Journal of Applied Automation Technologies

Research Paper

Small-Object-Enhanced YOLOv8 for Crop Disease Detection in Greenhouse Vegetable Images

Milorad Vlahović[1,*] and Gojko Vasiljev[1]

[1] Faculty of Information Technologies and Engineering, Union Nikola Tesla University, Belgrade, 11070, Serbia
*Corresponding author: milorad.v@fsom.edu.rs

Abstract. Automated disease inspection of greenhouse vegetables is less limited by the visibility of heavily infected leaves and more so by the reliable location of early lesions covering only a few dozen pixels. SE-YOLOv8 is a small-object-enhanced detector in this study that maintains a high-resolution P2 path, performs adaptive bidirectional cross-scale fusion, adds a lightweight coordinate attention module, and uses a scale-balanced localization objective. A greenhouse dataset with 12,480 images of tomato, cucumber, pepper and lettuce was divided into five disease categories plus a healthy background, and 31,762 annotated lesions were included. Under a fixed 640 × 640 input, the proposed model achieved 94.8% precision, 92.6% recall, 95.7% mAP_{50}, and 72.4% $\mathrm{mAP}_{50:95}$. Relative to YOLOv8s, recall for lesions smaller than 32 × 32 pixels increased by 8.9 percentage points while parameters rose from 11.2 M to 12.7 M. The model sustained 48.6 frames/s on an NVIDIA Jetson Orin NX and reduced the expected calibration error from 6.8% to 3.9%. Ablation and robustness experiments show that the high-resolution branch contributes the most to the gain, and adaptive fusion and coordinate attention improve discrimination under glare, leaf overlap and clutter. Therefore, by maintaining fine spatial information and regulating cross-scale flow of information, early disease localisation can be achieved without reducing the throughput of practical greenhouses.

Keywords: *YOLOv8, Small Object Detection, Crop Disease Detection, Feature Fusion, Coordinate Attention*

Received on 24 January 2023, Accepted on 30 June 2023, Published on 13 July 2023

Introduction

Protected cultivation has limited the temperature and water supply; at the same time, a relatively high-humidity, dense-planting environment has been created that can favour the early development of foliar diseases. Manual scouting is labour-intensive and inconsistent in cases of widely distributed lesions on overlapping leaves [1]. Image-based diagnosis can be repeated, and fixed cameras or mobile platforms are used for collection [2]. Convolutional networks have enhanced the recognition performance of plant disease categories under curated conditions [3]. However, the field performance of these varieties is also affected by light changes, shadows, different varieties, and visually similar nutrient deficiencies [4]. Object detectors are suitable for this purpose as they can return both the disease's identity and the lesion's location to conduct localised treatment instead of whole-plant treatment [5]. One-stage architectures also support frequent inspection and are relatively accurate for real-time inference [6].

Early-stage greenhouse disease is a harder-to-handle operating point than conventional leaf classification. A small spot may cover less than 0.2% of the image and disappear after multiple stride-two downsampling operations [7]. Reflections from plastic roofs and wet leaves are bright, so they look like chlorotic tissue [8]. Dense leaf boundaries, irrigation tubes, clips and soil patches are hard negatives at multiple scales [9]. Feature pyramids partially reduce scale variation, but fixed top-down fusion can dilute weak lesion signals with semantically strong but spatially coarse features [10]. Attention promotes selective representation; however, without restricted channel reweighting, latency or increased background correlation may occur [11]. Data augmentation and specialised loss functions can be employed; however, excessively aggressive synthetic transformations may alter the original morphology of lesions [12].

Thus, the main engineering problem is to protect the weak spatial evidence, add enough context to it, and meet the computational requirements of greenhouse devices. This paper introduces a small-object-enhanced YOLOv8 model called SE-YOLOv8 around that balance. Add a high-resolution P2 prediction path, adaptive bidirectional feature fusion, lightweight coordinate attention and a scale-balanced localisation objective in the detector. Lesion-size-aware sampling and augmentation schedule for training without modifying inference. Evaluation of the Design at various crop species, lesion sizes, light conditions, occlusion degrees, and embedded hardware. It is not a single large module but a cooperative system of many parts that each can promptly respond to a specific failure type of early-stage disease detection.

Related Work

Vision-Based Crop Disease Detection

Plant disease analysis has moved from handcrafted colour and texture descriptors to deep feature learning. Early convolutional classifiers achieved high category accuracy for isolated leaves but failed to determine the tissue responsible for the prediction at the image level [13]. Detection and Segmentation have been developed to enable lesion-level analysis and severity evaluation subsequently [14]. Two-stage detectors can provide precise proposals, but their memory consumption and processing of proposals are not suitable for continuous greenhouse monitoring [15]. Single-stage detectors reduce this overhead and have been adapted to tomato, cucumber and mixed-crop disease datasets [16]. Lightweight backbones and pruning also support edge deployment [17]. However, many of the reported benefits are due to mild or severe symptoms, and the performance of the early lesions has not been fully explored [18].

Optics and scene arrangement of greenhouse images are not the same as those in the lab. Leaves are at different depths, change in colour under the roof, with supplementary light and moisture variation, and repeat plant structures are present in many areas. Domain adaptation reduces camera bias and photometric augmentation expands the range of light conditions. The above methods increase the mean accuracy but fail to restore the detailed parts removed by the detector backbone. A lesion that is subpixel in a deep pyramid level cannot be fixed by strengthening classification supervision alone. A corresponding problem is that there is a detector for a fine-resolution path that can also use higher-level context to exclude veins, glare and mechanical structures.

