5. Verification of New Next Object for Labeling: The annotator examines the loaded image to determine if there are vine trunks or support posts that have not yet been annotated with bounding boxes. This judgment process creates a critical bifurcation in the workflow. If all objects have been annotated, the process proceeds to the sixth step for data saving. If objects requiring bounding box creation are identified, the process advances to the seventh step.

Step 6. Saving Annotation Data: When all objects in the current image have been annotated, or when no objects requiring annotation are present, the system saves the annotation data for the current image. After saving, the workflow returns to the third step to check for additional images requiring annotation.

Step 7. Selection of Object Class: For each identified object requiring annotation, the annotator selects and assigns the appropriate class label—either vine trunk (Class 1) or support post (Class 0). This classification is essential for subsequent specialized detection model training and must be accurately selected.

Step 8. Bounding Box Finalization: This step applies the key methodology in our approach. Unlike conventional annotation methods that require comprehensive bounding of the entire object, our approach focuses exclusively on the base endpoints of vine trunks and support posts. The annotator selects the left endpoint of the object base as the starting point and drags to an appropriate position on the y-axis while maintaining the x-axis value of the right endpoint, ensuring that the left and right positions of the object base are accurately marked. During this process, it is only necessary for both left and right endpoints of the object base to be included within the bottom boundary of the preliminary bounding box, without needing to encompass the entire trunk structure. The height of the bounding box created at this time is not important, and only a minimal height sufficient for visual identification of the bounding box is required during the verification stage. This bounding box finalization approach minimizes subjective judgment about object boundaries inherent in traditional annotation methods. Once the bounding box finalization for one object is complete, the workflow returns to the fifth step to check for additional objects requiring annotation.

Step 9. Extraction of Completed Annotation Dataset: When the bounding box finalization process for all prepared images is completed, the workflow concludes with the extraction of all annotated bounding box data. This extracted Pre-annotation dataset serves as input for the Standardized Bounding Box Generation Phase, where the system will algorithmically transform these base endpoint parameters into standardized bounding boxes with predetermined aspect ratios.

The Pre-annotation Phase concludes after all images have been processed and the complete annotation dataset has been extracted. This systematic approach to annotation represents a fundamental departure from conventional methods, particularly eliminating ambiguous boundary determination and removing the need to consider the vertical length and shape variations of the upper portions of agricultural objects requiring annotation.

After completing the extraction of the pre-annotation dataset in the Pre-annotation Phase, the second phase of Standardized Bounding Box Generation is performed. In this phase, the annotator's only task is to determine the aspect ratio of the bounding boxes to be generated. By applying the pseudocode algorithm in Algorithm 1, it receives the data prepared in the Pre-annotation Phase as input, processes it, and automatically generates the final annotation dataset. The steps of the standardized bounding box generation phase process are as follows.

Step 1. Load Pre-annotation Data: This initial step involves loading the dataset extracted from the Pre-annotation Phase, which contains the base endpoint coordinates for all annotated objects across all images. These coordinates serve as the foundation for generating standardized bounding boxes.

Step 2. Calculate Base Width: For each annotated object, the system calculates the reference width (1W) by measuring the distance between the left and right endpoints of the object base obtained during the Pre-annotation Phase. This reference width serves as the fundamental parameter for determining the dimensions of the standardized bounding box.

Step 3. Apply Predefined Aspect Ratios: The system applies predefined aspect ratios (aW × bW) to the reference width, where 'a' represents the width multiplier and 'b' represents the height multiplier. This step transforms the simple base width information into properly dimensioned bounding boxes with standardized proportions.

Step 4. Position Bounding Boxes: The system positions each standardized bounding box so that it is centered on the object base. This ensures consistent placement relative to the actual physical position of vine trunks and support posts, which is critical for autonomous navigation applications.

Step 5. Convert to Standard Annotation Format: The generated standardized bounding boxes are converted to a standard annotation format compatible with object detection model training frameworks. This typically involves normalizing the coordinates and dimensions according to the requirements of the selected training framework.

Step 6. Verify Generated Bounding Boxes: The system performs verification checks to ensure all standardized bounding boxes have been properly generated and positioned. This may include visual verification through sample renderings or automated validation of bounding box parameters.

Step 7. Generate Final Annotation Dataset: The system compiles all standardized bounding box data into a comprehensive annotation dataset that can be directly used for training object detection models. This dataset includes class information, standardized bounding box coordinates, and any additional metadata required for training.

The Standardized Bounding Box Generation Phase concludes with the generation of the standardized annotation dataset according to the process described above.

Algorithm 1. StandardizedBoundingBoxGeneration

```
Require: pre_annotation_dataset    // Dataset extracted from Pre-annotation Phase
Require: aspect_ratios             // Predefined list of aspect ratios ($a$W × $b$W)

function StandardizedBoundingBoxGen (pre_annotation_dataset, aspect_ratios)
object_base_endpoints ← LoadPreAnnotationData(pre_annotation_dataset)
objects_with_width ← CalculateBaseWidth(object_base_endpoints)
standardized_objects ← ApplyAspectRatios(objects_with_width, aspect_ratios)
positioned_boxes ← PositionBoundingBoxes(standardized_objects)
standard_format_annotations ← ConvertToStandardFormat(positioned_boxes)
verified_annotations ← VerifyBoundingBoxes(standard_format_annotations)
final_annotation_dataset ← GenerateFinalDataset(verified_annotations)
return final_annotation_dataset
end function
```

As shown in Figure 5, automatic bounding box generation is based on the following definitions: 1W (width) represents the distance between the left and right points of the trunk base or support post base; the trunk center point is the center point of the 1W horizontal line division; the bottom horizontal line is the horizontal line extending from the 1W division; and the bounding box baseline is the horizontal line located 0.3 × W below the bottom horizontal line.

