Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Design and performance validation of a lightweight YOLOv11n-based visual detection system for apple harvesting robots

Xinying Lin

283003677@qq.com

[redacted]

Zongde Lin

[redacted]

Ping Yu

[redacted]

Article

Keywords: Apple harvesting robot, YOLOv11n, dynamic snake convolution, efficient feature aggregation, global context attention, Wise-IoU v3, edge deployment, visual detection

Posted Date: November 25th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11683726/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Design and performance validation of a lightweight YOLOv11n-based visual detection system for apple harvesting robots

Xinying Lin[1]*, Zongde Lin[1], Ping Yu[1]

1 Minnan University of Science and Technology，Shishi 362700; 283003677@qq.com
*Minnan University of Science and Technology，Shishi 362700; Correspondence: 283003677@qq.com; Tel.: 15260767810

Abstract

Real-time and reliable fruit detection under uneven illumination, branch–leaf occlusion and overlapping clusters is a prerequisite for robotic apple harvesting. This study proposes YOLOv11n-EFAN, a lightweight detector built on YOLOv11n and deployed on an edge-computing harvesting platform. The technical contributions are: (i) a dynamic snake convolution (DSC) module embedded in the C3k2 backbone for adaptive sampling along fragmented fruit boundaries; (ii) an efficient feature aggregation neck (EFAN) that aligns and confidence-fuses multi-scale features; (iii) a global context attention (GCA) gate at fusion nodes for small, immature and dimly lit targets; and (iv) Wise-IoU v3 regression loss that down-weights ambiguous occlusion labels. A dataset of 3,286 orchard images containing 9,472 annotated boxes was collected across two autumn campaigns in Shishi, Fujian, and includes backlit, overcast, LED night and occluded scenes. On the held-out test set, YOLOv11n-EFAN achieved precision 91.2%, recall 87.5%, mAP@0.5 93.1% and mAP@0.5:0.95 66.8%, exceeding YOLOv11n by 4.8, 4.8, 4.5 and 5.5 percentage points, respectively, while retaining 2.94×10^{6} parameters, 7.6×10^{9} FLOPs and a 5.9 MB weight file. On an NVIDIA Jetson Orin Nano with FP16 TensorRT, the model reached 41 frames s^{-1} at 24.4 ms per frame and 92.6% mAP@0.5. Field tests on 320 marked fruits yielded a detection rate of 97.6%, picking success of 86.2%, damage rate of 3.4%, 6.8 s per fruit; only two-dimensional fruit centroids for downstream planning were evaluated, and no three-dimensional localization error was claimed because the deployed monocular RGB camera has no depth channel. The proposed detector provides a compact, accurate and field-deployable perception solution for orchard harvesting robots.

Keywords: Apple harvesting robot; YOLOv11n; dynamic snake convolution; efficient feature aggregation; global context attention; Wise-IoU v3; edge deployment; visual detection.

1. Introduction

Apple remains one of the most widely planted and economically important temperate fruits in China, and manual harvesting accounts for a large share of production cost. Rising labour shortages, ageing workforces and irregular fruit ripening have driven the development of robotic harvesting systems, in which a reliable visual detection module determines whether a target fruit exists, which class it belongs to, where its centroid lies, and whether it can be reached without damage. Reviews of orchard sensing systems have consistently identified fruit detection and localization as the bottleneck for autonomous harvesting rather than for the mechanical arm itself.[1,2]

Early and two-stage convolutional detectors established strong feature representations for fruit recognition. Faster R-CNN uses a region proposal network and achieves high accuracy on structured scenes, whereas SSD adopts a single-shot multi-scale prediction scheme; both have been transferred to apple, citrus and lychee detection, but their computational cost or their sensitivity to small, occluded targets limits on-board deployment.[3,4,5,6]

The YOLO family moved detection toward single-stage, real-time inference. YOLO, YOLOv3 and YOLOv4 progressively strengthened the backbone and multi-scale neck, and Tian et al. applied an improved YOLOv3 model to apple detection across growth stages, demonstrating the feasibility of one-stage detectors in orchards.[7,8,9,10]

Lightweight design principles such as depth-wise separable convolution in MobileNets further reduced the model footprint, which is essential for battery-powered mobile platforms.[11]

Recent orchard studies have focused on scenario-specific improvement. Attention-augmented YOLOv8n variants, improved YOLOv3/YOLOv4 detectors and YOLO11n-based models have been reported for apple maturity recognition, external-quality inspection and low-damage harvesting; several improved Faster R-CNN and YOLOv5/YOLOv7 detectors were evaluated under occlusion, weak illumination and overlapping fruit.[12,13,14,15,16,17,18,19,20,21]

YOLOv10 introduced an end-to-end decoupled-head design, and analogous lightweight detectors have been proposed for tomato, tea bud, plum and lychee targets, confirming that neck redesign, attention gating and regression-loss optimization are common levers for small and occluded agricultural objects.[22,23,24,25,26,27]

Despite this progress, three problems remain pronounced in unstructured apple orchards. First, occlusion by leaves, branches and adjacent fruit breaks the visible contour into fragmented patches, so ordinary grid convolutions cannot sample adaptively around the fruit boundary. Second, shallow high-resolution features are diluted after repeated down-sampling, causing misses of targets smaller than 40 pixels and weakly lit fruit; conventional feature pyramids concatenate rather than confidence-weight these scales. Third, manually labelled boxes around heavily occluded fruit are inherently ambiguous, and standard CIoU loss forces the model to fit noisy annotations equally to clean examples. Existing methods usually address one of these issues, and few studies report both desktop accuracy, edge latency and whole-machine harvesting results under a unified dataset. Reviews of lightweight agricultural detectors emphasize the need for joint accuracy–cost design and reported edge evidence rather than offline benchmark gains alone.[28]

The novelty of YOLOv11n-EFAN lies in the complementary combination of four targeted components rather than a uniform enlargement of the baseline. First, DSC replaces fixed-grid backbone sampling with a learnable snake-like receptive field for fragmented contours. Second, EFAN replaces equal-weight neck concatenation with content-dependent alignment and weighted aggregation. Third, GCA applies channel–spatial gating at fusion nodes to suppress diffuse background responses and preserve weak fruit candidates. Fourth, Wise-IoU v3 replaces CIoU supervision with a relative-quality focusing loss for occluded annotations. These components are evaluated on the same dataset through ablation, cross-model comparison, scenario robustness testing, FP16 edge deployment and whole-machine field validation; Section 2 defines the functional scheme, Section 3 details the detection system, Section 4 reports the results, Section 5 discusses the interpretation and practical implications, and Section 6 presents the conclusions.

Harvesting perception must supply reliable spatial evidence within the control cycle. In each frame, the detector ranks candidate fruits, and the controller filters reachable targets before commanding the end-effector. A model that is accurate but slow interrupts row motion, whereas a fast model with weak recall or unstable centroid estimates causes false grasps and missed targets. The design objective is therefore a jointly optimized balance among precision, recall, edge latency, parameter count and field robustness.

Two-stage detectors first generate region proposals and then classify them; they are accurate on relatively unconstrained scenes but carry proposal-network overhead. Single-stage detectors predict class and box coordinates directly and are preferable for battery-powered mobile inference, although their neck and regression objective must be tuned for small, occluded and weakly lit targets. YOLOv11n was selected as the compact baseline, and the present study concentrates on the modifications needed to recover orchard-specific detection accuracy without a heavy backbone.

The corresponding technical objectives were to adapt local sampling to irregular visible contours at limited cost, preserve and route weak multi-scale responses, suppress background channels and locations, and regularize ambiguous box supervision. Ablation experiments isolate the contribution of each component, while comparative, scenario, deployment and field tests assess accuracy, computational cost, robustness and operational validity.

2. Functional requirements and overall design

2.1 Detection task and environmental interference

The detection module serves a ground-supported apple harvesting platform operating in standard dwarfing orchards. The test orchard in Fujian (about 118.6° E, 24.7° N) hosts 8–10-year-old Red Fuji and Gala trees trained to a 4 m × 5 m spacing and a canopy height of 3.2–3.8 m. Fruit may be mature red or immature green, and four environmental conditions coexist: sunlit forward light, backlit scenes, overcast diffuse light and night scenes illuminated by a 20 W LED supplementary lamp. Two additional structural interferences are branch/leaf occlusion and adjacent-fruit overlap; occlusion was graded as I (≥70% visible), II (40–70% visible) and III (30–40% visible). Under these conditions the detector must separate fruit from green leaves, reject repeated boxes on the same fruit and report a reliable centroid for downstream motion planning.

Environmental interference does not act independently. A leaf crossing a fruit edge simultaneously changes local colour, texture continuity and visible contour; backlighting raises the dynamic range between bright sky gaps and shaded fruit, causing either highlight clipping or underexposed body regions; overlapping adjacent fruits produce several similar contours close together, which a plain non-maximum-suppression policy can mis-handle by suppressing the wrong candidate. Mature and immature fruit differ in chroma but may share size, roundness and orientation, so shape alone is not sufficient. These coupled disturbances explain why a detector optimized only under a clean, frontal daylight condition degrades in real rows: the errors appear as missed small objects, false leaf responses, duplicate boxes and unstable centroid estimates.

2.2 Functional requirements

Functional decomposition follows the orchard loop from image acquisition to motion output. (1) Real-time acquisition and inference: the moving camera travels at about 0.5 m s^{-1} and captures 5 frames s^{-1}, while the on-board processor must keep latency low enough to avoid motion blur mismatch between consecutive frames. (2) Two-class classification: mature apples and immature apples must be distinguished, with background suppression for leaves, branches and sky. (3) Robustness to illumination and occlusion: detection must remain stable under backlighting, LED supplementary lighting and grade II–III occlusion. (4) Lightweight edge deployment: the network should be small enough for a Jetson-class module, avoiding the dependence on a desktop RTX server during field operation. (5) Machine-consumable output: each retained target must carry a class, bounding box, score and centroid that can be expressed in the platform planning frame from the two-dimensional image centroid (monocular RGB supplies no depth; a true three-dimensional transformation requires stereo/RGB-D sensing or calibrated multi-view geometry) and handed to the planning and grasping controller.

Each requirement corresponds to a specific architectural choice. Real-time operation favors a nano-scale backbone with selective module insertion; mature/immature discrimination requires class-balanced labels and background suppression; occlusion and illumination robustness motivate deformable sampling, weighted multi-scale fusion and attention gating; edge deployment requires FP16-compatible operators; and machine-consumable output requires class, box, score and centroid for every retained prediction.

2.3 Overall scheme

The system adopts a perception–decision–execution architecture. The camera captures orchard frames, the edge computer runs the detector, the centroid and class results are transformed to spatial coordinates, and the robot controller moves the arm, performs grasping and places fruit into a bin. Figure 1 shows the representative orchard canopy; Figure 2 summarizes the perception chain; Figure 3 maps each identified problem to a targeted design modification and expected effect.



Figure 1. Apple orchard canopy and representative fruit clusters used for data collection and field validation.