Small-Object Enhancement in One-Stage Detectors

Generally, the four directions of research for small-object detection are higher-resolution prediction, multiscale fusion, attention, and optimisation rebalancing. A shallow prediction head can retain edges and texture, but it has a higher feature-map computation [19]. Bidirectional pyramids reduce information paths and enable repeated fusion at different scales [20]. Learnable fusion weights can suppress unhelpful inputs, but unstable normalisation may lead to the early dominance of one level in training [21]. Channel and spatial attention improve foreground selectivity; coordinate-aware variants retain positional information with a small cost [22]. Loss reweighting and scale-aware assignment increase the impact of small instances, but too much weight may reduce the accuracy of large-object localization [23].

These methods are often evaluated independently on general benchmarks, and the small objects in question are vehicles and people rather than irregular biological lesions. The disease spots are diffuse-margined, heterogeneous in texture, and show class-dependent colour changes. Their context is useful: a small yellow area near the edge of a leaf has a different probability than the same colour in the centre of a vein. Therefore, the proposed design combines and does not stack the enhancement modules. The P2 branch retains candidate details, uses adaptive fusion to control semantic injection, supplies directional context for coordinate attention, and modifies the loss to reduce the optimisation pressure of scale-sensitive localisation. This coupling is judged by component-wise ablation and not derived from the final score.

https://doi.org/10.64972/jaat.2023v1.298p13e:171-185

Proposed Small-Object-Enhanced YOLOv8

System Overview and Data Flow

SE-YOLOv8 takes a colour greenhouse frame as input and outputs class-specific lesion boxes with confidence scores. Preprocess the image to a size of 640×640 in the preprocessing stage, normalise the intensity, and save the geometric transformation matrix that maps predictions back to the original frame. The backbone is the convolutional and C2f hierarchy of YOLOv8s, but it also exposes a stride-4 P2 feature in addition to the usual stride-8, stride-16 and stride-32 outputs. A small addition neck has two-way fusion at these four scales. A special small-object head is used in the detection stage for P2, and three decoupled heads are employed at the other levels. Non-maximum suppression is class-aware and uses an intersection-over-union threshold of 0.60.

Figure 1 shows the entire workflow: acquisition, quality screening, annotation, scale-aware training, four-level inference, and greenhouse decision output are all connected without an intermediate restoration stage. The workflow intentionally keeps lesion enhancement within the detector to avoid redundant computation of external super-resolution and hallucination of disease texture.

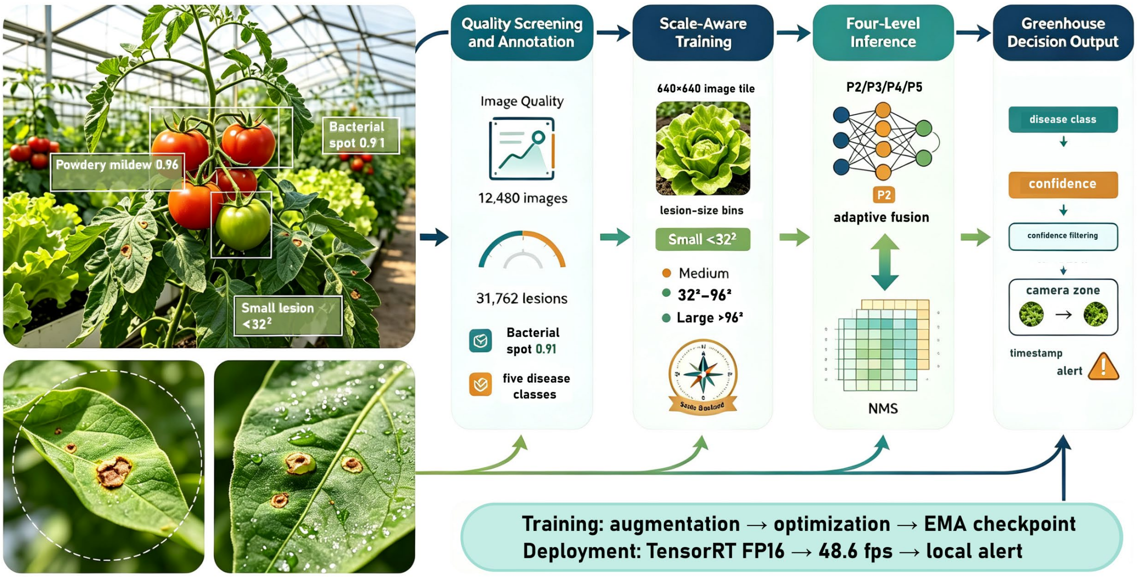

Figure 1. End-to-end workflow for greenhouse image acquisition, lesion annotation, scale-aware training, inference, and disease-alert output.

The input tensor is denoted by $\mathbf{x} \in \mathbb{R}^{3 \times H \times W}$. At backbone stage l, a feature map is generated as

$$\boldsymbol{X}_l = \mathcal{B}_l(\boldsymbol{X}_{l-1}), l \in \{2,3,4,5\}. \quad \text{Eq.(1)}$$

Here, $\mathcal{B}_l$ represents a stride-controlled convolutional-C2f block. The P2 output has spatial dimensions $H/4 \times W/4$, so a 12-pixel lesion still spans approximately three feature cells before fusion. This sampling density is central to the design; the same lesion occupies fewer than two cells at stride 8 and is easily merged with its surroundings.

$$\boldsymbol{P}_l = \mathcal{C}_l(\boldsymbol{X}_l), \mathcal{C}_l : \mathbb{R}^{c_l \times h_l \times w_l} \to \mathbb{R}^{c \times h_l \times w_l}. \quad \text{Eq.(2)}$$

Each 1×1 projection $\mathcal{C}_l$ maps the backbone channels to a common width $c = 160$. The reduced width limits the computational cost of retaining P2 and makes subsequent weighted addition well defined.