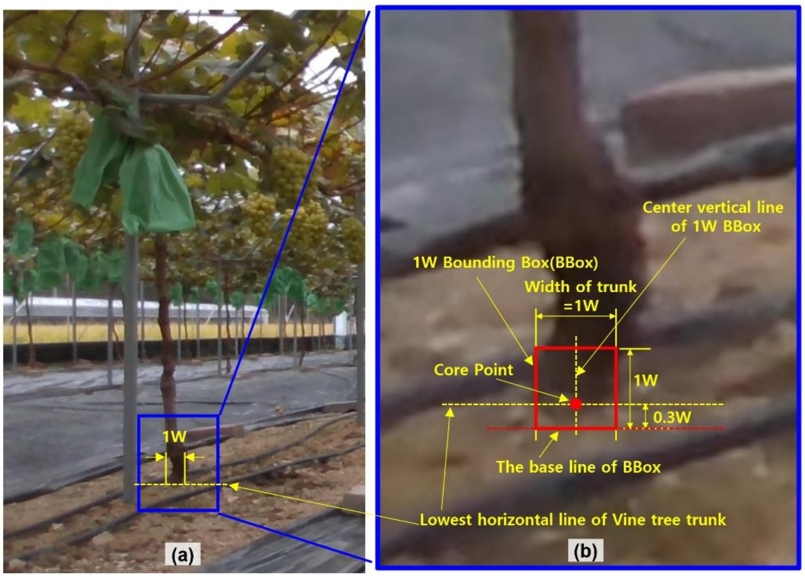

Figure 5. Base-width standardization parameters: (**a**) 1W (vine trunk base width), (**b**) Core Point (center of 1W), and baseline positioning (0.3W below Core Point) for symmetric bounding box generation. '1W' represents the reference width, defined as the distance between the left and right endpoints of the vine trunk or support post base.

All bounding boxes are defined by width (*a*W) and height (*b*W), where the area of the bounding box equals *a*W × *b*W. In this experiment, a = 1.0, 1.5, 2.0, 2.5 and $b = 1 \times a$, $2 \times a$. This creates eight different combinations of bounding boxes according to the rules shown in Figure 6, summarized in Table 1.

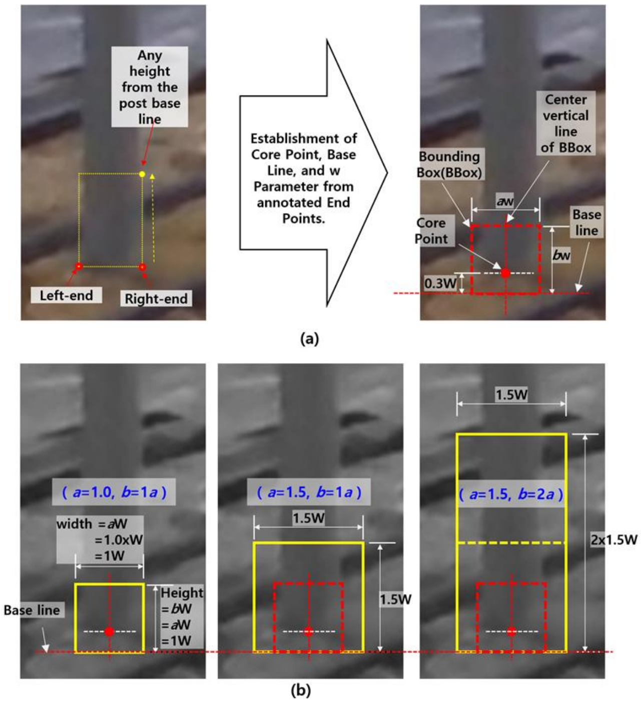

Figure 6. Definition of Bounding boxes of all rectangular shapes can be expressed in width *a*W and height. (**a**) Automatic parameter calculation process from manual endpoint annotation, (**b**) generation principles for eight standardized bounding box combinations. '*a*' represents the width multiplier that determines the horizontal dimension of the bounding box as a multiple of the base width (W) and '*b*' represents the height multiplier that determines the vertical dimension of the bounding box as a multiple of the base width (W).

Table 1. Bounding box aspect ratio configurations with width and height specifications.

Type [1]	Width	Height
T1010	1.0 × W	1.0 × W
T1020	1.0 × W	2.0 × W
T1515	1.5 × W	1.5 × W
T1530	1.5 × W	3.0 × W
T2020	2.0 × W	2.0 × W
T2040	2.0 × W	4.0 × W
T2525	2.5 × W	2.5 × W
T2550	2.5 × W	5.0 × W

[1] Type: Bounding box configuration identifier, where the alphanumeric code indicates width and height multipliers relative to the vine trunk base width.