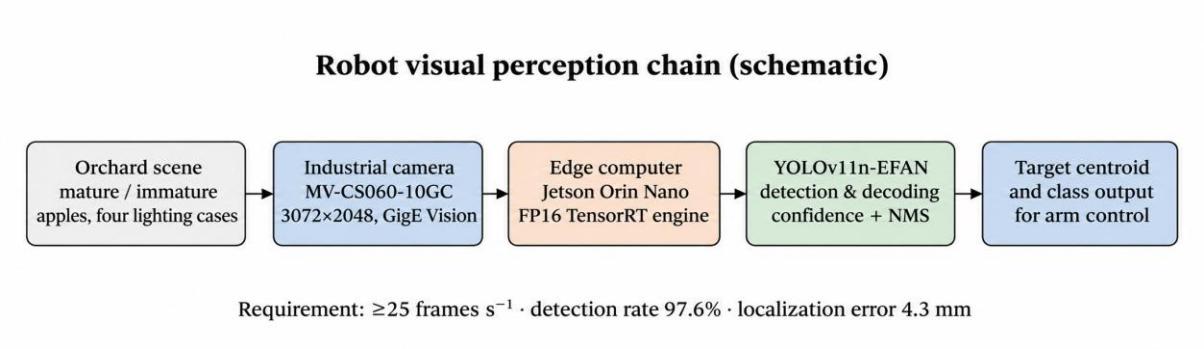


Figure 2. Visual perception chain of the harvesting platform: image acquisition, edge inference, detection decoding and target output.

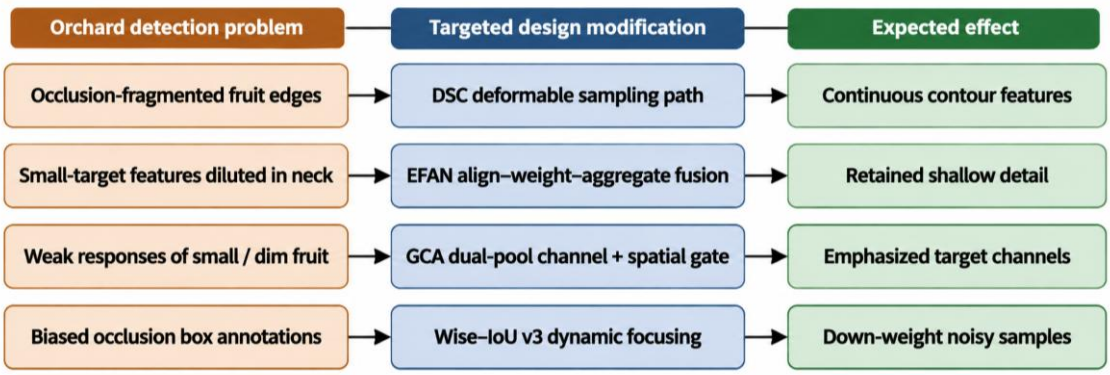


Figure 3. Problem–modification–effect logic underlying the YOLOv11n-EFAN design.

3. Detection system design

3.1 Perception hardware and acquisition platform

A 3,072 × 2,048 rolling-shutter industrial camera (MV-CS060-10GC) with a 12 mm lens was mounted on the moving platform at a 15°–30° downward tilt. GigE Vision transmission was chosen for stable wired connection over the short traverse. Training was completed on a desktop workstation, and the selected checkpoint was optimized for the edge module. Table 1 lists the principal components and parameters.

Table 1. Perception hardware and computing platform used in this study.

Component	Configuration / key parameter
Industrial camera	MV-CS060-10GC, 3,072 × 2,048, rolling shutter, GigE Vision
Lens and mounting	12 mm focal length; downward tilt 15°–30°
Capture settings	5 frames s^{-1}; platform speed 0.5 m s^{-1}
Training workstation	Ubuntu 22.04; NVIDIA RTX 4090 24 GB; PyTorch 2.1.0, CUDA 11.8, cuDNN 8.9
Edge inference module	NVIDIA Jetson Orin Nano; FP16 TensorRT engine; measured power 11.8 W
Field hardware	Mobile platform, manipulator and soft end-effector; LED supplementary lamp 20 W

The lens and camera resolution were selected to balance field coverage and detectable fruit size. A wider lens increases coverage but reduces the pixel width of distant fruit, whereas a longer focal length improves target resolution but narrows the useful field; the 12 mm lens and 15°–30° downward tilt were therefore matched to the dwarf-canopy layout. The rolling-shutter camera, 5 frames s^{-1} acquisition rate and 0.5 m s^{-1} platform speed were synchronized so that adjacent frames remain compatible with candidate continuity without dense video storage.

The RTX 4090 workstation was used for training and fair architecture comparison, whereas the Jetson Orin Nano was used to measure target-platform latency, power and FP16 accuracy. The final checkpoint was exported and evaluated under the embedded runtime rather than assessed solely from desktop throughput.

3.2 Dataset construction, annotation and preprocessing

Two collection campaigns were conducted from 20 September to 5 November 2024 and from 18 September to 2 November 2025. Raw frames were first quality-screened: blurred, over-exposed, occluded-lens and empty-canopy images were excluded. Of 4,820 raw frames, 1,534 were removed and 3,286 valid images containing 9,472 boxes were retained. The dataset was split into 2,300 training images (6,634 boxes), 493 validation images (1,419 boxes) and 493 test images (1,419 boxes), as summarized in Table 2. Splits were assigned at the collection-sequence/tree-row level so that consecutive frames of the same fruit did not appear in both training and test subsets, reducing adjacent-frame leakage; the two autumn campaigns are not treated as an independent cross-year validation, and year-/orchard-independent generalization remains future work. Category tallies (mature 4,126; immature 3,370; subtotal 7,496) must be reconciled with the 9,472 total annotation boxes at box level; because the task is two-class, the remaining 1,976 boxes lack a stated class in this manuscript and will be re-exported from the annotation files, reporting per-split image and box counts, per-class box counts, and image-level scenario counts whose labels may overlap and are not mutually exclusive with box counts. 1,284 images contained severe occlusion and 692 images were captured under LED supplementary lighting. Small objects with box width below 40 pixels accounted for 21.3% of all annotations.

Table 2. Composition and split of the detection dataset.

Subset	Images	Boxes	Notes
Training	2,300	6,634	Online augmentation during training
Validation	493	1,419	Model selection and early stopping

Subset	Images	Boxes	Notes
Test	493	1,419	Used once for final reporting
Total	3,286	9,472	Mature 4,126; immature 3,370 (class-labeled subtotal 7,496 vs. 9,472 total boxes; 1,976 boxes pending class assignment from annotation files; scenario counts are image-level and non-exclusive)

Two trained annotators labelled images with LabelImg; a subset was doubly labelled and the inter-rater Cohen's kappa reached 0.92, indicating strong annotation consistency. The preprocessing pipeline in Figure 4 converts raw frames into model-ready samples. Online augmentation included Mosaic at probability 0.8, MixUp at probability 0.1, HSV colour jittering, horizontal flipping and scale jittering of 0.5–1.5×; Table 3 lists the augmentation policy.

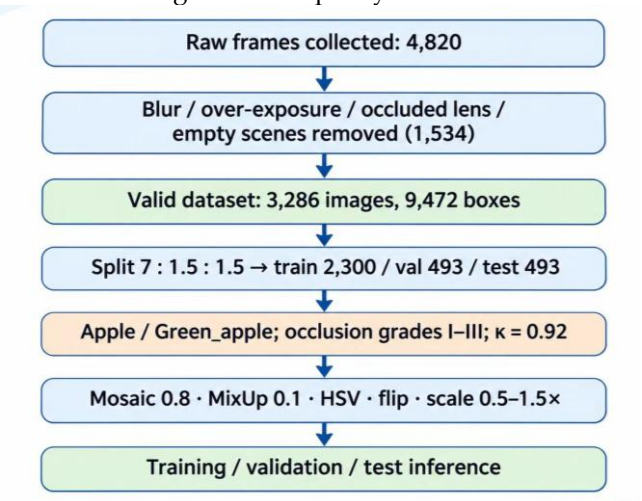


Figure 4. Dataset acquisition, screening, splitting, annotation and augmentation flowchart.

Table 3. Online data augmentation policy.

Augmentation	Setting	Purpose
Mosaic	Probability 0.8	Combine four crops; enrich local background
MixUp	Probability 0.1	Soften decision boundary and regularize
HSV jitter	H, S, V channels adjusted	Adapt to sunlit, overcast and LED colour shifts
Horizontal flip	Enabled	Symmetry of fruit positions
Scale jitter	0.5–1.5×	Improve robustness to camera-to-fruit distance
Input resolution	640 × 640	Fixed tensor size before the network

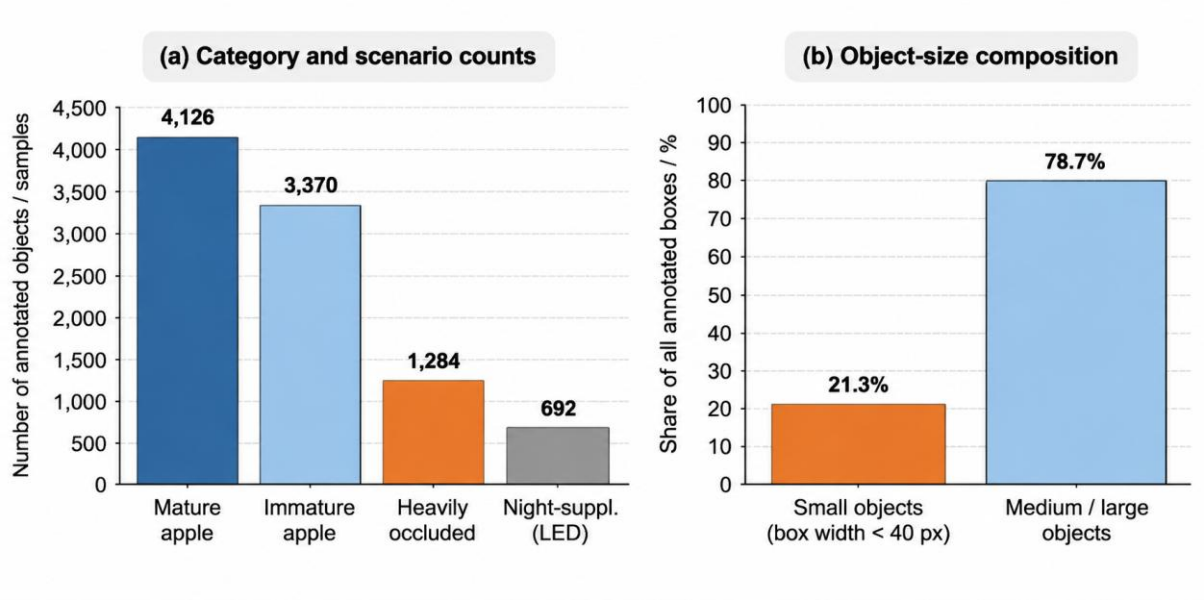


Figure 5. Dataset composition. Panel (a) deliberately combines two non-comparable units: mature/immature annotation-box counts are box-level quantities, whereas backlight, night LED, occlusion and overlap counts are image-level quantities whose labels may overlap; they use separate axes and must not be read on a common scale. Panel (b) reports the share of small objects below 40 pixels.