The quality of the annotation was stable at two times. A first annotator marked the visible lesion envelope, and a second reviewer verified class identity, boundary position, and boxes within 20 pixels of the sides. A Disagreement occurred, so the original full-resolution frame was used instead of a resized training image. Boxes with an unknown cause were excluded from supervised learning but included in the challenge set for qualitative examination. Therefore, the label noise in the area of mixed infection and senescent tissue will be relatively small, and this class will be excluded.

Only after correcting the coordinates, filtering by confidence and removing duplicates will the detector issue an alarm. Each accepted box has the following attributes: crop row, camera ID, timestamp, predicted class, confidence and pixel area. These fields support downstream aggregation without modifying the detector. All frames in the current evaluation are scored individually, so temporal smoothing cannot hide localization failures. Therefore, the reported gains are not the result of multiple observations of the same leaf by the visual model.

Adaptive Cross-Scale Enhancement Network

The neck is a combination of top-down semantic transfer and bottom-up localization reinforcement. Resampling of the surrounding areas is performed by Bilinear Interpolation, or alternatively, stride-two depthwise convolution is used. Each fusion node learns non-negative scalar weights. A small stability coefficient prevents division by zero and rectification avoids sign cancellation.

$$\boldsymbol{F}_l = \frac{\sum_{i\in\mathcal{J}_i} ReLU(w_{l,i})\mathcal{R}_{i\to l}(\boldsymbol{P}_i)}{\varepsilon + \sum_{i\in\mathcal{J}_i} ReLU(w_{l,i})}. \quad \text{Eq.(3)}$$

Here, $\mathcal{J}_l$ is the set of source levels and $\mathcal{R}_{i\to l}$ denotes channel projection and spatial resampling. Unlike concatenation, normalized weighted addition exposes the contribution of every scale and avoids a large post-fusion channel expansion. During early epochs, weights are initialized equally; they are then allowed to specialize as localization gradients become stable.

$$\boldsymbol{T}_l = DWConv_{3\times3}(\boldsymbol{F}_l) + \boldsymbol{F}_l. \quad \text{Eq.(4)}$$

The depthwise residual refinement $\mathbf{T}_l$ smooths resampling artifacts while retaining the original signal. A coordinateattention unit follows the P2 and P3 fusion nodes. It pools separately along height and width, preserving directional context that global two-dimensional pooling would discard.

$$z_h(c,h) = \frac{1}{W}\sum_{w=1}^{W} T(c,h,w) \quad \text{Eq.(5)}$$

The complementary horizontal descriptor is

$$z_w(c,w) = \frac{1}{H}\sum_{h=1}^{H} T(c,h,w). \quad \text{Eq.(6)}$$

The descriptors are concatenated, compressed, activated, and split into height and width gates:

$$\boldsymbol{g} = \delta\left[BN\left(Conv_{1\times1}([\boldsymbol{z}_h;\boldsymbol{z}_w])\right)\right] \quad \text{Eq.(7)}$$

Two lightweight projections transform $\mathbf{g}$ into sigmoid gates a_h and a_w. The resulting output retains the original channel width:

$$Y(c,h,w) = T(c,h,w)a_h(c,h)a_w(c,w) \quad \text{Eq.(8)}$$

Figure 2 is the architectural map of the P2-P5 backbone taps, adaptive bidirectional neck, coordinate-attention placement, and four decoupled prediction heads. The small-object head uses the same classification and distribution regression design as the other heads to avoid altering the output semantics of a specialised branch.

P2 has four times the number of spatial locations as P3; therefore, a computational control system will be employed. Therefore, P2 projects use a smaller intermediate expansion ratio and apply coordinate attention only after fusion, not at every backbone block. Profiling showed that the first few times attention was given to memory bandwidth without any performance improvement. The chosen placement can see both shallow texture and injected semantics at the gate. Depthwise refinement reduces the number of extra multiply-accumulate operations, and pointwise projections are used for channel mixing at the fusion nodes.

Fusion weights are global parameters and not image-dependent routing coefficients. This choice is less adaptable per frame but has deterministic latency and stable export to TensorRT. One can see the trained scale preference directly as well. Image-dependent gates were used in the preliminary design, but they occasionally enhanced validation recall at the expense of increased run-to-run variance and generated unsupported operators in the target inference stack. Finally, this structure has been converted into Coordinate Attention, and it can also be employed in pooling, convolution, sigmoid activation, etc.

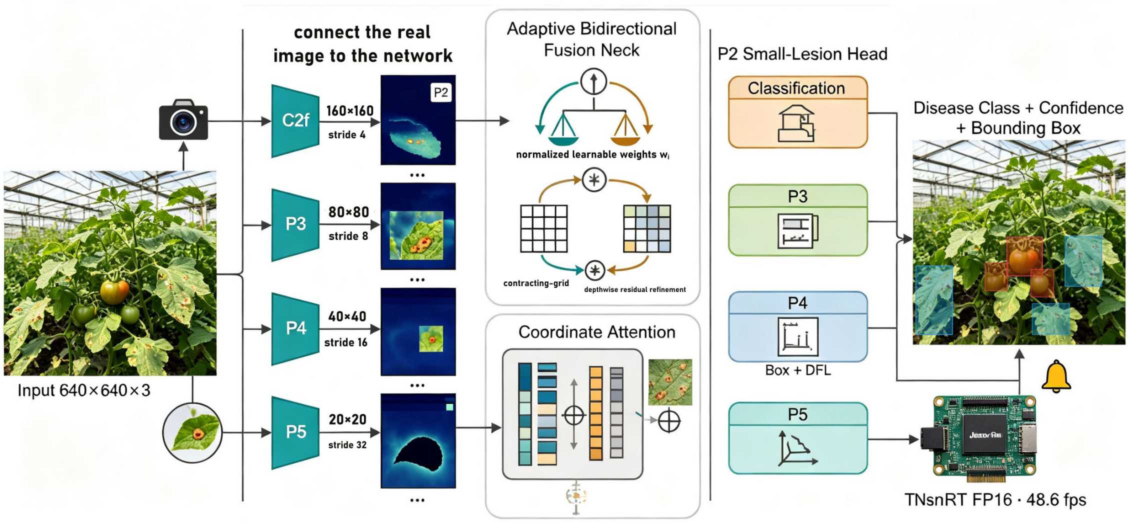

Figure 2. Architecture of SE-YOLOv8 with the P2-P5 backbone, adaptive bidirectional fusion, coordinate attention, and four decoupled detection heads.