2.3. Deep Learning Model Configuration

This study employed YOLOv3 (You Only Look Once version 3) with Darknet-53 as the backbone network. The model configuration was optimized for vineyard environment detection with the following specifications: input resolution of 416 × 416 pixels, batch size of 64 (testing) and 16 (subdivision), learning rate of 1×10^{-3} with a burn-in period of 1000 iterations, maximum batches of 12,000, momentum of 0.9, and decay of 5×10^{-4} [27].

Transfer learning was implemented using pre-trained darknet53.conv.74 model weights. The model was configured for 2-class detection as shown in Figure 3b: Class 0 for support posts/stakes and Class 1 for vine trunks. Data augmentation techniques included saturation of 1.5, exposure of 1.5, hue of 0.1, and angle of 0 (rotation).

2.4. Dataset Preparation

A total of 962 images were collected from the greenhouse vineyard environment. The dataset was divided as follows: training set with 770 images (80%), validation set with 86 images (9%) and test set with 106 images (11%).

Separate annotation datasets were generated for each of the eight bounding box configurations (T1010 through T2550) using the proposed base-width-based annotation method. Each dataset contained the same object locations but with different bounding box dimensions according to their respective aspect ratios. The annotation data was converted to YOLO format, with each bounding box defined by center coordinates (x, y) and dimensions (width, height) normalized by image dimensions.

2.5. Evaluation Metrics

Model performance was evaluated using standard object detection metrics: Precision calculated as TP/(TP + FP), Recall calculated as TP/(TP + FN), F1-score calculated as 2 × (Precision × Recall)/(Precision + Recall), Average Precision (AP) as the area under the precision-recall curve, and mean Average Precision (mAP) as the average of AP across all classes. Here, TP, FP, and FN represent true positives, false positives, and false negatives, respectively, calculated at IoU threshold 0.5 [28].

2.6. Experimental Procedure

Training was conducted using the Darknet framework with the following procedure. Data preprocessing involved resizing images to 416 × 416 pixels and normalization. Model training was performed separately for each of the eight annotation configurations. Validation involved evaluating model performance on the held-out validation set. Comparative analysis compared results across all eight configurations to identify the optimal aspect ratio. The training process was performed on a system equipped with GPU acceleration to ensure efficient model convergence within the specified 12,000 iterations.

3. Results

3.1. Performance Comparison Based on Bounding Box Configuration

Performance evaluation of eight different bounding box configurations (T1010 through T2550) showed significant differences in detection accuracy for both vine trunk and support post classes. The results demonstrate that the aspect ratio of annotation boxes significantly affects model performance, with specific configurations showing superior detection capabilities.

The performance metrics for support post detection across all eight configurations are presented in Figure 7a and summarized in Table 2. Results show a clear trend that larger bounding box configurations generally achieved better performance than smaller ones. Configuration T2020 (2.0W × 2.0W) achieved the highest performance for support post detection with accuracy 0.69, precision 0.60, recall 0.62, F1-score 0.61, and AP 0.47. This was closely followed by T2550 (2.5W × 5.0W), which showed similar accuracy (0.69) but slightly lower precision (0.59) and recall (0.60).

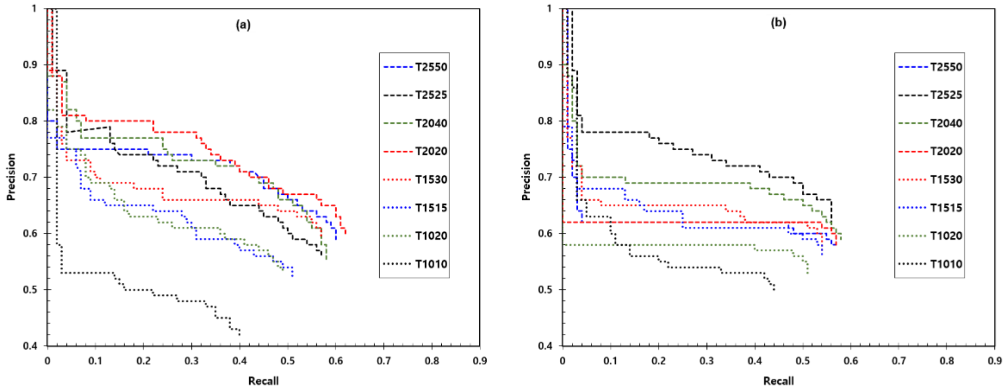

Figure 7. Interpolated Average Precision (AP) performance comparison across eight bounding box configurations for two object classes. (**a**) Interpolated Average Precision of Class 0 (support posts), (**b**) Interpolated Average Precision of Class 1 (vine trunks). In the notation Txxxx, 'T' represents Type, the first two digits indicate the width multiplier (**a**), and the last two digits indicate the height multiplier (**b**). For example, T1010 represents a 1.0 × W width by 1.0 × W height configuration, while T2550 represents a 2.5 × W width by 5.0 × W height configuration.

The smallest configuration, T1010 (1.0W × 1.0W), showed the lowest performance with accuracy 0.53, precision 0.42, recall 0.40, F1-score 0.41, and AP 0.20. This indicates that excessively small bounding boxes may fail to capture sufficient contextual information for effective support post detection.

