MMD-YOLO: A Multi-scale Maturity Detection Model For Occluded Apples In Complex Environments

Yuehua Zhang[1*], Xinguang Zhao[2], and Fengting Liu[1]

[1] School of Information Engineering Shandong Huayu University of Technology, De Zhou, China, 253034
[2] Finance Department, Huarui International Engineering Consulting Group Co., Ltd.
[*] Corresponding author. E-mail: yuehuazhang05@outlook.com

Received: Mar. 30, 2026; Accepted: May 13, 2026

Apple production is vital to global horticulture, with precision agriculture increasingly using computer vision and deep learning. This study evaluates apple detection and maturity classification using the AppleBBCH76 and AppleBBCH81 datasets, comprising 5,007 annotated orchard images. You Only Look Once (YOLO) models support fruit detection, yield prediction, and maturity analysis, yet orchard challenges like occlusion, overlap, variable sizes, and lighting reduce accuracy. To address this, a Multi-scale Maturity Detection-YOLO (MMD-YOLO) model integrates multi-scale feature extraction with a fine-tuned YOLOv11. Pre-processing included resizing, median filtering, Contrast Limited Adaptive Histogram Equalization (CLAHE), and normalization, with Bayesian optimization. Results show 93.40% mAP@0.5, 94.60% precision, 93.50% recall, and 94% F1-score, surpassing earlier YOLO versions.

Keywords: Apple maturity detection; YOLOv11; multi-scale feature extraction; Occluded apple detection; Orchard monitoring; Precision agriculture

http://dx.doi.org/10.6180/jase.202610_33.006

1. Introduction

Apples (Malus domestica) are among the most widely cultivated fruit crops worldwide, contributing significantly to global agriculture and food production [1]. Apples are nutrient-rich, economically vital, and show region-specific maturity influenced by temperature, sunlight, rainfall, and soil moisture [2], while climate change further disrupts flowering and ripening stages, affecting harvest timing and fruit quality [3]. Fruit quality depends on precise maturity, so accurate harvest timing and reliable datasets with digital tools ensure sweetness, firmness, shelf life, and trustworthy orchard planning [4]. AI and deep learning have advanced orchard monitoring, automating maturity detection early CNNs classified ripeness using color and texture features [5], while platforms such as Smart Harvest enabled rapid online maturity estimation [6]. Recent work emphasizes real-time object detection frameworks that identify apples in orchard scenes, estimating size and location using large annotated RGB-D datasets [7]. This research falls in line with the ongoing efforts in the design and fabrication of sustainable agricultural and industrial machinery. Precision farming now relies on smart technologies that incorporate sustainability into orchard surveillance and fruit ripeness assessment. Occlusion-aware detection improved recognition of overlapping fruits, and robotic harvesting systems highlighted the need for accurate perception models [8]. Multi-modal sensing combining RGB and depth data enhanced fruit identification in complex scenes [9]. The study advances from simple ripeness classification to robust real-time detection, introducing MMD-YOLO for multi-stage apple maturity under complex orchard occlusion.

1.1. Problem Statement

Changing light and shadows reduce the reliability of color-based maturity assessment [10], while varying camera distances and angles distort apple sizes, causing detection failures for small or hidden fruits [11]. Thus, orchard conditions demand a multi-scale feature extraction.

1.2. Research Gap

Despite advances with RGB-D sensors and multi-scale YOLO variants, apple detection struggles in crowded orchards. MMD-YOLO introduces a lightweight YOLOv11 framework with multi-scale fusion, optimized pre-processing, and Bayesian tuning, enabling robust maturity detection under occlusion and illumination variability.

The primary contributions of this study are summarized as follows:

- Develops an apple maturity detection model using MMD-YOLO with AppleBBCH76 and AppleBBCH81 datasets.
- Applies image pre-processing (CLAHE, median filtering, resizing, normalization) to improve training quality.
- Uses multi-scale feature extraction in YOLOv11 to identify apples of varying sizes and occluded fruits.
- Evaluates performance with precision, recall, and mAP@0.5 for precision agriculture applications.

1.3. Research Organization

The paper begins with section 2 which examines existing research about apple detection. section 3 explains the proposed MMD-YOLO method, section 4 presents the experimental results, and section 5 concludes the paper with future work.