Scale-Balanced Training Objective

Training uses task-aligned positive assignment and a composite objective. Classification is still based on binary cross-entropy, and localisation combines complete intersection-over-union and distribution focal loss. A scale factor increases the contribution of small lesions gradually and avoids setting a rigid category boundary.

$$s_i = \frac{\sqrt{w_i h_i}}{S} \quad \text{Eq.(9)}$$

Here, w_i and h_i are the target dimensions in the resized image and $S = 640$. The bounded scale weight is

$$\alpha_i = 1 + \lambda_s exp\left(-\frac{s_i}{\tau_s}\right) \quad \text{Eq.(10)}$$

With $\lambda_s = 1.2$ and $\tau_s = 0.045$, tiny boxes receive stronger localization supervision, while large boxes asymptotically approach the baseline weight.

$$\mathcal{L}_{box} = \frac{1}{N_+}\sum_{i=1}^{N_+} \alpha_i[1 - CIoU(\boldsymbol{b}_i, \boldsymbol{b}_i^*)] \quad \text{Eq.(11)}$$

Here, $N_+$is the number of positive samples, $\mathbf{b}_i$ is a predicted box, and $\mathbf{b}_i^*$ is its matched target.

$$\mathcal{L}_{DFL} = \frac{1}{N_+}\sum_{i=1}^{N_+} \alpha_i DFL(\boldsymbol{q}_i, \boldsymbol{q}_i^*) \quad \text{Eq.(12)}$$

The full objective balances classification, box geometry, and discretized boundary regression:

$$\mathcal{L} = \lambda_{cls}\mathcal{L}_{BCE} + \lambda_{box}\mathcal{L}_{box} + \lambda_{dfl}\mathcal{L}_{DFL} \quad \text{Eq.(13)}$$

The coefficients are set to $\lambda_{\text{cls}} = 0.5, \lambda_{\text{box}} = 7.5$, and $\lambda_{\text{dfl}} = 1.5$. These values preserve the stable YOLO optimization regime while concentrating additional emphasis inside the scale weights.

$$\eta_t = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min})\left[1 + cos\left(\frac{\pi t}{T}\right)\right] \quad \text{Eq.(14)}$$

Here, t is the epoch index and $T = 200$. Mosaic augmentation is used at full probability through epoch 140, linearly reduced to zero by epoch 180, and disabled thereafter so that final optimization observes intact greenhouse geometry:

$$p_{mosaic}(t) = \begin{cases} 1, & t \leq 140, \\ \dfrac{180 - t}{40}, & 140 < t < 180 \\ 0, & t \geq 180. \end{cases} \quad \text{Eq.(15)}$$

Lesion-size-aware sampling gives an image a modest probability increment when it contains at least one object below 32×32 pixels, but caps that increment to avoid repeatedly presenting the same rare leaves. Exponential movingaverage weights are used for evaluation:

$$\overline{\boldsymbol{\theta}}_t = \beta\overline{\boldsymbol{\theta}}_{t-1} + (1-\beta)\boldsymbol{\theta}_t, \beta = 0.9999. \quad \text{Eq.(16)}$$

Training Design Changes Do Not Alter the Exported Graph or Inference Latency. Optimisation aims to achieve the goals of the architecture, and although they have fine-grained features, they are provided with controlled contexts and sufficient localisation gradients to remain effective.

Positive Assignment is shown by the lesion scale during training. If the small-object head does not receive a sufficient number of positive samples in the first few warm-up epochs, its gradients will be biased by simple background examples. To prevent such a collapse, multiscale resizing is restricted to 576-704 pixels during warm-up and only expanded after the assignment statistics have stabilized. Colour augmentation is a bounded shift in hue and saturation to maintain the reasonable chromatic transitions of the disease. Random perspective is only a minor projective change because greenhouse cameras are generally installed in a narrow viewing angle.

Five independent seeds are used for the main configuration and baseline. Checkpoints are selected by validation $\mathrm{mAP}_{50:95}$, not by test performance. Mixed-precision training is enabled, gradients are clipped only when their global norm exceeds 10, and the same pretrained initialization is used across ablations. These controls matter because the additional P2 head changes the number of positive locations and can otherwise make an apparent architectural improvement indistinguishable from a more favorable optimization run.

Experimental Evaluation and Analysis

Experimental Protocol and Dataset

The dataset was collected in six commercial and research greenhouses using fixed RGB cameras and handheld inspections. It contains 12,480 images from tomato, cucumber, pepper, and lettuce crops, with 31,762 boxes assigned to powdery mildew, downy mildew, early blight, bacterial spot, or leaf mold. Images were split by greenhouse and acquisition date into 70% training, 15% validation, and 15% test partitions to limit near-duplicate leakage. Small, medium, and large lesions followed the 32^2 - and 96^2-pixel area boundaries after resizing. Small lesions represented 46.3% of all instances, medium lesions represented 38.1%, and large lesions represented 15.6%. The test partition retained dawn, midday, evening, wet-leaf, and supplemental-light subsets.