For vine trunk detection, as shown in Figure 7b and Table 2, the performance pattern differed from support post detection. Configuration T2525 (2.5W × 2.5W) achieved the best performance with precision 0.70, recall 0.61, accuracy 0.56, F1-score 0.58, and AP 0.42.

Comparing support post detection performance with vine trunk detection performance, the optimal bounding box ratio type for detecting vine trunks differed from that for support posts. Unlike the best configuration T2020 for support post detection, vine trunk detection performance evaluation showed that T2525 configuration achieved the best AP performance of 0.42, followed by T2040 configuration with good performance at AP 0.40. The smallest bounding box corresponding to T1010 bounding box ratio type showed the lowest performance in vine trunk detection with precision 0.62, recall 0.50, accuracy 0.44, F1-score 0.47, and AP 0.26.

Table 2. Detailed performance metrics for eight bounding box configurations showing Accuracy, Precision, Recall, F1-score, and Average Precision (AP) values for Class 0 (support posts) and Class 1 (vine trunks) corresponding to Figure 7 results.

Class	Type [1]	Accuracy	Precision	Recall	F1 Score	AP
Posts #0	T1010	0.53	0.42	0.40	0.41	0.20
	T1020	0.62	0.53	0.49	0.51	0.32
	T1515	0.61	0.52	0.51	0.52	0.33
	T1530	0.67	0.59	0.57	0.58	0.39
	T2020	0.69	0.60	0.62	0.61	0.47
	T2040	0.67	0.55	0.58	0.57	0.43
	T2525	0.65	0.56	0.57	0.57	0.40
	T2550	0.69	0.59	0.60	0.60	0.43
Vine trunks #1	T1010	0.62	0.50	0.44	0.47	0.26
	T1020	0.64	0.53	0.51	0.52	0.30
	T1515	0.66	0.56	0.54	0.55	0.35
	T1530	0.67	0.57	0.54	0.55	0.36
	T2020	0.69	0.58	0.57	0.58	0.36
	T2040	0.71	0.59	0.58	0.58	0.40
	T2525	0.70	0.61	0.56	0.58	0.42
	T2550	0.70	0.58	0.57	0.57	0.36

[1] Type: Bounding box configuration identifier, where the alphanumeric code indicates width and height multipliers relative to the vine trunk base width.

The mean average precision (mAP) results combining both classes are plotted in Figure 8 and presented in Table 3. The mAP analysis provides a comprehensive view of overall model performance across various bounding box configurations. Configuration T2040 (2.0W × 4.0W) achieved the highest mAP of 41.36% with individual AP values of 39.81% for vine trunks and 42.91% for support posts. This was followed by T2525 (2.5W × 2.5W) with mAP 41.19% and T2020 (2.0W × 2.0W) with mAP 41.08%.

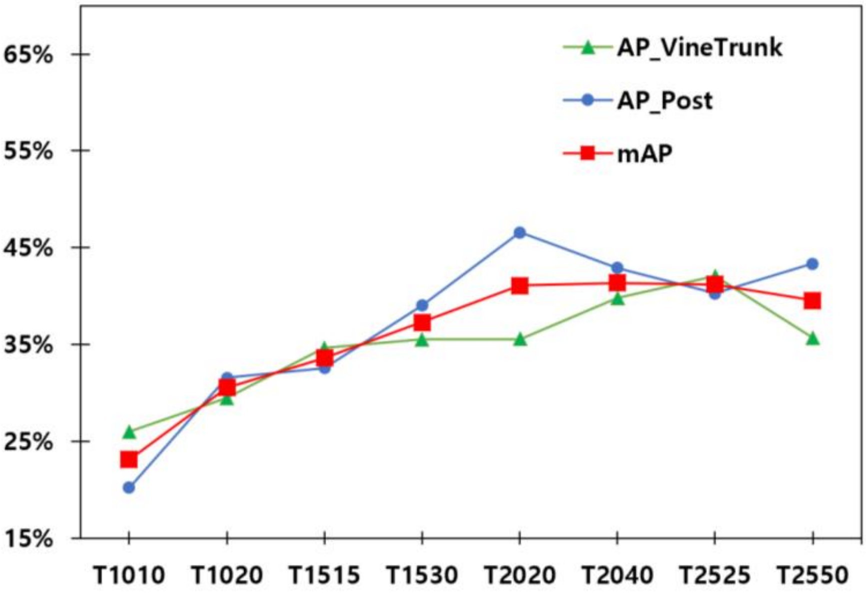

Figure 8. Mean Average Precision (mAP) comparison across eight bounding box configurations, consolidating Class 0 (support posts) and Class 1 (vine trunks) results. AP_VineTrunk: Average Precision for vine trunk detection; AP_Post: Average Precision for support post detection; mAP: mean Average Precision, representing the average of AP values across both classes.

Table 3. Mean Average Precision (mAP) and individual class Average Precision (AP) values for eight bounding box configurations corresponding.

Type [1]	AP_VineTrunk	AP_Post	mAP
T1010	26.01%	20.21%	23.11%
T1020	29.51%	31.57%	30.54%
T1515	34.61%	32.57%	33.59%
T1530	35.52%	39.03%	37.28%
T2020	35.57%	46.59%	41.08%
T2040	39.81%	42.91%	41.36%
T2525	42.08%	40.29%	41.19%
T2550	35.72%	43.36%	39.54%

[1] Type: Bounding box configuration identifier, where the alphanumeric code indicates width and height multipliers relative to the vine trunk base width.