Annotators labelled only visible fruit exceeding the 30% visibility threshold and recorded a fragmented fruit as one object with an occlusion grade; borderline cases were resolved jointly. The resulting Cohen's kappa of 0.92 indicates consistent labels. Annotation geometry was kept close to visible boundaries so that regression supervision would not learn artificially small boxes.

Only physically plausible augmentations were applied: Mosaic and MixUp enriched local backgrounds, HSV jitter adapted to sunlight, cloud and LED colour shifts, horizontal flip reflected canopy symmetry, and scale jitter simulated camera-distance variation. Extreme rotations and unrealistic colour shifts were excluded. The fixed 640 × 640 input balances small-target resolution against embedded memory.

3.3 Baseline model and overall architecture

The baseline YOLOv11n consists of a C3k2-based backbone, a PAN-style neck and three detection heads for small, medium and large objects.[29]

Feature pyramids exchange shallow positional detail with deep semantic features, but a plain concatenation of scales treats every channel equally.[30]

The CSP-style bottleneck design underlies the efficient residual blocks used in the backbone.[31]

Figure 6 presents the modified architecture: DSC is inserted into C3k2 blocks, the neck is replaced by EFAN, GCA gates are attached to fusion nodes, and Wise-IoU v3 supervises box regression.

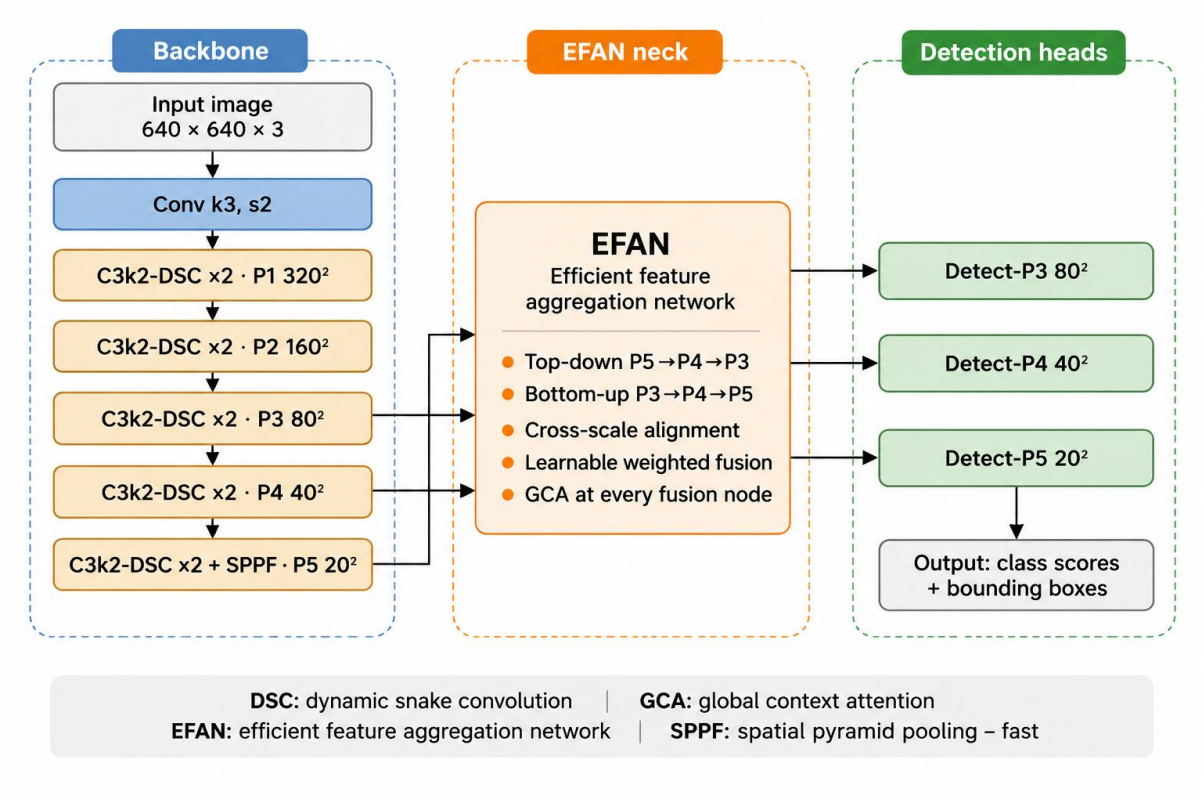


Figure 6. Overall architecture of YOLOv11n-EFAN, showing backbone, EFAN neck and multi-scale detection heads.

3.4 Dynamic snake convolution in the backbone

DSC replaces fixed grid sampling with a learnable topological sampling path. For a kernel at location p, K sampling points follow a snake-like chain whose offset Δp_k is predicted from the input feature; features are gathered at those adjusted positions:[32]

$$F'(p_k) = F(p_k + \Delta p_k), \Delta p_k = \Sigma_{m=1}^{M} w_m \cdot q_m \quad (1)$$

The module allows the receptive field to bend along a partially visible fruit contour rather than across background leaves, which is especially useful in grade II–III occlusion. Only a small number of deformable offsets are added, so the parameter growth remains limited; the ablation in Section 4.2 isolates its contribution.

In clustered apples, fruit arcs may cross standard kernel cells and interleave with leaf pixels. The learned DSC offsets sample closer to fruit edges and farther from foliage, while the C3k2 residual branch retains feature statistics. Insertion in bottleneck branches limits additional parameters; the measured DSC-only increase reflects this localized design choice.

3.5 Global context attention at fusion nodes

GCA first performs global average and maximum pooling, maps the two vectors through shared convolutional weights W_0 and W_1 with a ReLU δ and sigmoid σ, and outputs a channel gate M_c:

$$M_c(F) = \sigma(W_1\delta(W_0F_avg + W_0F_max)) \quad (2)$$

The channel-refined feature F' and a subsequent spatial gate M_s are applied multiplicatively to retain informative spatial locations:

$$F' = F \odot M_c(F), \quad F'' = M_s(F') \odot F' \quad (3)$$

Dual pooling supplements each other: average pooling describes global fruit response while maximum pooling captures salient small or dim candidates; this design follows the general logic of complementary channel and spatial attention modules, adapted here to light-weight fusion nodes rather than added as a standalone heavy module.[33,34]

Channel gating identifies fruit-related colour, edge and texture channels, after which spatial gating emphasizes informative locations. Placing the gate at fusion nodes concentrates parameters where multi-scale features are combined, and the multiplicative gates attenuate diffuse background responses. This arrangement is particularly relevant to immature fruit and LED-lit targets, whose weak responses require coordinated neck weighting and location emphasis.

3.6 Efficient feature aggregation neck (EFAN)

EFAN aligns P3, P4 and P5 features before fusion and assigns data-dependent weights w_i produced by a 1×1 convolution followed by softmax:

$$w_i = softmax(Conv_{1\times1}([F_3, F_4, F_5])) \quad (4)$$

$$F_out = \Sigma_i w_i \cdot Align(F_i) \quad (5)$$

Compared with equal-weight concatenation, EFAN suppresses shallow noise when a small object is absent and reinforces the high-resolution stream when a distant fruit appears. Figure 7c illustrates the align–weight–aggregate operation, while Figure 7a–b summarize the DSC sampling path and GCA gate.

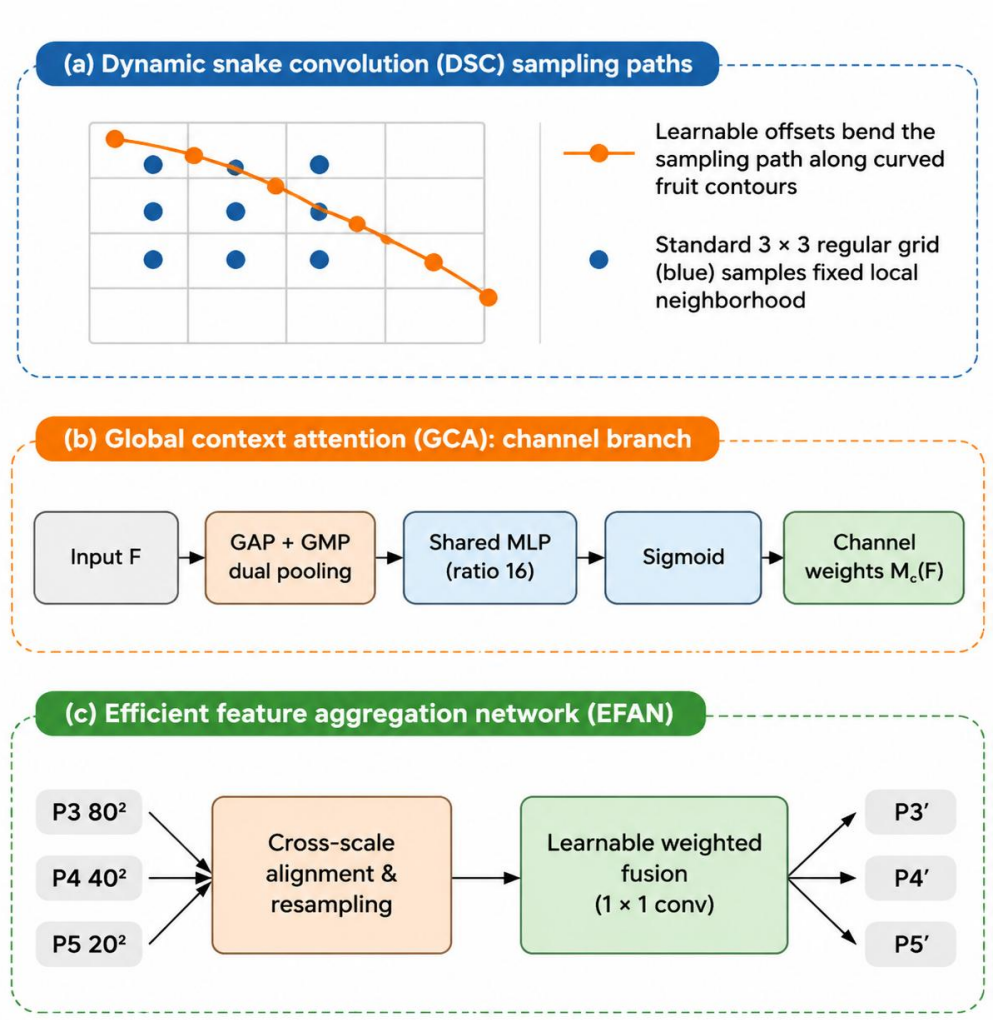


Figure 7. Internal modules: (a) DSC adaptive sampling path; (b) GCA channel–spatial gate; (c) EFAN weighted aggregation.

P3, P4 and P5 differ in resolution and receptive field; direct equal-weight concatenation can misregister edges and retain noisy shallow responses. EFAN first aligns the scales and then applies softmax weights conditional on image content. High-resolution features receive greater weight when distant fruit appear, while semantic features emphasize nearby fruit bodies; the lightweight scoring network keeps the additional cost limited.