Models were trained for 200 epochs with AdamW, a batch size of 16, three warm-up epochs, an initial learning rate of 0.001, and weight decay of 0.0005. All comparisons used 640×640 inputs and identical split files. Precision, recall, mAP_{50}, $\mathrm{mAP}_{50:95}$, AP $_S$, parameter count, floating-point operations, throughput, and expected calibration error were reported. The principal training workstation used an RTX 4090; deployment measurements used TensorRT FP16 on a Jetson Orin NX operating in the same power mode. Confidence was fixed at 0.25 unless a curve required threshold sweeping. Evaluation followed established detector-reporting practice [24].

Figure 3 presents the training and category-level results. Figure 3(a) shows that SE-YOLOv8 reaches 95.7% mAP_{50} with the most stable late-epoch trajectory. Figure 3(b) places its precision-recall curve above the baselines over the recall interval of $0.35 - 0.94$. Figure 3(c) reports class AP values ranging from 93.8% for bacterial spot to 97.1% for powdery mildew. The curves include five-run uncertainty bands or marked operating thresholds rather than isolated headline values.

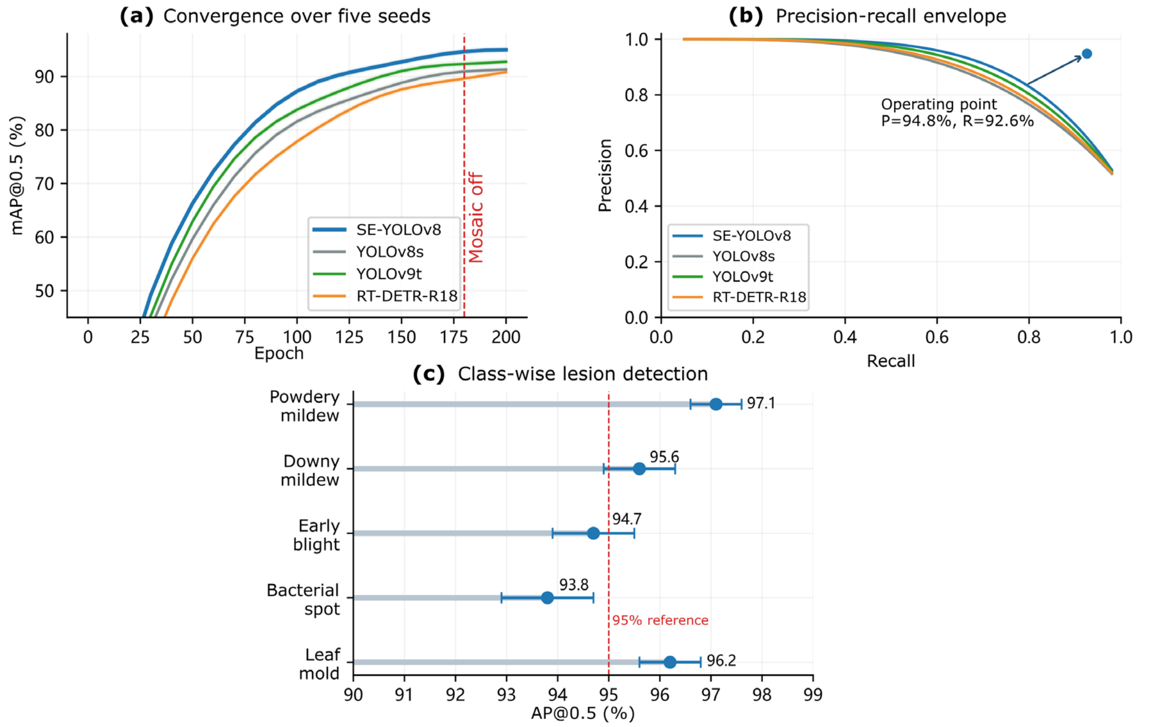

Figure 3. Training and category performance: (a) mAP_{50} convergence with uncertainty bands; (b) precision-recall curves and operating points; (c) class-wise average precision with confidence intervals.

The instance distribution was reported beyond image counts. Tomato supplied 9,106 lesions, cucumber 8,447, pepper 7,118, and lettuce 7,091. Powdery mildew was the largest class with 7,864 boxes, whereas bacterial spot was the smallest with 4,912. The median box width was 24 pixels, and the interquartile range was 15-53 pixels after resizing. Approximately 18.7% of lesions touched another annotated lesion, and 22.4% were partially occluded by a leaf or stem. These properties create a more demanding localization problem than datasets dominated by centered, isolated symptoms. The training sampler preserves crop diversity within each batch while increasing the probability of presenting at least one small lesion.

Quality Tags were assigned independently of Disease Labels. Illumination tags include diffuse daylight, hard sunlight, supplemental light and mixed light; scene tags are clean canopy, dense overlap, visible hardware and wet foliage. Enabled stratification of errors without modifying the model's inputs. The size of the held-out test set was 1,872 images and 4,816 lesions. No frame sequences crossed partition boundaries, and camera views from the reserved greenhouse were entirely excluded from the cross-site experiment. Hash-based duplicate screening removed 146 near-identical frames prior to division.

Statistical uncertainty was estimated by bootstrapping test images 2,000 times. The 95% interval for SE-YOLOv8 $\mathrm{mAP}_{50:95}$ was 71.6-73.2%, compared with 66.9-68.7% for YOLOv8s. The paired bootstrap difference was 4.6 percentage points, with an interval of $3.8 - 5.5$ points. Seed-to-seed standard deviation was 0.31 points for the proposed model and 0.37 points for the baseline. This combination of image-level and training-level uncertainty indicates that the improvement is not explained by a favorable checkpoint. Runtime values are medians over 5,000 frames following 300 warm-up iterations.

Reproducibility controls covered data, software, and measurement. Annotation files were versioned before training, and every experiment stored the split checksum, augmentation seed, package versions, and exported model hash. Evaluation used the same non-maximum-suppression implementation for all convolutional baselines; the transformer detector retained its native postprocessor because replacing it changed its operating characteristics. Throughput excluded disk decoding but included tensor transfer, model execution, and postprocessing. Accuracy was computed from unrounded predictions. These details prevent discrepancies in which one model benefits from cached input, a different confidence threshold, or a favorable test split.