2. Literature survey

The detection of apples in orchard environments is challenged by variable lighting, occlusions, and overlapping fruits. Chu et al. [12] localized partially occluded apples using the O2RNet model, achieving 94% precision and an F1-score of 0.88. Li et al. [13] reduced localization errors by 59% and 43% with a frustum point cloud segmentation model based on RGB-D data. Fu et al. [14] improved YOLOv10 with multi-scale layers, SE attention, and diversity blocks, reaching 92.8%mAP and 89.8% recall. Hu et al. [10] integrated RGB-D features into YOLO, reporting 94.1%mAP, 93% F1-score, and 167 FPS speed. Designing and fabricating sustainable agricultural and industrial machinery involves lightweight and energy-efficient detection mechanisms. Previous research shows that efficient YOLO architectures and multimodal sensing promote sustainability through low resource utilization and enhanced performance in challenging orchard settings. Liu et al. [15] fused RGB, depth maps, feature maps, and point clouds, achieving 95.8% precision in cluttered orchard scenes. To overcome these challenges, this research proposes an optimized YOLOv11 model that efficiently determines apple fruit maturity by addressing identification issues in complex orchard conditions.

3. Methodology

The MMD-YOLO workflow begins with orchard image collection under diverse conditions, then proceeds through pre-processing (resize 640×640, median filtering, CLAHE, normalization) and Bayesian optimization before YOLOv11 training; as shown in Fig. 1, this enables robust detection and maturity classification under occlusion.

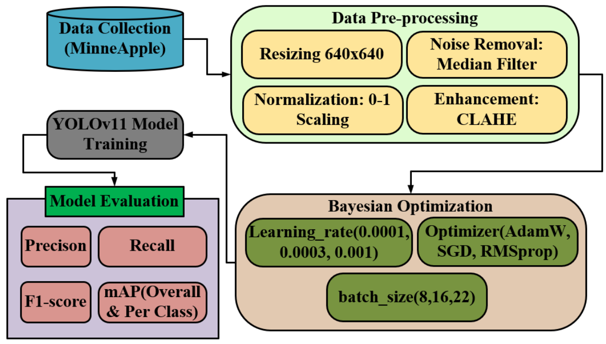

Fig. 1. Proposed MMD-YOLO Workflow

YOLOv11 trains on annotated orchard images with Bayesian-optimized parameters, then evaluates precision, recall, F1, and mAP for efficient apple maturity detection.

3.1. Data Collection

The study uses the AppleBBCH76 [16] and AppleBBCH81 [17] datasets, combining 3,169 and 1,838 orchard images respectively for a total of 5,007. The datasets provide ripe/unripe annotations across 5,007 orchard images under varied conditions, serving as ground truth for supervised maturity classification.

Annotation formats from AppleBBCH76 and AppleBBCH81 were standardized into a unified schema, ensuring consistent ripe/unripe labels, reducing bias, and improving reproducibility.

3.2. Data Pre-Processing

The MMD-YOLO model pre-processes all apple images, resizing them to 640×640 for uniform input and improved detection as expressed in Eq. (1):

$$\tilde{x}_i = \mathcal{R}\left(x_i; 640, 640\right) \tag{1}$$

In this case, the original image is denoted as x_i, the resized image as $\tilde{x}_i$ and the resizing function as $\mathcal{R}(.) ensuringthatallinputsfedintothemodelshareidenticalspatiald$ The filtered image is expressed as shown in Eq. (2):

$$\hat{x}_i = \mathcal{M}\left(\tilde{x}_i\right) \tag{2}$$

Here, the $\mathcal{M}(\cdot)$ is the median filtering operation, and also $\hat{x}_i$ is the noise filtered image. Implementation of median filter with kernel size 3×3 is beneficial in eliminating the effect of noise while retaining edges, and setting of parameters such as clip limit equal to 2.0 and tile grid size equal to 8×8 for CLAHE makes it capable of adapting to varying lighting conditions. CLAHE technique could be applied to $\hat{x}_i$, because it may affect to the outdoor different lighting conditions. Then the normalized pixel $p'_{i,j}$ is prime for a pixel $p_{i,j}$ at location (i, j) can be calculated as follows in Eq. (3):

$$p'_{i,j} = \frac{p_{i,j}}{255} \tag{3}$$

Here, the original pixel intensity is represented by $p_{i,j} \in [0, 255]$ and the normalized value is represented by $p'_{i,j} \in [0, 1]$. After all the pre-processing steps, the dataset is split into training and validation. Median filtering (3×3) and CLAHE (clip limit 2.0, tile grid 8×8) were empirically chosen to balance noise reduction and detail preservation, ensuring reproducibility across orchard imaging conditions.