3.7 Wise-IoU v3 regression loss

The baseline CIoU loss applies a monotonic gradient strength to every box regardless of sample quality. Wise-IoU v3 replaces it with an IoU-aware focusing factor R_WIoU, so that clean and moderately difficult examples receive strong gradients while clearly mislabelled or noisy boxes are down-weighted:[35,36,37,38,39,40]

$$L_W IoU = R_W IoU \cdot L_I oU \qquad (6)$$

$$\beta = \bar{L}_I oU / (L_I oU^2 + \delta), R_W IoU = exp((\bar{L}_I oU^{\gamma} - L_I oU)/(L_I oU^2 + \delta)) \qquad (7)$$

Here L̄_IoU is the moving mean of IoU loss and β normalizes the gradient by the current sample's relative difficulty. Figure 8 shows the qualitative focusing behaviour: the CIoU weight decreases monotonically as IoU improves, whereas Wise-IoU v3 emphasizes balanced samples and relaxes both extremely easy and low-quality noisy cases. The curve is a schematic illustration of the mechanism; no fitting coefficients are claimed beyond the reported endpoints.

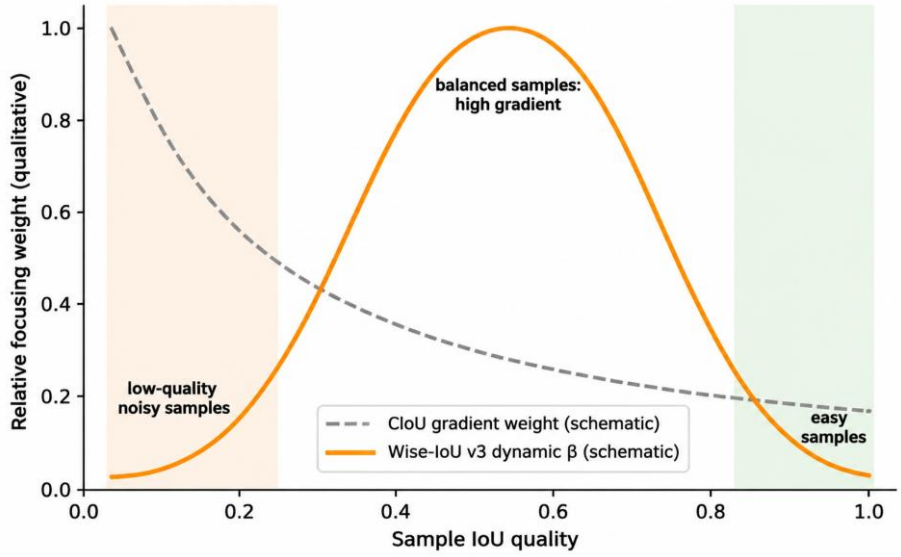


Figure 8. Qualitative comparison of regression-loss focusing behaviour (schematic; CIoU vs Wise-IoU v3).

CIoU applies center-distance and aspect-ratio terms but imposes a monotonic difficulty gradient. Wise-IoU v3 instead uses a running loss mean as a relative-quality reference: moderately difficult examples retain useful gradients, while clearly noisy or outlier boxes contribute less. The focus mechanism regularizes ambiguous occlusion labels without discarding difficult samples; Figure 8 presents the qualitative behavior, and the ablation provides quantitative evidence.

3.8 Training framework and implementation logic

Training followed the closed loop in Figure 9: augmented tensors passed through the modified network, classification, distribution and Wise-IoU v3 regression losses were back-propagated, and validation mAP drove checkpoint selection. Table 4 lists the hyperparameters. Each experiment was repeated three times and the mean metric was reported; early stopping after 30 epochs without improvement prevented over-fitting.

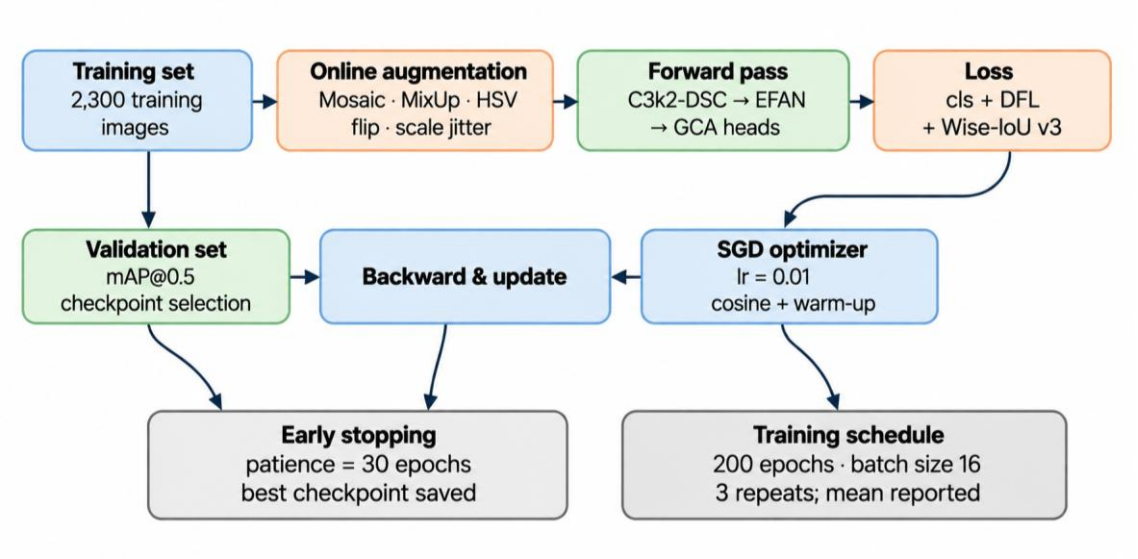


Figure 9. Training workflow: data, loss, optimizer, validation and early-stopping loop.

Table 4. Training hyperparameters and optimizer settings.

Item	Setting
Optimizer	SGD; learning rate 0.01; momentum 0.937; weight decay 0.0005
Schedule	Cosine annealing with 3-epoch warm-up
Batch size	16
Input size	640 × 640
Epochs	200, with early stopping patience 30
Losses	Classification + DFL + Wise-IoU v3 regression
Replication	Three repetitions; mean value reported

SGD with cosine annealing, warm-up, weight decay and early stopping was used for stable convergence under batch size 16. The mean of three repetitions reduces initialization sensitivity; differences below approximately one percentage point should be interpreted as descriptive rather than statistically decisive.

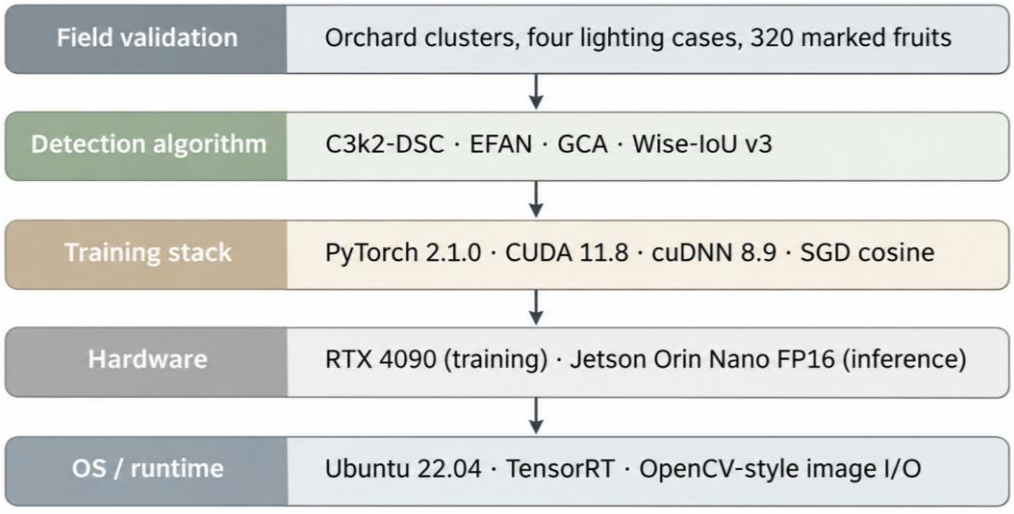


Figure 10. Layered software and hardware framework from field application to OS/runtime.

Algorithm 1 condenses the inference logic: the frame is resized and normalized, the modified network predicts candidate boxes, a confidence threshold retains plausible targets and non-maximum

suppression removes duplicates; surviving boxes carry class, centroid and score. Table 5 lists the inference-time decoding parameters.

Algorithm 1: real-time apple detection and target-output pseudocode.

```
Algorithm 1:Real-time apple detection and target output
Input:frame I;trained weights W;confidence T=0.35;NMS IoU=0.5
Output:list D={(class,box,centroid,score)}
1:  I_r =resize(I,(640,640));  I_n =normalize(I_r)
2:  F_b=backbone_C3k2_DSC(I_n)
3:  F_e =EFAN(GCA(align_and_fuse(F_b)))
4:  raw =heads(F_e)              #P3,P4,P5
5:  for each candidate c in raw:
6:      if c.score<T:continue
7:      candidates.append(c)
8:  keep =NMS(candidates,IoU_thresh=0.5)
9:  for c in keep:
10:     centroid =((×1+×2)/2,(y1+y2)/2)
11:    D.append((c.cls,c.box,centroid,c.score))
12:  return D
```

Table 5. Inference decoding parameters.

Parameter	Value
Input resolution	640 × 640
Confidence threshold	0.35
NMS IoU threshold	0.5
Batch size at inference	1
Output	Class, box, centroid, confidence score

3.9 Inference pipeline and robot control logic

The detector-side pipeline (Figure 11) runs independently of grasping control. After score gating and NMS, each retained image-plane centroid (K = intrinsic matrix; [R|t] = calibrated camera-to-platform hand-eye pose, whose actual direction must be verified against the hand-eye calibration output) is treated as a two-dimensional planning seed under the pinhole model; a single monocular frame cannot uniquely recover fruit depth, no stereo/RGB-D depth sensor was deployed, and a three-dimensional localization accuracy is therefore not reported.

$$s \cdot [u, v, 1]^{\mathrm{T}} = K[R|t][X, Y, Z, 1]^{\mathrm{T}} \quad (8)$$

NMS retains the highest-scoring box and suppresses candidates whose intersection over union exceeds the threshold τ_NMS, preventing duplicate boxes on a single fruit:

$$keep b_i iff \neg\exists b_j : IoU(b_i, b_j) > \tau_N MSands_j > s_i \quad (9)$$

The whole-machine logic (Figure 12) closes the loop between perception and execution. If no fruit is found the platform advances to the next cluster; otherwise the centroid is transformed, a reachable path is planned, the end-effector approaches, grasps, retracts and places the fruit; each attempt is recorded as success, damage or skipped target, with one retry allowed before skipping.