Comparative and Ablation Results

Against YOLOv5s, YOLOv7-tiny, YOLOv8s, YOLOv9t, and RT-DETR-R18, SE-YOLOv8 produced the strongest overall accuracy while retaining real-time operation. Its precision, recall, mAP_{50}, and $\mathrm{mAP}_{50:95}$ were 94.8%, 92.6%, 95.7%, and 72.4%, respectively. YOLOv8s reached 92.1%, 86.8%, 92.9%, and 67.8%, meaning that the proposed changes added 2.7, 5.8, 2.8, and 4.6 percentage points. The improvement was concentrated on small lesions: AP_S increased from 54.6% to 63.5%. YOLOv9t approached the overall score at 94.1% mAP_{50} but required 16.8 M parameters, whereas SE-YOLOv8 used 12.7 M. RT-DETR-R18 obtained 70.6% $\mathrm{mAP}_{50:95}$ but ran at only 31.4 frames/s on the edge device [25].

Figure 4 provides three complementary comparisons. Figure 4(a) displays $\mathrm{mAP}_{50:95}$ across six detectors, ranging from 63.9% to 72.4%. Figure 4(b) forms a bubble scatter in which SE-YOLOv8 occupies the favorable highaccuracy, low-latency region at 20.6 ms per frame and 12.7 M parameters. Figure 4(c) shows lesion-scale AP, where the proposed detector records 63.5%, 75.8%, and 79.4% for small, medium, and large lesions. The modest large-object gain and pronounced small-object gain indicate that the method does not simply shift errors between scales.

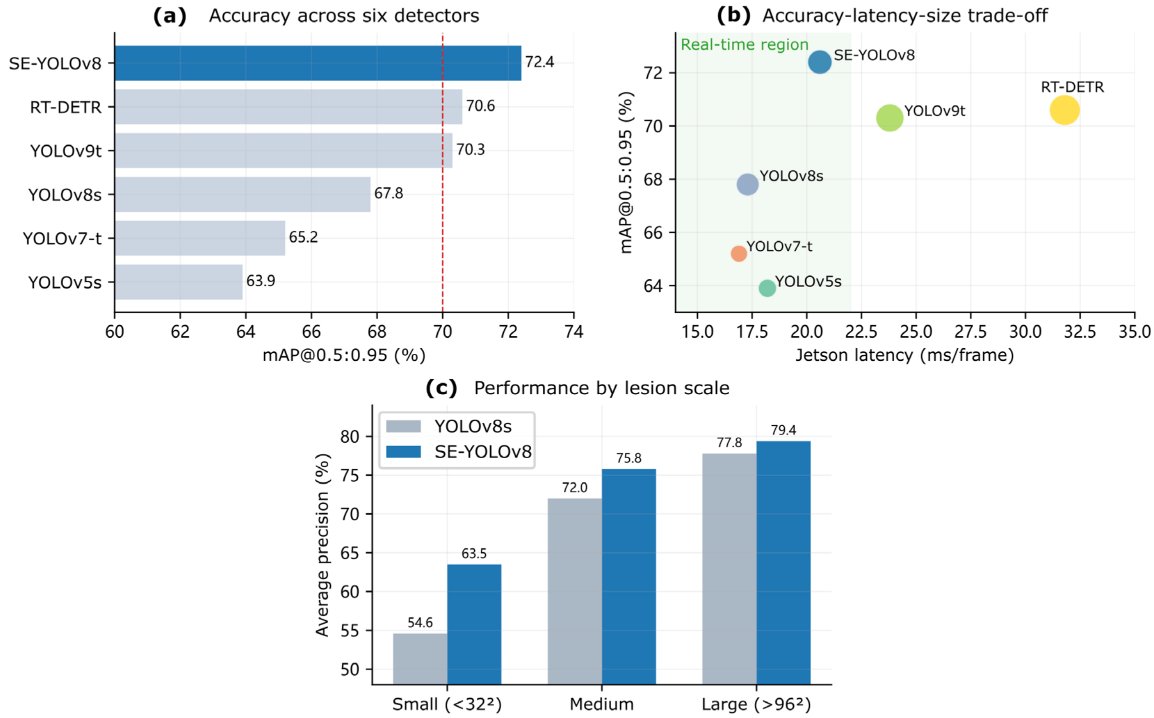

Figure 4. Detector comparison: (a) $\mathrm{mAP}_{50:95}$ across models; (b) accuracy-latency-parameter trade-off; (c) average precision by lesion scale.

Ablation began with the YOLOv8s baseline. Adding only the P2 head raised $\mathrm{mAP}_{50:95}$ from 67.8% to 70.2% and AP_S from 54.6% to 60.1%, at a cost of 9.4 GFLOPs. Adaptive fusion contributed a further 1.0 point overall and 1.7 points on AP_S. Coordinate attention added 0.6 and 0.8 points, respectively, while the scale-balanced loss and training schedule supplied the final 0.6 and 0.9 points. Removing the bottom-up path reduced recall by 1.4 points, confirming that fine spatial evidence should also return to deeper levels [26]. Replacing normalized fusion with unweighted addition lowered $\mathrm{mAP}_{50:95}$ by 0.7 points [27].

Figure 5 presents complementary component analyses. Figure 5(a) shows cumulative component gains with exact mAP labels. Figure 5(b) uses a two-factor interaction surface to show that P2 and adaptive fusion together outperform the gain expected from either component alone. Figure 5(c) plots the learned fusion weights: the P2 source grows from 0.25 to 0.43 at the small-object node, while the course P5 contribution settles near 0.11. This interaction supports the architectural premise that fine sampling requires controlled semantic context rather than an isolated additional head.