3.3. Model Architecture

Built on YOLOv11, the system inputs 640×640 orchard images, applies C3K2 blocks, and outputs ripe/unripe apple bounding boxes, effectively capturing shape, texture, and edges for accurate orchard detection. The C3K2 blocks combine multiple convolutional kernels with residual connections to expand the receptive field while maintaining efficiency, enabling the network to capture fine textures and broader contextual cues. By preserving edge continuity and texture gradients under occlusion, they strengthen feature robustness and help distinguish partially hidden apples from complex orchard backgrounds, thereby improving detection accuracy in dense scenes. Efficient feature extraction separates apples from complex orchard backgrounds, and the SPPF module integrates multi-scale receptive fields to enable detection across different spatial scales. The SPPF module uses multi-scale pooling to aggregate fine details and broader context, enhancing apple detection under foliage, occlusion, and variable illumination.

The YOLOv11 architecture shown in Fig. 2 enables precise detection of apples at various maturity stages throughout orchard environments that include leaf and branch obstacles as well as changing light conditions.

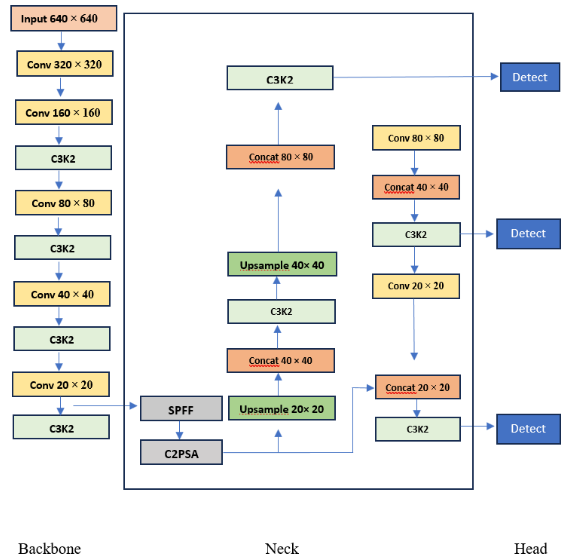

Fig. 2. YOLOv11 Model Architecture

3.3.1. *Multi-Scale Feature Integration for Apple Detection*

The input orchard image is standardized to 640×640 RGB for spatial and color consistency, and multi-scale feature maps are extracted to capture fine details like edges and occluded fruits alongside broader contextual cues. Its mathematical expressions are provided below in Eqs. (4) to (7).

a) Input Representation:

$$I \in R^{H \times W \times 3}, H = W = 640 \tag{4}$$

b) Multi-Scale Feature Maps:

$$F_s = \varnothing_s(I), \quad s \in \{20, 40, 80\} \tag{5}$$

c) Feature Fusion:

$$F_{\text{fusion}} = \sum_s w_s \cdot U\left(F_s\right) \tag{6}$$

d) Classification Probability:

$$P(c \mid I) = \text{Softmax}\left(W \cdot F_{\text{fusion}} + b\right), \quad c \in \{\text{ripe}, \text{unripe}\} \tag{7}$$

3.3.1.1 Softmax vs. Sigmoid for Binary Classification

In the MMD-YOLO framework, softmax was adopted for binary maturity classification (ripe vs. unripe) because it produces well-calibrated probability distributions that sum to one, improving interpretability and confidence calibration. Although ripe and unripe apples may share similar color tones under shadow or partial ripening, softmax mitigates inflated confidence by balancing probabilities, while multi-scale features such as texture and shape strengthen separability, ensuring reliable detection even when color similarity challenges accuracy. Compared to sigmoid, which outputs independent probabilities for each class and often suffers from poor calibration under class imbalance, softmax enforces mutual exclusivity by normalizing probabilities across classes. This improves calibration quality, prevents dominance of one maturity stage, and yields more consistent confidence scores across varying orchard conditions, ensuring stable detection even when ripe and unripe apples share similar visual traits.

The detection head uses scale-specific predictions to adapt to apples of varying sizes and distances, with high-resolution maps (80 × 80) capturing small or occluded fruits, intermediate maps (40 × 40) balancing detail and context for medium apples, and low-resolution maps (20 × 20) identifying larger apples and clusters. By combining outputs across these scales, the network reduces false negatives for small fruits and false positives in cluttered backgrounds, enhancing overall detection robustness in complex orchard scenes. Bounding box regression minimizes localization error using CIoU, and its expression is provided in Eq. (8) while the total loss combines classification, objectness, and regression terms with weighting factors for balanced optimization and its expression is provided in Eq. (9).

e) Bounding Box Regression:

$$b=(x, y, w, h), L_{\text{box}}=1-\operatorname{CIoU}\left(b, b^{g t}\right) \tag{8}$$

f) Total Loss Function:

$$L_{\text{total}}=\lambda_{\text{cls}} L_{\text{cls}}+\lambda_{\text{obj}} L_{\text{obj}}+\lambda_{\text{box}} L_{\text{box}} \tag{9}$$

Here, I is the input orchard image with spatial dimensions H, W and $\mathrm{RGB}(3)$; Fs is represented as the feature map at scale transformed by $\varnothing_s(.) with resolutions$; F_{fusion} is the fused representation weighted by w_s and aligned via $U\left(F_s\right)$, then summed across scales; $P(c \mid I)$ gives class probability using Softmax normalization. Detection uses weight matrix W, bias b and ground-truth box b^{gt}.Losses include $L_{cls}, L_{obj}, L_{\text{box}}$ with $CIoU$ combined as L_{total} with weighting factors $\lambda_{cls}, \lambda_{obj}, \lambda_{box}$ to balance optimization and ensure balance detection.