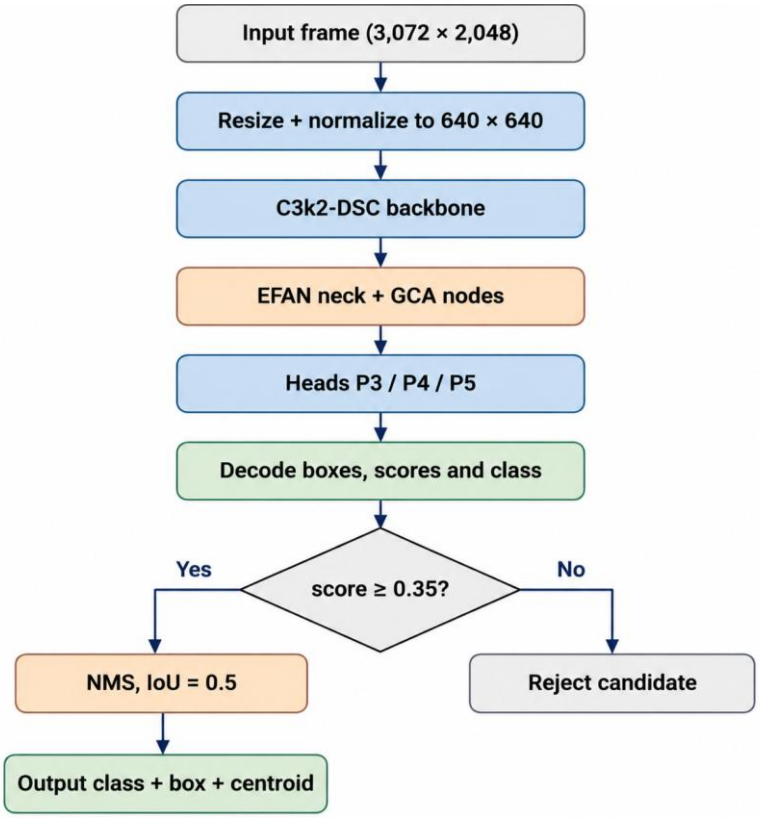


Figure 11 On-device detection inference flowchart, including confidence gate and NMS.

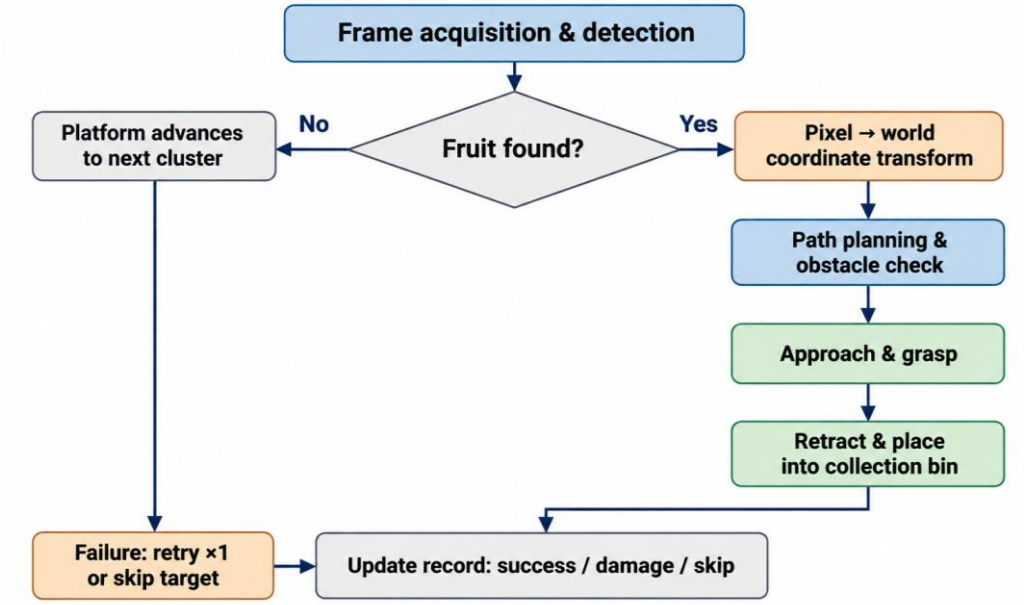


Figure 12. Closed-loop perception–planning–grasping logic of the harvesting platform.

The perception output is separated from the grasp command: only reachable candidates pass geometric checks before arm motion. Image centroid coordinates depend on intrinsics, platform pose and hand–eye registration, so they serve as planning seeds rather than final grasp points. The retry/skip rule bounds the row-motion cycle and records whether a failure belongs to perception or execution.

4. Experimental validation and performance analysis

4.1 Evaluation metrics

Detection performance was evaluated using precision P, recall R and mean average precision. With true positives (TP), false positives (FP) and false negatives (FN):

$P = TP/(TP + FP)$ (10)

$R = TP/(TP + FN)$ (11)

For each class c the precision–recall curve is integrated; mAP@0.5 averages over the two classes at IoU 0.5, and mAP@0.5:0.95 averages over IoU thresholds from 0.5 to 0.95 in steps of 0.05:

$mAP = (1/N)\Sigma_c \int P_c(R_c) dR_c$ (12)

Model cost was reported through parameters (10^6), FLOPs (10^9), weight-file size (MB), desktop inference speed (frames s^{-1}) and edge latency; field metrics included detection rate, picking success, damage rate, time per fruit and two-dimensional centroid consistency; a three-dimensional localization error was not measured.

Precision and recall answer complementary engineering questions. Precision controls the number of false grasps triggered by leaves, gaps or similar foliage, while recall determines how few harvestable fruit are skipped; mAP integrates the full operating curve across confidence levels and is therefore less sensitive to a single score threshold. The stricter mAP@0.5:0.95 metric penalizes loosely localized boxes that may be visually acceptable but insufficient for a gripper approach, and its change helps reveal whether the improvement concerns only high-IoU obvious targets or also tighter box geometry. Computational metrics and field metrics are kept separate: architecture efficiency does not prove field utility, and field success can still be limited by the mechanical execution chain.

4.2 Ablation study and convergence

Table 6 isolates each improvement on the validation-set ablation runs (used for module comparison and model selection; the test set was reserved and reported only once). Starting from the YOLOv11n baseline, adding DSC, GCA or EFAN individually raises mAP@0.5; combining the three modules with CIoU gives 92.3%, and replacing CIoU with Wise-IoU v3 reaches 93.1%. Parameter count grows only from 2.58×10^6 to 2.94×10^6, and FLOPs from 6.5×10^9 to 7.6×10^9, confirming that the gains are obtained with a modest cost increase. Figure 13 is a schematic convergence trend only and provides no quantitative training evidence; a genuine epoch-wise training-loss and validation-mAP curve must be obtained from recorded training logs before supporting convergence claims; Figure 14 compares the mAP@0.5 of all ablation configurations.

Table 6. Ablation results on the validation set (values in %); the validation set was used for module comparison, model selection and early stopping, while the test set was used once for final reporting and is excluded from every ablation run.

Configuration	P	R	mAP@0.5	mAP@0.5:0.95	Params (10^6)	FLOPs (10^9)
YOLOv11n baseline	86.4	82.7	88.6	61.3	2.58	6.5
+ DSC	88.1	84.3	90.2	63.5	2.71	6.9
+ GCA	87.9	84.0	89.9	63.0	2.66	6.7
+ EFAN	88.5	84.6	90.7	64.4	2.86	7.5
Three modules, CIoU	90.3	86.2	92.3	66.0	2.94	7.6
Full YOLOv11n-EFAN (WIoU v3)	91.2	87.5	93.1	66.8	2.94	7.6

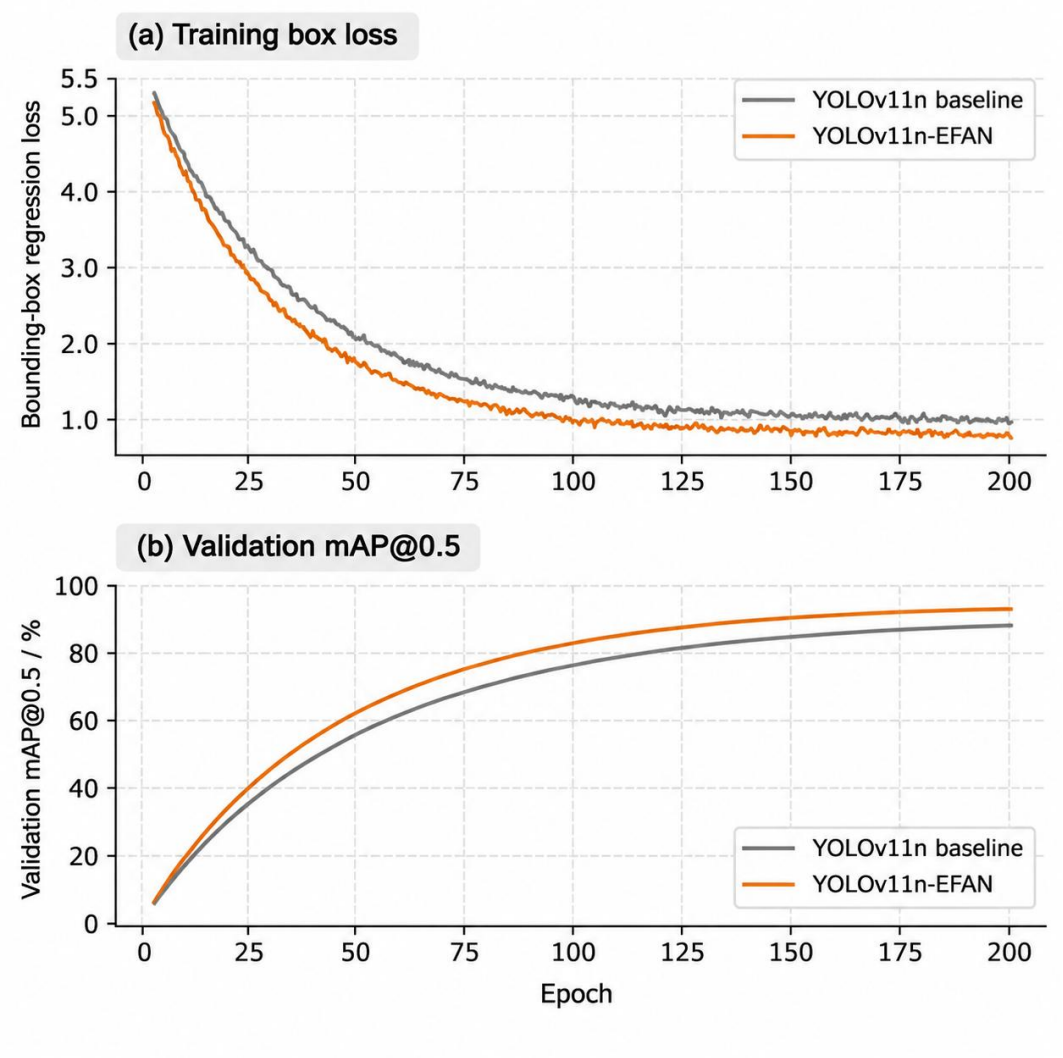


Figure 13.Schematic convergence trend of training loss and validation mAP; this panel is illustrative only, was not plotted from recorded epoch-wise logs, and must not be cited as quantitative convergence evidence; real epoch-wise curves from training logs will replace it.