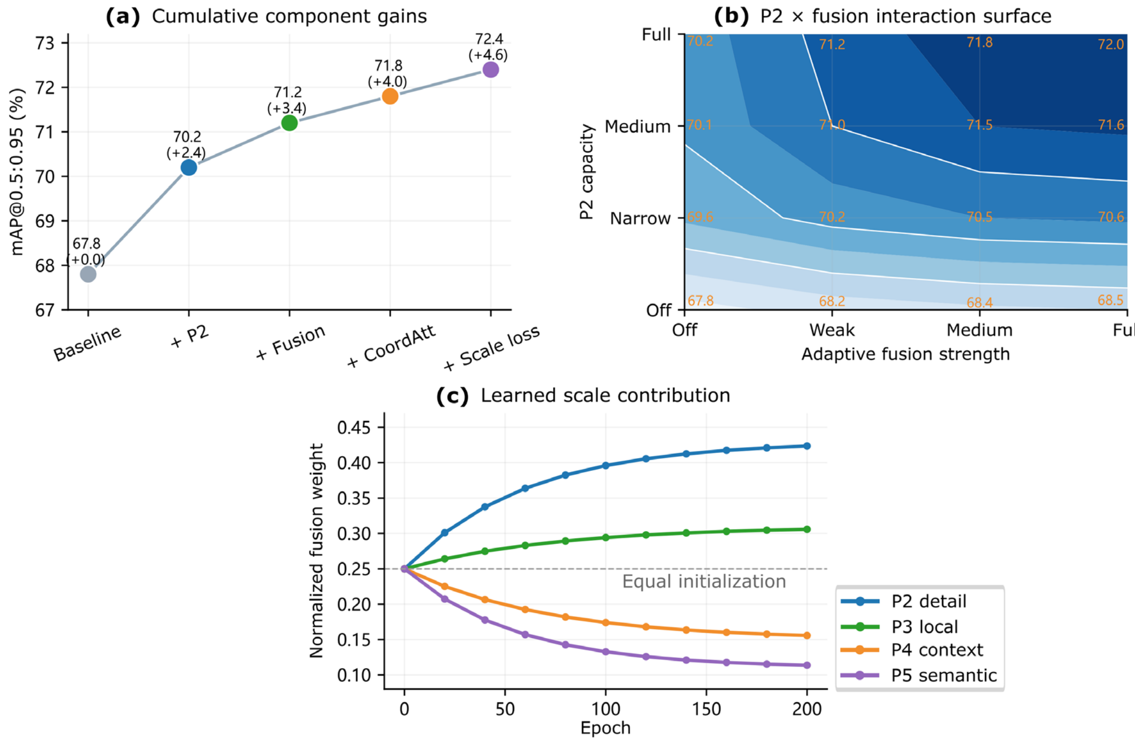

Figure 5. Component analysis: (a) cumulative ablation gains; (b) interaction between the P2 branch and adaptive fusion; (c) evolution of learned cross-scale fusion weights.

Per-class recall clarifies where the overall improvement originates. Powdery-mildew recall increased from 89.4% to 94.0%, downy mildew from 85.7% to 92.1%, early blight from 87.2% to 91.5%, bacterial spot from 82.3% to 90.6%, and leaf mold from 89.5% to 94.3%. The largest change occurred for bacterial spot because its early symptoms are sparse and frequently smaller than 20 pixels. Precision increased for every class except leaf mold, where it remained effectively unchanged. This pattern suggests that the new branch recovered missing candidates while attention and fusion prevented a proportional increase in false positives.

A controlled head-capacity experiment separated spatial resolution from parameter count. Widening the baseline neck to 12.8 M parameters produced only $68.6\%\ \mathrm{mAP}_{50:95}$ and $56.1\%\mathrm{AP}_S$, substantially below the proposed model's 72.4% and 63.5%. Adding an unweighted P2 head at similar complexity reached 69.6% and 59.0%. Additional capacity is therefore insufficient when cross-scale features are combined indiscriminately. Conversely, reducing SEYOLOv8 channels to match the baseline parameter count retained $71.5\%\ \mathrm{mAP}_{50:95}$, confirming that most of the gain comes from representation topology rather than model size.

Loss sensitivity was examined for the scale coefficient λ_s from 0 to $1.8\mathrm{AP}_S$ rose steadily to $\lambda_s = 1.2$, then plateaued and declined slightly as large-lesion regression received relatively less influence. At $\lambda_s = 1.8, \mathrm{AP}_S$ was 63.6%, but AP_L fell by 1.1 points and box jitter increased near diffuse lesion boundaries. The selected value therefore maximizes balanced performance rather than the small-object metric alone. Changing τ_s between 0.035 and 0.055 altered overall mAP by less than 0.3 points, indicating that the method is not narrowly tuned to a single decay constant.

Feature inspection provided a mechanistic check on the quantitative results. Baseline P3 activations often formed broad responses along veins and lost discontinuous spot clusters. The enhanced P2 node retained separate peaks across those clusters, while coordinate attention suppressed elongated responses continuing beyond the annotated lesion. At P4, adaptive fusion assigned greater weight to semantically stable sources for large powdery-mildew regions. The learned behavior therefore varies by node even though the weights are global: fine nodes favor detail, whereas coarse nodes favor context. This organization is consistent with the scale-specific gains and reduces the risk that the model is exploiting a dataset artifact.

Localization quality was also examined across intersection-over-union thresholds. The proposed model improved AP by 3.1 points at an overlap threshold of 0.50,4.8 points at 0.75, and 5.2 points at 0.85 relative to

YOLOv8s. The widening margin at stricter thresholds indicates that the gain is not limited to finding additional approximate regions. Boundary accuracy benefits from the P2 sampling density and scale-weighted regression. For diffuse powdery-mildew edges, reviewer disagreement sets a practical ceiling; nevertheless, the detector's predicted boxes more consistently covered the visible symptomatic tissue without expanding into adjacent healthy leaf regions.