3.2. Optimal Configuration Analysis

Analysis of results across all eight configurations reveals several important patterns. Configuration size impact shows that small configurations (T10xx and T15xx) consistently underperformed, suggesting that excessively small bounding boxes fail to capture sufficient contextual information for effective detection. Aspect ratio importance demonstrates that rectangular bounding boxes with twice the height of the width (T1530, T2040) generally showed better performance than square configurations, particularly for elongated objects such as vine trunks. Class-specific optimization reveals that different classes showed preferences for different optimal configurations: support posts achieved best performance at T2020 (2.0W × 2.0W), vine trunks at T2525 (2.5W × 2.5W), and overall (mAP) at T2040 (2.0W × 4.0W).

The consistent underperformance of smaller configurations indicates that insufficient contextual information is captured when bounding boxes are too small. This finding aligns with computer vision literature indicating that adequate context around target objects is crucial for effective feature learning in convolutional neural networks. The superior performance of larger configurations suggests that including more contextual information around vine trunk bases improves detection accuracy. However, performance plateauing and slight decrease in the largest configuration (T2550) suggests there is an optimal size beyond which additional context may introduce more noise than useful information.

The analysis shows that vertically elongated rectangular bounding boxes generally outperform square configurations. This finding is particularly relevant for elongated objects such as vine trunks and support posts, where natural object geometry favors vertical rectangles. Configuration T2040 (2.0W × 4.0W) achieved the highest mAP of 41.36%, demonstrating that vertically elongated bounding boxes effectively capture essential features of both vine trunks and support posts. This aspect ratio appears to provide optimal balance between capturing sufficient trunk information and maintaining computational efficiency.

3.3. Practical Implementation Results

To validate the practical feasibility of models trained with the proposed Base-Width-Based Annotation method for real agricultural robot applications, detection experiments were conducted using collected images as input data to detect vine trunk bases and support posts. The experimental results applying the T2040 trained model are presented in Figure 9. The specifications of the computer used for detection experiments are as follows: Intel(R) Core(TM) i9-12900HX CPU @2.30 GHz, RAM 32 GB, Windows 11 Pro, 64-bit operating system, NVIDIA GeForce RTX 2080. Figure 9 shows successful detection test results of vine trunk bases and support post lower portions, which showed potential for connecting to form navigation paths for autonomous agricultural operation vehicles. The detected lower points of trunks and support posts can be processed to generate left and right regression lines connecting detected points, vanishing point calculation for perspective correction, navigation path determination by comparing camera optical axis with detected center aisle line, and steering angle calculation for autonomous vehicle guidance.

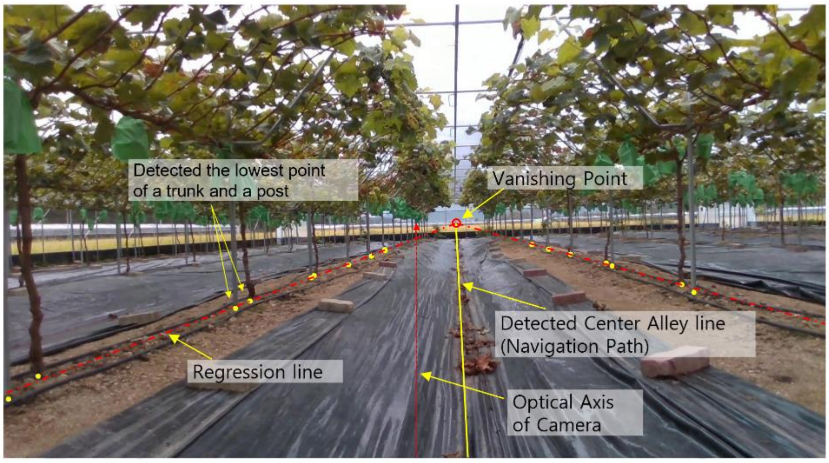
Detected the lowest point of a trunk and a post
Vanishing Point
Detected Center Alley line (Navigation Path)
Regression line
Optical Axis of Camera

Figure 9. Detection test results showing successful identification of vine trunk bases and support posts using the proposed base-width-based annotation method. The detected lower portions provide reference points for generating navigation paths and steering guidance for autonomous agricultural operation vehicles in vineyard environments.

The detected trunk and support post base points can be processed to generate several useful outputs for autonomous navigation. Left and right regression lines can be created by connecting the detected points, vanishing points can be calculated for perspective correction, navigation paths can be determined by comparing the camera's optical axis with the detected central aisle line, and steering angles can be calculated for autonomous operation vehicle guidance. These results confirm that the proposed annotation method can be used to generate training data capable of supporting real agricultural robot applications.

The successful detection of vine trunk bases and support posts demonstrates that focusing on the traversal-relevant portion of objects rather than complete visible structures aligns with the specific requirements of agricultural robots. The detection information generated by the proposed method can be readily integrated with existing agricultural robot control systems, providing computational efficiency that can meet real-time processing requirements.