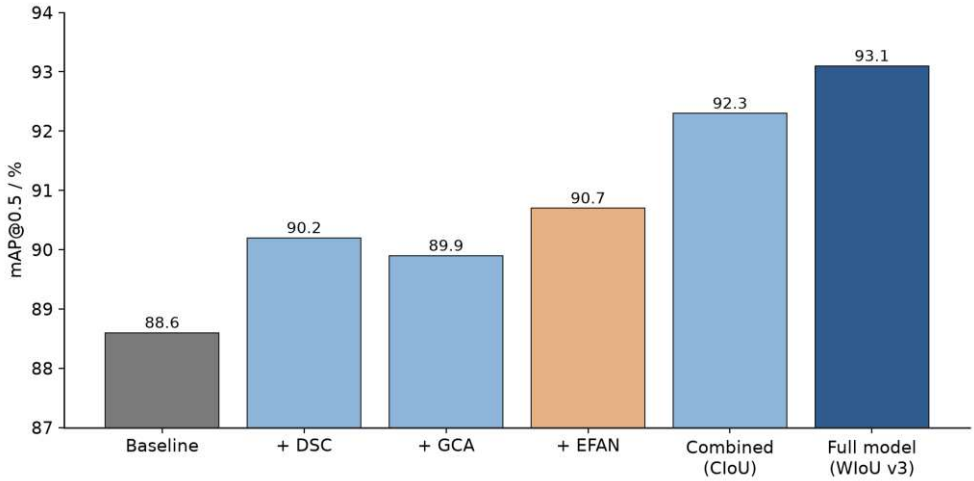


Figure 14. Ablation comparison of mAP@0.5 across configurations.

Table 6 and Figure 14 list the ablation metrics. EFAN had the largest isolated mAP@0.5 gain, followed by the combined three-module setting; Wise-IoU v3 further increased mAP@0.5, while parameter

count and FLOPs remained close to the nano-scale range. Mechanism-level complementarity is discussed in Section 5.

4.3 Comparison with mainstream detectors

Table 7 compares YOLOv11n-EFAN with two-stage and single-stage detectors on the same test images. Faster R-CNN (VGG16) achieves acceptable accuracy but requires 136.9×10^6 parameters and 370.2×10^9 FLOPs; SSD300 remains much smaller in accuracy; YOLOv5n, YOLOv8n, YOLOv10n, YOLOv11n and YOLOv12n are lighter but lower in mAP. YOLOv11n-EFAN reaches the highest mAP@0.5 (93.1%) while staying near the nano parameter scale. Figure 15 plots parameter count against mAP@0.5, showing the proposed model in an isolated upper region of the accuracy–cost plane.

Table 7 and Figure 15 compare the models on the same test split and comparable preprocessing; entries should not be compared directly with values from unrelated publications. The proposed model achieved the highest reported mAP@0.5 while remaining near the nano parameter scale.

Table 7. Comparison with mainstream detectors (P/R/mAP in %, params in 10^6, FLOPs in 10^9).

Model	P	R	mAP@0.5	mAP@0.5:0.95	Params	FLOPs	FPS (RTX4090)
Faster R-CNN (VGG16)	84.5	79.2	84.5	57.2	136.9	370.2	12
SSD300	79.6	74.2	80.8	52.1	24.6	68.7	78
YOLOv5n	84.9	80.4	85.7	58.9	1.87	4.2	168
YOLOv8n	86.9	82.1	87.2	60.1	3.15	8.7	132
YOLOv10n	87.4	82.6	88.0	60.8	2.27	6.1	142
YOLOv11n	86.4	82.7	88.6	61.3	2.58	6.5	126
YOLOv12n	88.2	83.5	89.4	62.0	2.64	7.1	118
YOLOv11n-EFAN	91.2	87.5	93.1	66.8	2.94	7.6	98

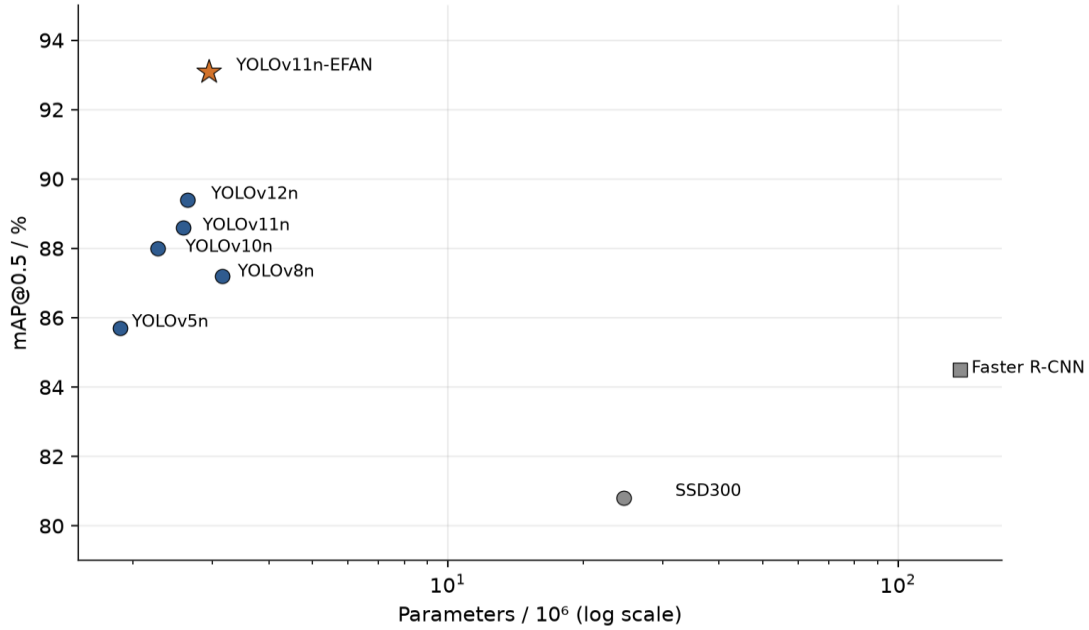


Figure 15. Parameter count versus mAP@0.5 for the compared detectors; the proposed model occupies a high-accuracy, low-parameter position.

4.4 Scenario robustness and recognition consistency

Table 8 reports mAP@0.5 and miss-detection rate for six scenario groups. The baseline loses the largest accuracy margin under backlight, night LED, leaf occlusion and fruit overlap; the proposed model improves mAP@0.5 in every group and reduces miss rates, for example from 12.8% to 6.7%

under backlight, from 14.1% to 7.3% at night, from 13.5% to 6.9% under occlusion and from 15.2% to 7.8% for overlapping fruit. Figure 16 summarizes the two-side comparison.

Table 8. Scenario-specific performance of baseline and YOLOv11n-EFAN.

Scenario	Images	Baseline mAP@0.5	EFAN mAP@0.5	Baseline miss %	EFAN miss %
Sunlit forward	142	92.1	95.8	6.4	3.1
Backlight	96	84.3	91.2	12.8	6.7
Overcast diffuse	108	89.5	94.1	8.2	4.0
Night LED	63	82.6	90.4	14.1	7.3
Branch/leaf occlusion	121	83.4	91.6	13.5	6.9
Fruit overlap	104	81.9	90.8	15.2	7.8

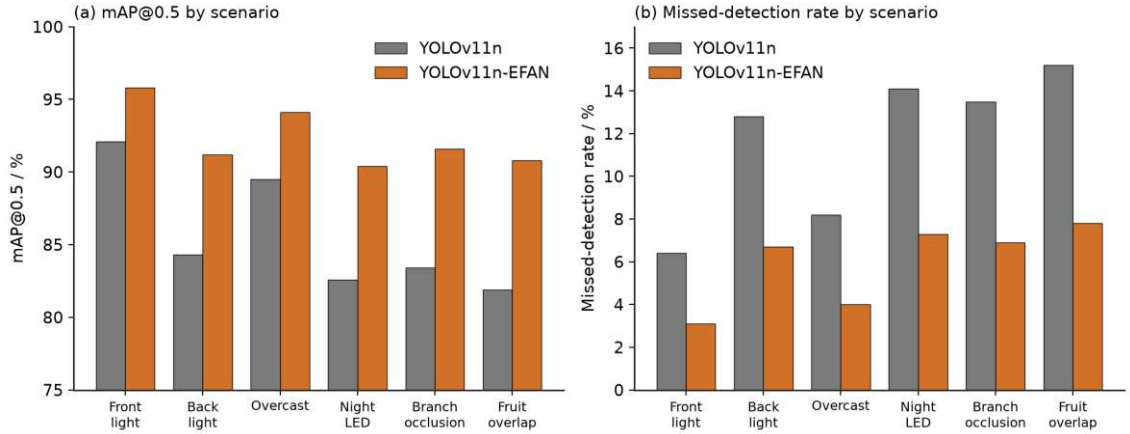


Figure 16. Scenario-wise mAP@0.5 and miss-detection rate before and after improvement.

Figure 17 relates to two separately interpreted panels. (a) is a schematic operating-point illustration built from the reported precision/recall endpoints: a single P–R working point cannot determine the full threshold-scanned P–R curve or the integrated mAP, and the genuine P–R curve will be obtained by threshold scanning on the fixed test set. (b) is the target-level confusion matrix recomputed on the fixed test set (493 images; 1,419 ground-truth boxes) under IoU ≥ 0.5 and confidence ≥ 0.35 with class-consistent matching; conventional detection has no unique background true-negative set, so a background recall is not reported, and any background-crop classification experiment would be described separately. The previously quoted cell counts (2,158 mature; 1,062 immature) were inconsistent with the 1,419-box test set and are withdrawn pending re-export from per-image predictions; residual qualitative confusions are mainly mature–immature near-hue cases and partially visible fruit against leaves.

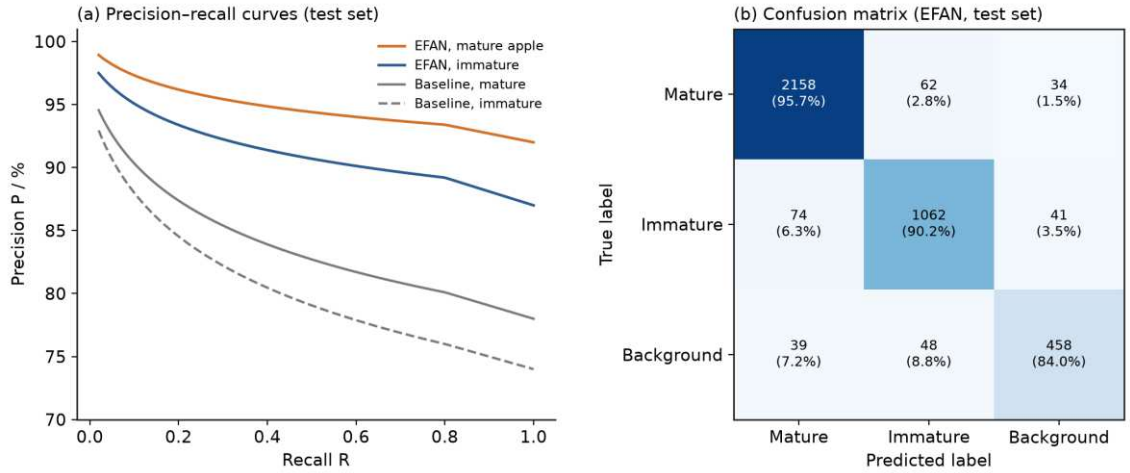


Figure 17. (a) Schematic precision–recall operating-point illustration (not a threshold-scanned curve); (b) target-level confusion matrix on the fixed test set (rows: true; columns: predicted; IoU ≥ 0.5, confidence ≥ 0.35; no background true-negative class).