In practice, the multi-scale fusion mechanism integrates feature maps at $80 \times 80, 40 \times 40$, and 20×20 resolutions, with fine-scale maps capturing edges and occluded fruits, medium-scale maps balancing detail and context, and large-scale maps representing orchard structures for larger apples and clusters.

In YOLOv11, multi-scale fusion is implemented through the Neck, which up-samples lower-resolution maps and concatenates them with higher-resolution maps, combining semantic information from deeper layers with fine details from shallow ones. Together, this hierarchical fusion enhances detection accuracy and stability under diverse orchard conditions.

3.4. Hyperparameter Optimization

The MMD-YOLO network achieves its best performance when training hyperparameters are properly tuned, with learning rate, optimizer, and batch size being key factors that determine convergence and detection accuracy. Bayesian optimization was used to find appropriate hyperparameter settings for the study. Hyperparameter settings were optimized using Bayesian methods, which reduce unnecessary trials and provide efficient tuning well suited to the computational demands of YOLOv11 training.

3.4.1. Gaussian Process Kernel Selection

In Bayesian optimization, the surrogate model was implemented using a Gaussian Process (GP). The Matérn kernel was chosen for its balance of smoothness and flexibility, enabling the surrogate to capture non-linear relationships between hyperparameters and validation loss. Compared to the RBF kernel, its better models' sharp performance changes when hyperparameters vary.

3.4.2. Hyperparameter Priors

The search space was defined with prior ranges:

- Learning rate ($\theta 1$) was assumed to follow a log-uniform prior in the range $[1 \times 10-4, 1 \times 10-3]$.
- Optimizer ($\theta 2$) was treated as a categorical prior with values {AdamW, SGD, RMSprop}.
- Batch size ($\theta 3$) was modelled as a discrete uniform prior over integers [8,32]. These priors ensured that the optimization process explored realistic ranges without biasing toward extreme values.

3.4.3. Acquisition Function Behavior

The EI acquisition function was used to balance exploration and exploitation by selecting hyperparameter configurations that maximize expected gain over the current best validation loss. It explores uncertain regions when surrogate variance is high and exploits promising regions when the surrogate mean suggests likely improvement. This adaptive behavior reduces wasted trials compared to grid search.

Bayesian optimization sequentially builds a probabilistic surrogate of the target function, selecting new hyperparameters based on prior results to reduce computation

Table 1. Hyperparameter Values

Hyperparameter	Type	Search Range
learning rate	Continuous	$1e-4-1e-3$
optimizer	Categorical	AdamW, SGD, RMSprop
batch size	Integer	$8-32$

compared to exhaustive grid search. The candidate values considered during optimization are listed in Table 1.

The Bayesian optimization framework uses a Gaussian Process surrogate to estimate how hyperparameters affect validation loss, with the EI acquisition function guiding exploration of uncertain regions and exploitation of promising ones. The presented MMD-YOLO system combines pre-processing, multiscale fusion, and Bayesian optimization within a framework derived from sustainable agriculture and industry machinery design. The design process guarantees computation efficiency for immediate deployment in orchards without extensive hardware requirements.

4. Results and discussion

The objective of this research is to build an accurate apple detection system for real orchard environments, evaluated under extreme conditions using Precision, Recall, F1-score, and mAP. Implemented in Python with Ultralytics YOLOv11, OpenCV, and supporting libraries, the model achieves high accuracy while reflecting principles of sustainable agricultural and industrial machinery. By combining robust detection with reduced computational overhead, it supports eco-friendly orchard management and scalable industrial applications.

4.1. Performance Evaluation

Performance evaluation uses metrics like mAP, precision, recall, and F1-score. Sensitivity analysis shows CLAHE (clip limit 2.0), a 3×3 median filter, and Min-Max normalization yield the best accuracy (mAP@0.5 $= 0.934$, precision $= 0.946$, recall $= 0.935$, F1 $= 0.94$), while larger kernels or other normalization reduce robustness under orchard variability.