4. Discussion

4.1. Optimal Bounding Box Configuration Analysis

The comprehensive evaluation of eight different aspect ratios provides important insights into optimal annotation strategies for agricultural object detection. The results demonstrate that bounding box size and aspect ratio significantly affect model performance, with different configurations showing preferences for different object classes. The consistent underperformance of small configurations (T10xx and T15xx) suggests that insufficient contextual information is captured when bounding boxes are too small. This finding aligns with computer vision literature indicating that adequate context around target objects is crucial for effective feature learning in convolutional neural networks [29]. The superior performance of larger configurations (T20xx and T25xx) indicates that including more contextual information around vine trunk bases improves detection accuracy. However, performance plateauing and slight decrease in the largest configuration (T2550) suggests there is an optimal size beyond which additional context may introduce more noise than useful information.

The analysis reveals that rectangular bounding boxes with twice the height of the width generally show superior performance compared to square configurations. This finding is particularly relevant for elongated objects such as vine trunks and support posts, where natural object geometry favors vertical rectangles. Configuration T2040 (2.0W × 4.0W) achieved the highest mAP of 41.36%, demonstrating that vertically elongated bounding boxes effectively capture essential features of both vine trunks and support posts. This aspect ratio provides optimal balance between capturing sufficient trunk information and maintaining computational efficiency. The superior performance of T1530 compared to T1515 and T2040 compared to T2020 consistently supports the hypothesis that vertically elongated bounding boxes are more suitable for detecting elongated agricultural objects than square configurations.

Different optimal configurations for different classes emphasize the importance of considering object-specific characteristics in annotation design. Support posts, being more uniform and rigid structures, achieved best performance at T2020 (2.0W × 2.0W), while vine trunks, having more variable and curved nature, showed best performance at T2525 (2.5W × 2.5W). This class-specific variation can be attributed to different information content available within bounding boxes for each object type. Support posts maintain consistent vertical alignment and uniform thickness, enabling effective detection with more compact bounding boxes. However, vine trunks may exhibit curvature and varying thickness, requiring larger bounding boxes to capture sufficient discriminative features.

4.2. Effectiveness of Base-Width-Based Annotation Method

The experimental validation demonstrates that the base-width-based annotation method effectively generates valid training data for object detection models in agricultural environments. Eight aspect ratio bounding boxes were generated based on vine trunk base width, and optimal aspect ratios for each class (vine trunk, post) were experimentally confirmed. The experimental results show that the T2040 combination (width 2.0×, height 4.0×) achieved the highest mAP performance (41.36%), establishing the optimal bounding box configuration for vine trunk and post detection. This empirically proves that the base-width-based annotation method serves as an effective data generation approach for training object detection models in agricultural environments.

The base-width-based annotation method provides logical improvements to the annotation process compared to traditional annotation approaches in several aspects. First, annotators only need to specify the left and right endpoints of vine trunk bases, minimizing subjective judgment regarding ambiguous boundary determination of trunk upper portions. This significantly improves annotation consistency, particularly in agricultural environments where upper boundaries are unclear due to irregular plant forms. Second, since annotators only need to drag to appropriate points on the y-axis while maintaining the x-axis position of right endpoints, mouse movement distance is substantially reduced. This simplification of physical movements reduces wrist and shoulder muscle strain, decreasing muscle fatigue. This reduction in physical burden is particularly significant in agricultural robot development projects requiring large-scale annotation data generation.

The base-width standardized annotation method demonstrates the potential for establishing unified bounding box generation criteria for object recognition model development in autonomous agricultural operations. Automatic bounding box generation based on base width eliminates inter-annotator variability and ensures dataset consistency, enabling more stable deep learning model training. This methodological improvement leads to three main expected effects. First, improved annotation speed enables more efficient construction of large-scale training datasets required for agricultural robot development. Second, minimization of subjective judgment and standardized generation processes improve training data quality. Third, this methodology is expected to be applicable not only in vineyards but also in other orchard environments such as apple orchards and pear orchards, contributing to the proliferation of AI-driven agricultural automation technology.

4.3. Practical Implementation Considerations

Detection results demonstrate the practical feasibility of the proposed method for real agricultural robot applications. Successfully detected vine trunk bases and support post base points can be effectively processed to generate traversal information for autonomous agricultural vehicles. The ability to connect detected points into regression lines and calculate vanishing points provides the foundation for autonomous operation guidance systems. By comparing the camera's optical axis with detected central aisle lines, steering angles can be calculated to guide unmanned ground vehicles (UGVs) along vineyard rows. This practical application validates the core hypothesis that vine trunk base position detection is sufficient for autonomous traversal in semi-structured agricultural environments.

The method's focus on traversal-relevant portions of objects rather than complete visible structures aligns with the specific requirements of agricultural robots. Detection information generated by the proposed method can be readily integrated with existing agricultural robot control systems, providing computational efficiency that meets real-time processing requirements. The successful integration potential extends beyond immediate detection applications to comprehensive navigation systems that can handle the dynamic requirements of agricultural environments while maintaining the precision necessary for safe and effective autonomous operation.