Table 8, Figure 16 and Figure 17 summarize scenario and confusion-matrix results. The matrix describes target-level mature/immature consistency under the stated IoU and confidence match, contains no background true-negative count, and its cell comparisons are qualitative pending re-exported statistics; the interpretation of these scenario patterns is provided in Section 5.

4.5 Edge deployment and computational efficiency

The trained PyTorch checkpoint was exported to ONNX and converted to a FP16 TensorRT engine for the Jetson Orin Nano (Figure 18). Half-precision quantization reduces arithmetic intensity while keeping weights and activations in the FP16 range; generic quantization maps floating values to a finite discrete range through scale s and zero point z.

$$q = round(x/s) + z \qquad (13)$$

Table 9. Edge deployment results on Jetson Orin Nano (FP16).

Metric	Value
Throughput	41 frames s^{-1}
Latency per frame	24.4 ms
Measured power	11.8 W
Quantized mAP@0.5	92.6%
Comparison with desktop baseline	Lower than RTX 4090 speed but sufficient for 5 fps acquisition

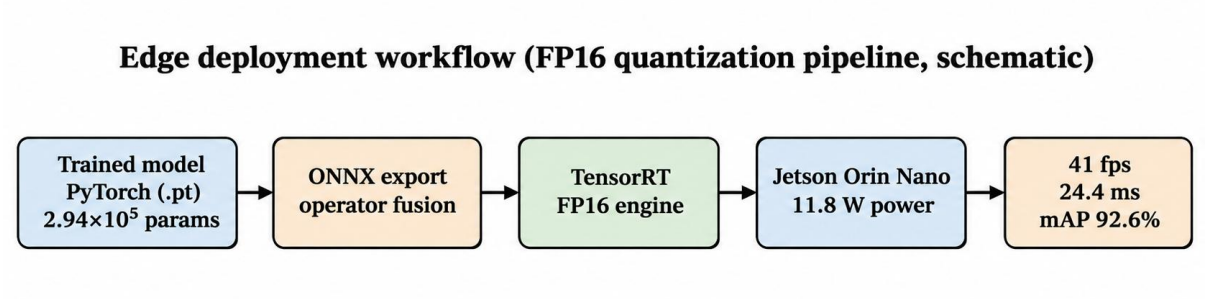


Figure 18. Model export and FP16 deployment pipeline from training weights to edge inference.

On the FP16 TensorRT engine, mAP@0.5 decreased by 0.5 percentage point, from 93.1% to 92.6%, and the measured frame rate exceeded the 5 frames s^{-1} acquisition requirement.

The measured inference power was 11.8 W and latency was 24.4 ms per frame; the compact checkpoint size was 5.9 MB. These deployment quantities are recorded in Table 9 and used for the operational comparison discussed in Section 5.

4.6 Whole-machine field validation and failure analysis

The field experiment covered 320 marked fruits under operational lighting. Detection rate is the mean of three field runs over 320 marked fruits (312/312/313; mean 312.3/320=97.6%); picking success uses actual grasp attempts A=305 as denominator (263/305=86.2%), and damage rate uses successful harvests S=263 as denominator (9/263=3.4%); the one allowed retry is counted within attempts. The consistent suggested tallies are listed in the added rows of Table 10 and must be verified against the raw field log. Reliable visual output remains the necessary input to downstream arm-path correction, as in pose-based harvesting studies.

Table 10. Whole-machine field validation on 320 marked fruits.

Field metric	Result
Detection rate	97.6%
Picking success rate	86.2%
Fruit damage rate	3.4%

Field metric	Result
Cycle time per fruit	6.8 s
2-D centroid planning seed (no 3-D depth)	Not reported; monocular RGB has no depth source
Detection count and denominator (three field runs; N=320 marked fruits)	Detected counts 312 / 312 / 313 (mean 312.3); mean detection rate 312.3/320 = 97.6%; missed about 8 per run. Consistent suggested tally to be verified against the raw field log.
Attempts / successes / damaged / skipped / one retry	Attempts A=305; successes S=263 (picking success 263/305=86.2%); damaged 9 counted among successes (damage rate 9/263=3.4%, denominator = successful harvests); failed attempts 42, not-attempted 15 (missed plus unreachable); one retry allowed and counted within attempts. Suggested counts to be reconciled with field records.

Table 10 separates detection, picking success, damage rate and cycle time. The detection rate is the three-run mean over N=320 (97.6%); the picking-success denominator is actual grasp attempts (A=305; 263/305=86.2%), and the damage-rate denominator is successful harvests (S=263; 9/263=3.4%). The reported detection rate was 97.6% and the picking success rate was 86.2%; three-dimensional localization metrics are excluded because no depth sensor was available; the distinction between these two metrics is interpreted in Section 5.

Table 11 and Figure 19 classify residual failure cases. Total visual error declined from 12.9% to 8.6%, with severe-occlusion misses and overlapping duplicate boxes reduced most; among whole-machine failures, peduncle entanglement accounted for 52.3% and neighbouring-fruit path blockage for 31.8%. Execution-related implications are discussed in Section 5.

Table 11. Residual failure-category comparison (share of tested samples, %).

Failure category	Baseline	YOLOv11n-EFAN
Severe-occlusion miss	5.4	3.0
Overlap duplicate box	3.4	2.2
Backlight over-exposure false positive	2.1	1.6
Small-target miss	1.5	1.3
Leaf misclassified as fruit	0.5	0.5
Total	12.9	8.6

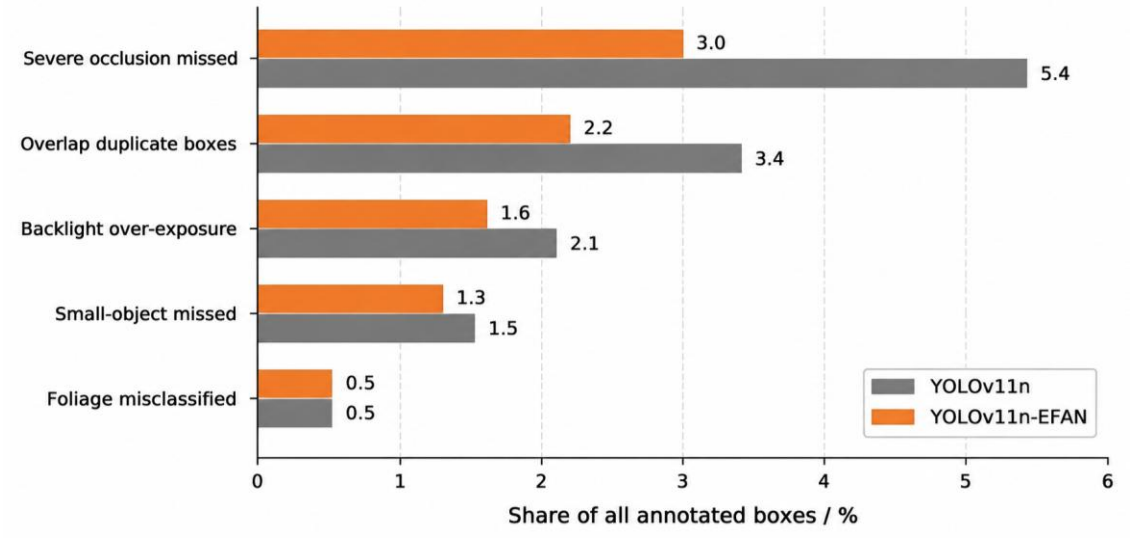


Figure 19. Distribution of the five residual failure categories before and after model improvement.

4.7 Summary of reported results

Sections 4.2–4.6 report the numerical evidence for module contributions, model comparison, scenario robustness, embedded inference and field operation. The principal observed pattern is that YOLOv11n-EFAN improved difficult-scene detection while remaining close to the nano-scale model size; the mechanism-level interpretation, deployment trade-off and operational gaps are consolidated in Section 5.

4.8 Scope and applicable conditions

The results apply to the tested dwarf-orchard type, cultivars, camera setup and autumn campaigns described in Section 3.2. Validation across additional cultivars, orchards, seasons and sensor configurations remains necessary; the corresponding boundary conditions and validation directions are discussed in Section 5.

5. Discussion

Combined ablation evidence indicates that YOLOv11n-EFAN improves orchard detection through stage-specific compensation rather than a uniform increase in network capacity. EFAN showed the largest isolated mAP@0.5 gain, suggesting that weak multi-scale routing was the principal limitation of the compact neck; DSC improved matching to fragmented contours under branch and leaf overlap; GCA reinforced fruit-relevant channels and locations; and Wise-IoU v3 reduced destabilizing gradients from ambiguously labelled occluded boxes. Because the four modifications operate at backbone, neck, fusion and regression stages, their effects can complement one another, although the combined gain is smaller than the sum of isolated contributions because several mechanisms address related errors. The design implication is that targeted structural modifications can be preferable to enlarging the backbone for this resource-constrained task.

The detector should be interpreted as a lightweight operating point rather than as an unrestricted benchmark maximum. The added adaptive offsets, weighted aggregation and gating operations reduce peak desktop throughput relative to the nano baseline, which is an expected trade-off when feature routing and weak-target discrimination are strengthened. The relevant criterion for a harvesting platform is not desktop frame rate alone but whether the embedded inference engine remains available with limited accuracy loss and acceptable power consumption. The edge results should therefore be read together with parameters, FLOPs, embedded latency and field robustness; a single speed or accuracy value would not adequately describe the operational balance.

Scenario results indicate that the modifications contribute most where conventional detectors are weakest: backlit scenes, LED night imaging, branch–leaf occlusion and overlapping fruit. This pattern is consistent with EFAN preserving weak high-resolution evidence for small or partially exposed

targets, GCA suppressing diffuse foliage responses, and DSC following irregular visible boundaries; Wise-IoU v3 is particularly relevant when annotation boxes intersect fruit edges because the visible fruit extent is uncertain. Confusion-matrix results also show that mature fruit were recognized more reliably than immature fruit, while residual background responses indicate further room for false-positive suppression. These findings support model evaluation under operationally difficult scenes rather than reliance on clean daylight imagery alone.

Edge deployment and whole-machine tests distinguish visual recognition from successful harvesting. The embedded engine provided practical inference availability, but the field picking success rate remained below the visual detection rate because an image centroid is not equivalent to a reachable grasp point. End-effector approach, peduncle position, neighbouring fruit, arm reach and collision avoidance jointly determine a successful grasp, and the physical failure records were dominated by stem entanglement and path blockage rather than by isolated visual misclassifications. Perception confidence should therefore be coupled with reachability estimation and contact-aware grasping; detection accuracy alone should not be treated as a substitute for harvesting performance.