The model's ability is measured using the confusion matrix. Fig. 3 shows the confusion matrix of the proposed model.

The confusion matrix shows most ripe and unripe apples were correctly classified, confirming the model's ability to learn distinguishing visual traits.

Here, Table 3 shows YOLOv11 achieving the highest accuracy (95.01%) and precision (98.19%), with few incorrect

Table 2. Sensitivity Analysis of Pre-Processing Parameters

Method	Para	mAP@0.5	Prec	Rec	F1
CLAHE	1.0	0.912	0.931	0.906	0.918
	2.0	**0.934**	**0.946**	**0.935**	**0.94**
	3.0	0.928	0.941	0.929	0.935
Median Filter	3×3	**0.934**	**0.946**	**0.935**	**0.94**
	5×5	0.926	0.938	0.928	0.933
	7×7	0.918	0.931	0.92	0.925
Norm	Min Max	**0.934**	**0.946**	**0.935**	**0.94**
	Z-score	0.927	0.939	0.929	0.934

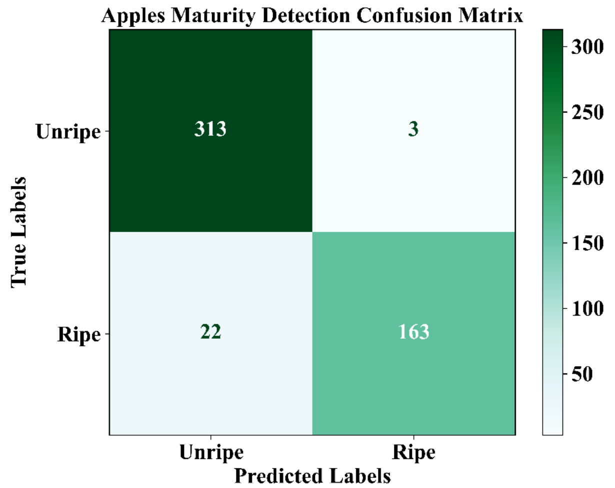

Fig. 3. Confusion Matrix

Table 3. Classification Performance Comparison

Model	Accuracy	Precision	Recall	F1-Score
YOLOv5	0.9024	0.9142	0.8613	0.8870
YOLOv7	0.9236	0.9421	0.8834	0.9118
YOLOv8	0.9412	0.9638	0.8946	0.9279
YOLOv10	0.9445	0.9712	0.9015	0.9350
YOLOv11 (Proposed)	**0.9501**	**0.9819**	**0.9211**	**0.9588**

maturity predictions an essential feature for preventing premature fruit harvesting.

The Fig. 4 shows precision rising with confidence threshold, with ripe apples performing better due to distinct color and texture, while unripe apples are harder to detect because of foliage similarity and fewer annotated samples, leading to slight imbalance and lower precision.

The proposed model is compared with other versions of YOLO to see if there is any possibility to use any other model for apple detection in orchard environments.

The Table 4 shows YOLOv11 outperforming earlier versions with mAP@0.5 $= 0.9340$, F1-score $= 0.9400$, and consistent gains in precision, recall, and IoU, highlighting its stronger balance in complex orchard conditions.

Table 4. Object Detection Performance Comparison

Model	mAP@0.5	mAP@0.5:0.95	Precision	Recall	F1-score	Avg IoU
YOLOv5	0.8924	0.6127	0.9016	0.8875	0.8945	0.7632
YOLOv7	0.9083	0.6341	0.9187	0.9013	0.9099	0.7814
YOLOv8	0.9215	0.6519	0.9324	0.9142	0.9232	0.7968
YOLOv10	0.9287	0.6598	0.9405	0.9221	0.9312	0.8043
YOLOv11 (Proposed)	**0.9340**	**0.6680**	**0.9460**	**0.9350**	**0.9400**	**0.8120**