4.4. Comparison with Traditional Methods

The proposed method addresses several key limitations of traditional annotation approaches identified in the literature. Manual bounding box annotation has been recognized as a significant bottleneck in dataset generation [23]. The proposed base-width-based annotation method substantially reduces this time by eliminating the need for precise upper boundary determination. Traditional annotation methods require annotators to make continuous decisions about bottom width, top width, and overall height for each bounding box. This process becomes increasingly inefficient as the volume of required annotations increases. The proposed method standardizes these decisions by automatically generating final bounding boxes based on vine trunk base width, eliminating inter-annotator variability and improving dataset consistency.

The consistency improvement achieved through automatic bounding box generation is particularly valuable in agricultural applications where high natural object variation can cause inter-annotator inconsistencies. Studies have shown that annotation consistency is crucial for stable object detection model training [22], and the proposed method's standardization directly addresses this issue. The physical ergonomics of the annotation process also benefits from this approach. Traditional annotation of elongated objects requires large mouse movements from top-left to bottom-right corners, leading to accumulated fatigue and increased processing time with continued annotation work. The proposed method's short mouse drag distance reduces physical strain and enables faster and more accurate annotation work.

Furthermore, the method's adaptation to the specific requirements of agricultural robots represents a departure from general-purpose annotation approaches. By focusing on the lower portion crucial for object traversal, the method optimizes annotation effort for specific application domains rather than attempting to capture complete object boundaries.

This domain-specific optimization approach demonstrates how annotation strategies can be tailored to meet the unique demands of specialized applications while improving both efficiency and effectiveness.

4.5. Limitations and Future Research

While the proposed method demonstrates significant advantages, several limitations must be acknowledged. The method is specifically designed for vertically oriented objects in semi-structured environments and may not directly generalize to other agricultural settings or object types without modification. The fixed aspect ratio approach, while improving consistency, may not be optimal for all variations in trunk shape and size. Future research could explore adaptive aspect ratio selection based on individual object characteristics or integrate multiple aspect ratios within a single training dataset.

The evaluation was conducted in a controlled greenhouse environment, and validation in outdoor vineyard conditions with varying lighting, weather, and seasonal changes would strengthen the generalizability of the findings. Additionally, verification of applicability across different grape varieties and cultivation methods is needed. For more quantitative validation of the base-width-based annotation method's effectiveness, actual annotation work time measurement, annotator fatigue assessment, and statistical analysis of annotation consistency are required. Research verifying applicability across various crop types and cultivation environments is also needed to expand the versatility of this methodology.

The method's performance could potentially be enhanced through integration with semi-automatic annotation tools or active learning approaches that could further reduce annotation time while maintaining quality. Through such additional research, standardized annotation protocols for agricultural robot development can be established. The development of adaptive systems that can automatically adjust aspect ratios based on object characteristics, combined with machine learning approaches that can learn optimal configurations from minimal user input, represents promising directions for future development in AI-driven agricultural automation applications.

5. Conclusions

This study proposed a base-width-based annotation method for vine trunk detection in agricultural robotics applications and comprehensively evaluated eight aspect ratio configurations. The experimental results demonstrated that bounding box size has a decisive impact on object detection performance in agricultural environments. Larger bounding box configurations consistently outperformed smaller ones, confirming the importance of capturing sufficient contextual information for agricultural object detection. Vertically elongated rectangular boxes proved more suitable than square configurations for elongated agricultural objects, and different object types required different optimal aspect ratios, emphasizing the need for object-specific annotation strategies. The base-width-based annotation method was designed to minimize subjective boundary judgment, reduce physical strain through shortened mouse movements, enhance data consistency through standardized bounding box generation, and focus on object regions critical for robot traversal. This method addresses practical challenges in agricultural robot dataset construction by eliminating ambiguous judgment about trunk upper boundaries and significantly reducing mouse movement distance required for annotation.

The study contributes to establishing efficient annotation protocols for agricultural robot development and demonstrates expandability to other fruit tree environments. The standardized bounding box approach is expected to improve the efficiency of artificial intelligence technology development for agricultural automation.

Future research will focus on validation in diverse field agricultural environments, quantitative analysis of annotation efficiency improvements, and application to other crops and cultivation systems. The standardized annotation protocol established through this research is expected to accelerate the development and deployment of autonomous agricultural robot systems, contributing to the advancement of AI-driven agricultural automation technology and the practical implementation of AI-driven solutions in agricultural environments.

Author Contributions: Conceptualization, software, writing and editing, H.-K.L.; data collection, annotation work, and dataset preparation, S.Y.; validation, review and discussion, S.P. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by the Basic Research Program (25-IT-03) through the Daegu Gyeongbuk Institute of Science and Technology (DGIST), and funded by the Ministry of Science and ICT (MSIT).

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Acknowledgments: The authors would like to express special thanks to the staff members of AMR Labs Inc. for their technical support in agricultural environment autonomous mobile robot (AMR) technology.