External validity is bounded by the sampled region, cultivars, two autumn campaigns, camera geometry and training system. A monocular RGB system supplies two-dimensional appearance evidence but cannot directly recover rear-side geometry or depth, so the remaining field gap reflects both detector performance and system-level sensing/manipulation limitations. Subsequent validation should include independent orchards and additional cultivars, depth or stereo evidence, fruit/stem keypoint estimation, quantization-aware compression, and closed-loop reachability planning. The principal contribution remains targeted rather than comprehensive: the proposed modifications improve compact orchard detection while preserving edge deployability, whereas a robust full harvesting system still requires depth-supported grasp control and physical execution optimization. This manuscript makes no three-dimensional localization accuracy claim; the earlier 4.3 mm 3-D error was removed for lack of a depth source, and depth or stereo measurement must be added before any grasp-point 3-D validation.

6. Conclusions

This study designed and validated a lightweight visual detection scheme for apple harvesting robots, organized from functional requirements to model design, experiments and field deployment. (1) A C3k2-DSC backbone, EFAN weighted-aggregation neck, GCA attention gate and Wise-IoU v3 loss were combined into YOLOv11n-EFAN; on the test set it achieved P 91.2%, R 87.5%, mAP@0.5 93.1% and mAP@0.5:0.95 66.8%, outperforming the YOLOv11n baseline by 4.8, 4.8, 4.5 and 5.5 percentage points. (2) The model remains compact at 2.94×10^{6} parameters and 7.6×10^{9} FLOPs with a 5.9 MB weight file, and exports to a Jetson Orin Nano FP16 engine running at 41 frames s^{-1} (24.4 ms, 11.8 W) with 92.6% mAP@0.5. (3) Scenario analysis shows consistent gains under backlight, night LED lighting, occlusion and overlap, reducing the total residual visual error from 12.9% to 8.6%. (4) Closed-field tests on 320 fruits reached 97.6% detection, 86.2% picking success, 3.4% damage and two-dimensional detection/centroid planning-seed consistency; no three-dimensional localization metric was measured because no depth sensor was available.

Overall, YOLOv11n-EFAN provides a compact, field-tested apple detection solution that balances detection accuracy, lightweight model size and embedded inference availability for harvesting robots.

Author contributions: Xinying Lin: conceptualization, data collection, model training, visualization and original manuscript drafting. Zongde Lin: research supervision, experimental design, robot-platform integration, manuscript revision and corresponding-author responsibility. Ping Yu: dataset annotation, comparative experiments and field-test statistics. All authors have read and approved the final manuscript.

Data availability Statement: Statistical summaries, model configurations and result tables supporting this study are available from the corresponding author on reasonable request. Raw orchard imagery remains subject to orchard cooperation agreements.

Code Availability: Training and inference configurations, modified module definitions, FP16 export settings and trained weights are available from the corresponding author on reasonable request. The YOLOv11n baseline follows the public YOLOv11 implementation; only the proposed DSC, EFAN, GCA modules and Wise-IoU v3 integration were modified.

Acknowledgements: The authors thank the orchard cooperators in Shishi, Fujian, for permitting field image collection and platform testing during the two harvest campaigns.

Funding: This work was supported by the Opto-Mechatronics Innovation Team (No. 23XTD111) of Minnan University of Science and Technology and the Fujian Provincial Key Laboratory of Industrial Robot Measurement, Control and Mould Rapid Manufacturing (Grant No. Min Jiao Ke [2017] No. 8).
Conflicts of Interest: The authors declare no competing interests. The funders had no role in the study design, data collection or analysis, interpretation of results, manuscript preparation, or the decision to submit and publish the manuscript.

References

1. Gongal, A., Amatya, S., Karkee, M., Zhang, Q. & Lewis, K. Sensors and systems for fruit detection and localization: a review. Comput. Electron. Agric. 116, 8–19 (2015).
2. Koirala, A., Walsh, K. B., Wang, Z. & McCarthy, C. L. Deep learning – method overview and review of use for fruit detection and yield estimation. Comput. Electron. Agric. 162, 219–234 (2019).
3. Bargoti, S. & Underwood, J. Deep fruit detection in orchards. In Proc. 2017 IEEE International Conference on Robotics and Automation (ICRA), 3626–3633 (IEEE, 2017).
4. Jia, W., Tian, Y., Luo, R., Zhang, Z., Lian, J. & Zheng, Y. Detection and segmentation of overlapped fruits based on optimized mask R-CNN application in apple harvesting robot. Comput. Electron. Agric. 172, 105380 (2020).
5. Ren, S., He, K., Girshick, R. & Sun, J. Faster R-CNN: towards real-time object detection with region proposal networks. Adv. Neural Inf. Process. Syst. 28, 91–99 (2015).
6. Liu, W. et al. SSD: single shot MultiBox detector. In Computer Vision – ECCV 2016, LNCS 9905, 21–37 (Springer, 2016).
7. Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. You only look once: unified, real-time object detection. In Proc. IEEE CVPR, 779–788 (2016).
8. Redmon, J. & Farhadi, A. YOLOv3: an incremental improvement. Preprint at arXiv:1804.02767 (2018).
9. Bochkovskiy, A., Wang, C.-Y. & Liao, H.-Y. M. YOLOv4: optimal speed and accuracy of object detection. Preprint at arXiv:2004.10934 (2020).
10. Tian, Y., Yang, G., Wang, Z., Wang, H. & Li, E. Apple detection during different growth stages in orchards using the improved YOLO-V3 model. Comput. Electron. Agric. 157, 417–426 (2019).
11. Howard, A. G. et al. MobileNets: efficient convolutional neural networks for mobile vision applications. Preprint at arXiv:1704.04861 (2017).
12. Qi, P., Yuan, J., Gaoerkenbieke et al. Design and implementation of a lightweight real-time apple maturity detection model GCA-YOLOv8n. J. South China Agric. Univ. 47(1), 128–138 (2026).
13. Zhao, H., Qiao, Y., Wang, H. et al. Apple fruit recognition in complex orchard environments based on improved YOLOv3. Trans. Chin. Soc. Agric. Eng. 37(16), 127–135 (2021).

14. Li, C., Yang, K., Shen, T. et al. Apple harvesting visual localization and detection method based on improved Faster R-CNN. Trans. Chin. Soc. Agric. Mach. 55(1), 47–54 (2024).
15. Wang, Z., Wang, J., Wang, X. et al. Lightweight apple recognition method under natural environment based on improved YOLOv4. Trans. Chin. Soc. Agric. Mach. 53(8), 294–302 (2022).
16. Qi, P., Yuan, J., Yang, Y. et al. High-precision target detection algorithm for apple maturity based on improved YOLOv8n. Shandong Agric. Sci. 58(6), 179–188 (2026).
17. Niu, J., Yang, X., Zhang, T. et al. Apple maturity recognition and low-damage harvesting method based on YOLO11n-LDA. Trans. Chin. Soc. Agric. Mach. 57(11), 343–353 (2026).
18. Miao, R., Li, G., Huang, Z. et al. Apple maturity detection method in complex orchard environment based on YOLOv7-ST-ASFF. Trans. Chin. Soc. Agric. Mach. 55(6), 219–228 (2024).
19. Hu, S., Chen, W., Zhang, J. et al. Target detection method for apple harvesting robot based on improved YOLOv5. Res. Agric. Mechanization 46(6), 48–55 (2024).
20. Wang, H., Li, X., Bo, P. et al. Natural-scene apple external-quality detection method based on improved YOLOv8n. Trans. Chin. Soc. Agric. Eng. 41(11), 173–182 (2025).
21. Huang, Z., Lu, C., Cui, Y. et al. YOLO-AP: a lightweight apple fruit detection algorithm based on improved YOLO11n. Bull. Chin. Acad. Agric. Sci. 27(10), 118–133 (2025).
22. Wang, A., Chen, H., Liu, L. et al. YOLOv10: real-time end-to-end object detection. Adv. Neural Inf. Process. Syst. 37, 107984–108011 (2024).
23. Li, H., Aierken, H., Hou, Y. et al. Efficient tomato maturity detection method based on improved YOLOv7. Res. Agric. Mechanization 48(5), 120–127 (2026).
24. Tao, H., Zhang, R., Zhang, L. et al. Dense tea bud recognition algorithm using multiple attention-improved YOLOv8. Res. Agric. Mechanization 48(6), 203–212 (2026).
25. Zhang, D., Chen, N., Zhang, Q. et al. Lightweight orchard plum detection method based on improved YOLOv8s. Trans. Chin. Soc. Agric. Eng. 41(1), 154–160 (2025).
26. Ren, J., Wan, E., Shan, M. et al. Real-time tomato maturity detection algorithm based on improved YOLOv8n network. Trans. Chin. Soc. Agric. Mach. 56(3), 374–382 (2025).
27. Wang, C., Wen, S., Lan, Y. et al. Lightweight lychee detection method fusing attention mechanisms. Res. Agric. Mechanization 47(3), 10–15 (2025).
28. Xu, Y., Wu, Q., Zhang, B. et al. Advances in lightweight deep learning networks for crop target detection. J. Chin. Agric. Mechanization 46(3), 261–270 (2025).
29. Jocher, G., Chaurasia, A. & Qiu, J. Ultralytics YOLOv11 (Ultralytics, 2024). https://github.com/ultralytics/ultralytics.
30. Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B. & Belongie, S. Feature pyramid networks for object detection. In Proc. IEEE CVPR, 2117–2125 (2017).
31. Wang, C.-Y., Liao, H.-Y. M., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W. & Yeh, I.-H. CSPNet: a new backbone that can enhance learning capability of CNN. In Proc. IEEE/CVF CVPR Workshops, 1571–1580 (2020).
32. Qi, Y., He, Y., Qi, X. et al. Dynamic snake convolution based on topological geometric constraints for tubular structure segmentation. In Proc. IEEE/CVF ICCV, 6047–6056 (2023).
33. Woo, S., Park, J., Lee, J.-Y. & Kweon, I.-S. CBAM: convolutional block attention module. In Computer Vision – ECCV 2018, LNCS 11211, 3–19 (Springer, 2018).
34. Hu, J., Shen, L. & Sun, G. Squeeze-and-excitation networks. In Proc. IEEE CVPR, 7132–7141 (2018).
35. Tong, Z., Chen, Y., Xu, Z. & Yu, R. Wise-IoU: bounding box regression loss with dynamic focusing mechanism. Preprint at arXiv:2301.10051 (2023).
36. Jacob, B. et al. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In Proc. IEEE CVPR, 2704–2713 (2018).
37. Liu, J., Yan, F., Dong, H. et al. Pitaya target detection and localization method based on improved MBS-YOLOv8. Trans. Chin. Soc. Agric. Mach. 56(5), 425–432 (2025).

38. Jiang, Z., Zhou, J., Han, H. et al. Apple picking pose construction method based on RANSAC-fitted point cloud denoising. Trans. Chin. Soc. Agric. Mach. 55(10), 72–81 (2024).
39. Yang, Z., Wang, X., Qi, Z. et al. Strawberry recognition and peduncle picking keypoint detection based on improved YOLOv8. Trans. Chin. Soc. Agric. Eng. 40(18), 167–175 (2024).
40. Scharstein, D. & Szeliski, R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. Comput. Vis. 47(1–3), 7–42 (2002).