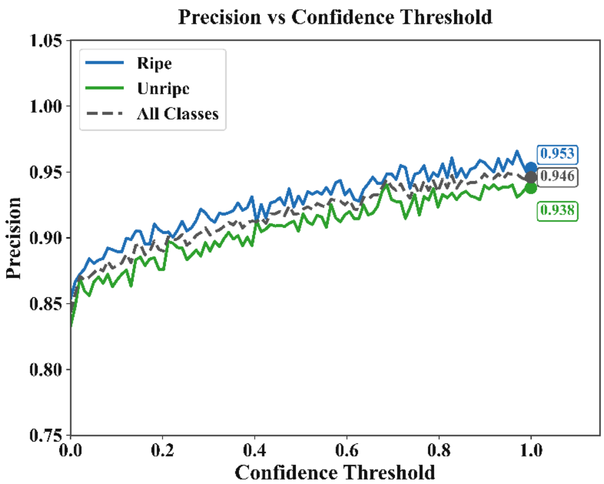

Fig. 4. Precision vs Confidence Threshold

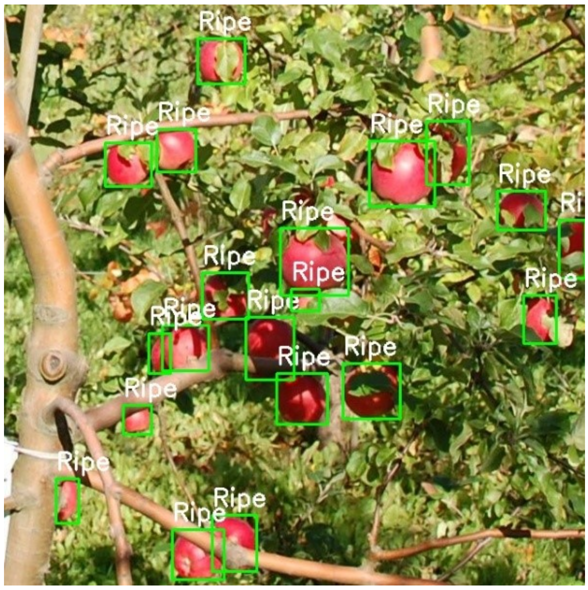

Fig. 5. Ripe Apples Detection

Fig. 6. Apple Unripe Detection

4.2. Qualitative Detection Results

The performance of the model can be assessed by using the metrics to show how the model can be operated efficiently in actual environment.

Fig. 5 shows ripe apples detected in red bounding boxes, even in dense foliage with partial occlusion, correctly locating and classifying them as ripe.

Fig. 6 shows unripe apples detected in green bounding boxes, correctly classified despite orchard challenges, demonstrating the model's strong performance and readiness for real-world deployment.

4.3. Computational Complexity Analysis

MMD-YOLO was evaluated for inference speed, parameter count, and memory usage to assess real-time feasibility. On an RTX 3060 GPU, the model achieves $\sim$ 45 FPS ($\approx$ 22 ms /image), while on a mid-range CPU inference time is $\sim$ 120 ms /image. The optimized YOLOvl1 backbone contains $\sim$ 38M parameters, lighter than multi-modal detection models. GPU memory usage is $\sim$ 2.8 GB and CPU memory consumption $<$ 1.5 GB, confirming compatibility with consumer-grade GPUs and embedded devices. These results demonstrate that MMD-YOLO is suitable for real-time orchard deployment, supporting integration with drones and robotic harvesters for automated monitoring.

The Fig. 7 presents the class-wise Average Precision at

Table 5. Error Categorization Analysis

Error Type	Percentage (%)	Description
Occlusion	41.80%	Apples partially hidden by leaves/branches
Illumination Variation	27.60%	Shadows / uneven lighting
Background Similarity	19.30%	Green apples vs foliage
Small Object Detection	11.30%	Distant or tiny apples

IoU = 0.5, showing 0.948 for ripe apples and 0.920 for unripe apples. The results highlight slightly higher detection accuracy for ripe apples, indicating class-specific performance differences.

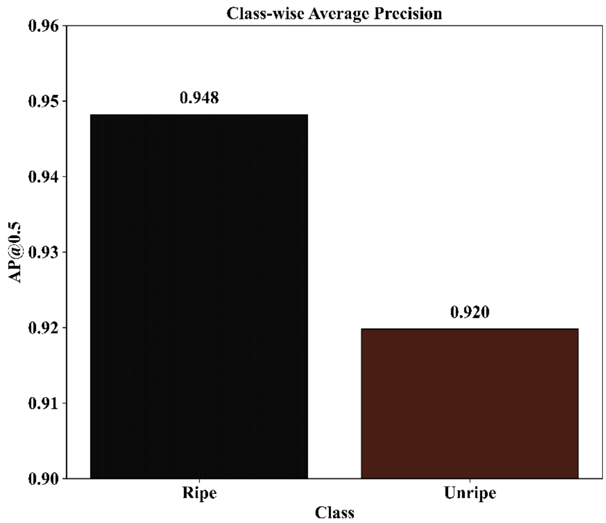

Fig. 7. Class-wise Average Precision

The Fig. 8 shows mAP declining steadily as IoU thresholds tighten, highlighting localization precision and robustness, with mAP@0.5 : 0.95 serving as a comprehensive reliability measure.