Conflicts of Interest: Author Seung Park was employed by the company AMR LABS Inc. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Lowenberg-DeBoer, J.; Huang, I.Y.; Grigoriadis, V.; Blackmore, S. Economics of robots and automation in field crop production. *Precis. Agric.* **2020**, *21*, 278–299. [CrossRef]
2. Shamshiri, R.R.; Weltzien, C.; Hameed, I.A.; Yule, I.J.; Grift, T.E.; Balasundram, S.K.; Pitonakova, L.; Ahmad, D.; Chowdhary, G. Research and development in agricultural robotics: A perspective of digital farming. *Int. J. Agric. Biol. Eng.* **2018**, *11*, 1–14. [CrossRef]
3. Chen, J.; Qiang, H.; Wu, J.; Xu, G.; Wang, Z. Navigation path extraction for greenhouse cucumber-picking robots using the prediction-point Hough transform. *Comput. Electron. Agric.* **2021**, *180*, 105911. [CrossRef]
4. Lyu, H.K.; Park, C.H.; Han, D.H.; Kwak, S.W.; Choi, B. Orchard free space and center line estimation using naive Bayesian classifier for unmanned ground self-driving vehicle. *Symmetry* **2018**, *10*, 355. [CrossRef]
5. Zhang, L.; Liu, Y.; Wang, J.; Chen, X. An autonomous navigation method for orchard rows based on a combination of an improved A-star algorithm and SVR. *Precis. Agric.* **2024**, *25*, 1142–1165. [CrossRef]
6. Li, H.; Huang, K.; Sun, Y.; Lei, X.; Yuan, Q.; Zhang, J.; Lv, X. An autonomous navigation method for orchard mobile robots based on octree 3D point cloud optimization. *Front. Plant Sci.* **2024**, *15*, 1510683. [CrossRef]
7. Aghi, D.; Mazzia, V.; Chiaberge, M. Local motion planner for autonomous navigation in vineyards with a RGB-D camera-based algorithm and deep learning synergy. *Machines* **2020**, *8*, 27. [CrossRef]
8. Han, Z.; Li, J.; Yuan, Y.; Fang, X.; Zhao, B.; Zhu, L. Path Recognition of Orchard Visual Navigation Based on U-Net. *Trans. Chin. Soc. Agric. Mach.* **2021**, *52*, 30–39.
9. Mazzia, V.; Salvetti, F.; Chiaberge, M. Position-agnostic autonomous navigation in vineyards with deep reinforcement learning. *Eng. Appl. Artif. Intell.* **2022**, *112*, 104868.
10. Adhikari, S.P.; Yang, C.; Kim, H. Learning semantic segmentation of large-scale point clouds with random sampling. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 8793–8808.
11. Chen, J.; Wang, H.; Zhang, L.; Wu, Y. An Autonomous Navigation Method for Orchard Robots Based on Machine Vision and YOLOv4 Model. *Sensors* **2022**, *22*, 3215.
12. Zhang, Q.; Liu, Y.; Wang, J.; Zhang, H.; Chen, X. Improved Hybrid Model of Autonomous Navigation in Orchard Environment Based on YOLOv7 and RRT. *Sensors* **2024**, *24*, 975.
13. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
14. Zhu, H.; Jing, D. Optimizing Slender Target Detection in Remote Sensing with Adaptive Boundary Perception. *Remote Sens.* **2024**, *16*, 2643. [CrossRef]

15. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
16. Biondi, F.N.; Cacanindin, A.; Douglas, C.; Cort, J. Overloaded and at work: Investigating the effect of cognitive workload on assembly task performance. *Hum. Factors* **2021**, *63*, 813–820. [CrossRef]
17. Liu, L.; Ouyang, W.; Wang, X.; Fieguth, P.; Chen, J.; Liu, X.; Pietikäinen, M. Deep learning for generic object detection: A survey. *Int. J. Comput. Vis.* **2020**, *128*, 261–318. [CrossRef]
18. Zhao, Z.Q.; Zheng, P.; Xu, S.T.; Wu, X. Object detection with deep learning: A review. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *30*, 3212–3232. [CrossRef]
19. Silva Aguiar, A.; Monteiro, N.N.; Santos, F.N.; Pires, E.J.S.; Silva, D.; Sousa, A.J.; Boaventura-Cunha, J. Bringing semantics to the vineyard: An approach on deep learning-based vine trunk detection. *Agriculture* **2021**, *11*, 131. [CrossRef]
20. Santos, T.T.; de Souza, L.L.; dos Santos, A.A.; Avila, S. Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association. *Comput. Electron. Agric.* **2020**, *170*, 105247. [CrossRef]
21. Papadopoulos, D.P.; Uijlings, J.R.; Keller, F.; Ferrari, V. Extreme clicking for efficient object annotation. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4930–4939.
22. Konyushkova, K.; Uijlings, J.; Lampert, C.H.; Ferrari, V. Learning intelligent dialogs for bounding box annotation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 9175–9184.
23. Ma, J.; Liu, C.; Wang, Y.; Lin, D. The effect of improving annotation quality on object detection datasets. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), New Orleans, LA, USA, 19–24 June 2022; pp. 3121–3130.
24. Murrugarra-Llerena, R.; Price, B.L.; Cohen, S.; Liu, B.; Rehg, J.M. Can we trust bounding box annotations for object detection? In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), New Orleans, LA, USA, 19–24 June 2022; pp. 3001–3010.
25. Shi, C.; Yang, S. The devil is in the object boundary: Toward annotation-free instance segmentation. In Proceedings of the International Conference on Learning Representations (ICLR), Vienna, Austria, 5–9 May 2024.
26. Li, J.; Xiong, C.; Socher, R.; Hoi, S.C.H. Towards noise-resistant object detection with noisy annotations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11457–11465.
27. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
28. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767. [CrossRef]
29. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 818–833.

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.