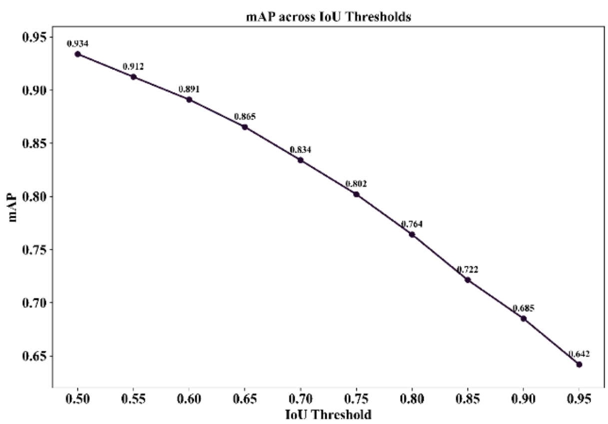

Fig. 8. mAP across IoU Thresholds

The misclassifications were categorized into occlusion, illumination variation, background similarity, and small object detection. Occlusion contributed the largest share of errors (41.8%), followed by illumination variation (27.6%) and background similarity (19.3%).

4.4. Ablation Studies

An ablation study systematically removes or varies model components to assess their impact; Table 6 summarizes the effectiveness of the C3K2 blocks.

Table 6. Ablation Study on C3K2 Blocks

Model Variant	mAP@0.5
Baseline	0.912
Backbone + C3K2	0.925
Neck + C3K2	0.928
Full Model (Backbone + Neck C3K2)	0.934

The baseline YOLOv11 achieved 0.912 mAP@0.5; adding C3K2 blocks to the backbone improved it to 0.925, to the neck 0.928, and to both 0.934 the highest accuracy showing that C3K2 blocks progressively enhance feature representation under occlusion and are essential for robust orchard detection.

5. Conclusion

The study developed an apple maturity detection framework using YOLOv11 on 5,007 orchard images with preprocessing (resizing, median filtering, CLAHE, normalization). The model achieved 95.01% accuracy, 98.19% precision, 88.11% recall, F192.88%, and detection mAP@0.5 = 0.9340, mAP@0.5 : 0.95 = 0.6680, precision 0.9460, recall 0.9350, IoU 0.8120. This research contributes to the design and fabrication of sustainable agricultural and industrial machinery by presenting a lightweight, efficient detection model for apple maturity. The framework enhances sustainability in agriculture, enabling integration with drones and robotic harvesters for automated, resource-conscious orchard monitoring. These results confirm robust performance under occlusion, overlap, and variable lighting, enabling efficient real-time orchard deployment superior to previous YOLO versions.

The proposed MMD-YOLO model enhances orchard management by accurately detecting apple maturity in real conditions. Reliable classification of ripe and unripe apples helps optimize harvest timing, preserve fruit quality, and reduce losses, while supporting labour and supply chain planning. Integration with drones or robotic harvesters enables automated, efficient, and sustainable large-scale orchard monitoring.

5.1. Future Work

Future work includes expanding datasets across diverse orchards and seasons to strengthen robustness, extending to more apple varieties for generalization, integrating with drones and robotic harvesters for automation, and combining multi-spectral and depth imaging to improve detection under foliage and lighting challenges.

6. Acknowledgements

- **Funding:** This paper was funded by the Dezhou Engineering Research Center for Big Data and Intelligent Perception Technology (Platform No. PT2025KJT002).

- **Data Availability:** The datasets used and/or analyzed during the current study are available from the corresponding author upon reasonable request. All relevant data supporting the findings of this study are provided in the paper or accessible via supplementary materials.

- **Conflicts of Interest:** The authors declare that there are no conflicts of interest related to this study.

- **Author Contribution:**

  Yuehua Zhang: Conceptualization, methodology, model development, experimental design, data analysis, and manuscript writing.

  Xinguang Zhao: Data collection, image preprocessing, and experimental setup.

  Fengting Liu: Model training, Bayesian optimization, and manuscript review.

- **Ethical Approval:** This study does not involve human or animal subjects, and thus ethical approval was not required. All experimental procedures were carried out in compliance with the relevant laws and institutional guidelines.

- **Consent to Participate:** As this study did not involve human participants, informed consent is not applicable.

- **Consent to Publication:** The authors give consent for the publication of the manuscript in the Journal of Applied Science and Engineering.

- **Competing Interests:** The authors declare that there are no competing interests in the publication of this paper.

References

[1] H. Tang, X. Zhai, and X. Xu, (2025) *"Flowering variation induces apple maturity variation at harvest"* **Frontiers in Agronomy** ***7***: 1545070. DOI: 10.3389/fagro.2025.1545070.

[2] J.-C. Lee, Y.-S. Park, H.-N. Jeong, J.-H. Kim, and J.-Y. Heo, (2023) *"Temperature Changes Affected Spring Phenology and Fruit Quality of Apples Grown in High-Latitude Region of South Korea"* **Horticulturae** ***9***(7): 794. DOI: 10.3390/horticulturae9070794.

[3] M. Lv, Y.-X. Xu, Y.-H. Miao, and W.-H. Su, (2025) *"A comprehensive review of deep learning in computer vision for monitoring apple tree growth and fruit production"* **Sensors** ***25***(8): 2433. DOI: 10.3390/s25082433.

[4] Y. Pan, Y. Chen, X. Tong, M. Liu, A. Huang, M. Zhou, and Y. Hu, (2026) *"YOLO-CSB: A Model for Real-Time and Accurate Detection and Localization of Occluded Apples in Complex Orchard Environments"* **Agronomy** ***16***(3): 390. DOI: 10.3390/agronomy16030390.

[5] B. Xiao, M. Nguyen, and W. Q. Yan. "Apple ripeness identification using deep learning". In: *International symposium on geometry and vision*. Springer. 2021, 53–67. DOI: 10.1007/978-3-030-72073-5_5.

[6] Z. Yan, Y. Wu, W. Zhao, S. Zhang, and X. Li, (2025) *"Research on an Apple Recognition and Yield Estimation Model Based on the Fusion of Improved YOLOv11 and DeepSORT"* **Agriculture** ***15***(7): 765. DOI: 10.3390/agriculture15070765.

[7] L. Sun, G. Hu, C. Chen, H. Cai, C. Li, S. Zhang, and J. Chen, (2022) *"Lightweight apple detection in complex orchards using YOLOV5-PRE"* **Horticulturae** ***8***(12): 1169. DOI: 10.3390/horticulturae8121169.

[8] Y. Huang, S. Xu, H. Chen, G. Li, H. Dong, J. Yu, X. Zhang, and R. Chen, (2025) *"A review of visual perception technology for intelligent fruit harvesting robots"* **Frontiers in Plant Science** ***16***: 1646871. DOI: 10.3389/fpls.2025.1646871.

[9] D. Wang and D. He, (2022) *"Apple detection and instance segmentation in natural environments using an improved Mask Scoring R-CNN Model"* **Frontiers in Plant Science** ***13***: 1016470. DOI: 10.3389/fpls.2022.1016470.

[10] T. Hu, W. Wang, J. Gu, Z. Xia, J. Zhang, and B. Wang, (2023) *"Research on Apple Object Detection and Localization Method Based on Improved YOLOX and RGB-D Images"* **Agronomy** ***13***(7): 1816. DOI: 10.3390/agronomy13071816.

[11] X. Wang, Y. Huang, S. Wei, W. Xu, X. Zhu, J. Mu, and X. Chen, (2025) *"Eld-yolo: A lightweight framework for detecting occluded mandarin fruits in plant research"* **Plants** ***14***(11): 1729. DOI: 10.3390/plants14111729.

[12] P. Chu, Z. Li, K. Zhang, D. Chen, K. Lammers, and R. Lu, (2023) *"O2RNet: Occluder-occludee relational network for robust apple detection in clustered orchard environments"* **Smart Agricultural Technology** ***5***: 100284. DOI: 10.1016/j.atech.2023.100284.

[13] T. Li, Q. Feng, Q. Qiu, F. Xie, and C. Zhao, (2022) *"Occluded Apple Fruit Detection and Localization with a Frustum-Based Point-Cloud-Processing Approach for Robotic Harvesting"* **Remote Sensing** ***14***(3): 482. DOI: 10.3390/rs14030482.

[14] H. Fu, Z. Guo, Q. Feng, F. Xie, Y. Zuo, and T. Li, (2024) *"MSOAR-YOLOv10: Multi-Scale Occluded Apple Detection for Enhanced Harvest Robotics"* **Horticulturae** ***10***(12): 1246. DOI: 10.3390/horticulturae10121246.

[15] X. Liu, C. Hu, X. Huang, C. Sun, R. Zhu, C. Wang, Y. Zhang, Q. Shen, H. Zhou, and C. Ruan, (2025) *"The apple detection method based on multimodal features"* **PLoS One** ***20***(10): e0334911. DOI: 10.1371/journal.pone.0334911.

[16] RTA & LatHort projects. *AppleBBCH76*. https://www.kaggle.com/datasets/projectlzp201910094/applebbch76 [Accessed: 2026-04-11]. 2026.

[17] RTA & LatHort projects. *AppleBBCH81*. https://www.kaggle.com/datasets/projectlzp201910094/applebbch81 [Accessed: 2026-04-11]. 2026.