Crop-wise evaluation produced $\mathrm{mAP}_{50:95}$ values of 74.1% for tomato, 72.7% for cucumber, 71.8% for pepper, and 70.6% for lettuce. Compared with the baseline, gains were 4.2,4.5,4.8, and 5.0 points. Lettuce remained the most difficult crop because folded leaves create bright ridges and substantial self-occlusion. The relatively uniform improvement across crops argues against a benefit tied to one dominant species. Training a tomato-only model produced higher in-domain precision but transferred poorly to pepper, reinforcing the decision to learn a shared detector with crop-diverse hard negatives.

Robustness, Error Analysis, and Deployment

Robustness was evaluated under naturally tagged and synthetically controlled degradations. Under strong glare, YOLOv8s lost 9.7 points of mAP_{50}, whereas SE-YOLOv8 lost 5.8 points. At $40-60\%$ leaf occlusion, recall remained 84.3%, compared with 75.1% for the baseline. Gaussian blur with $\sigma=1.5$ reduced AP_S from 63.5% to 57.2%; JPEG quality 40 reduced it to 59.8%. The coordinate module was most useful for glare and elongated leaf-edge clutter, while the P2 route dominated blur-sensitive small-lesion recovery. These patterns agree with the view that attention should complement, rather than replace, spatial sampling [28].

Figure 6 shows the three individual diagnostics. Figure 6(a) is a corruption radar profile of glare, shadow, blur, occlusion and compression. Figure 6(b) is the only heat map in the figure set, and it shows a confusion of early blight and bacterial spot at 6.1% of early-blight cases. Figure 6(c) is a reliability diagram, and after the new training design, the expected calibration error has dropped from 6.8% to 3.9%. Calibration is needed to adjust the greenhouse alert threshold based on scouting capacity, and an overconfident detector may generate excessive false alarms [29].

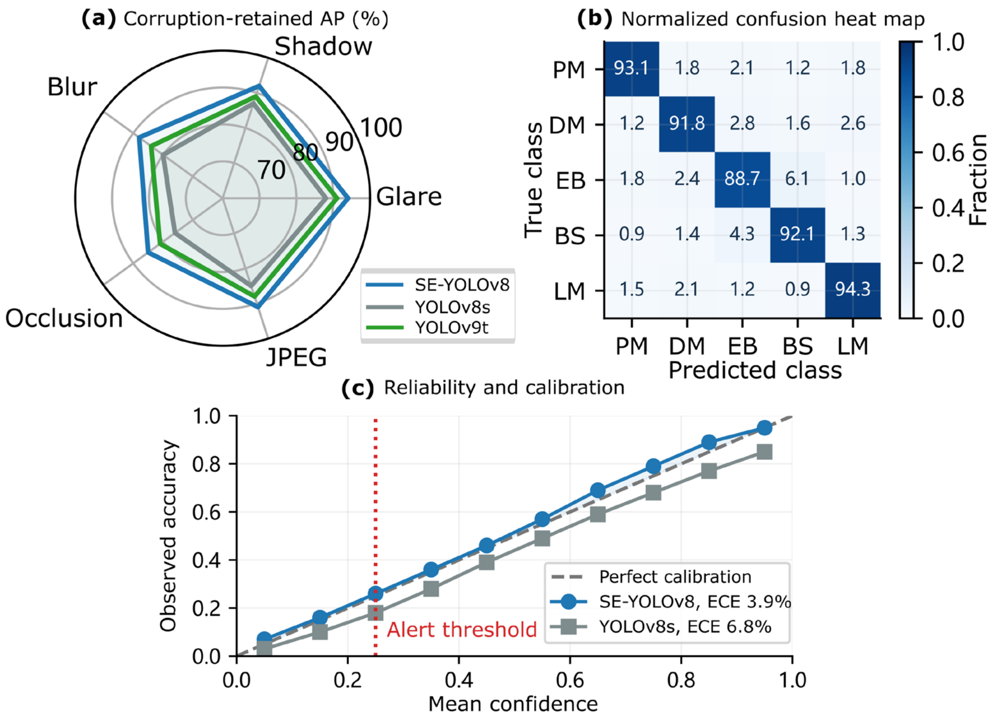

Figure 6. Robustness and confidence diagnostics: (a) corruption-resistance radar; (b) normalized class-confusion heat map; (c) reliability diagrams and expected calibration error.

Error review on 500 difficult images separated false positives into vein junctions, glare patches, senescent edges, irrigation hardware, and soil or background texture. Their counts fell from 118, 96, 73, 41, and 35 for YOLOv8s to 71, 49, 55, 29, and 27 for SE-YOLOv8. Remaining false negatives were dominated by sub-12-pixel lesions and lesions hidden behind overlapping leaves. Class-specific threshold tuning improved powdery-mildew recall by 0.8 points but increased false alerts, so the common threshold was retained for the principal comparison. Such explicit error accounting is preferable to relying only on aggregate mAP [30].

Figure 7 presents the deployment evidence. Figure 7 (a) shows throughput values of 48.6,41.2, and 22.8 frames /s for TensorRT FP16, ONNX FP16, and PyTorch FP32 on the Orin NX. Figure 7(b) decomposes the 20.6 ms TensorRT latency into 1.7 ms for preprocessing, 16.4 ms for inference, and 2.5 ms for suppression and coordinate mapping. Figure 7(c) plots energy per frame against $\mathrm{mAP}_{50:95}$: SE-YOLOv8 requires 0.31 J, compared with 0.27 J for YOLOv8s and 0.46 J for RT-DETR-R18. The 0.04 J increase over the baseline is acceptable for fixed greenhouse power and is offset by materially higher lesion recall [31].

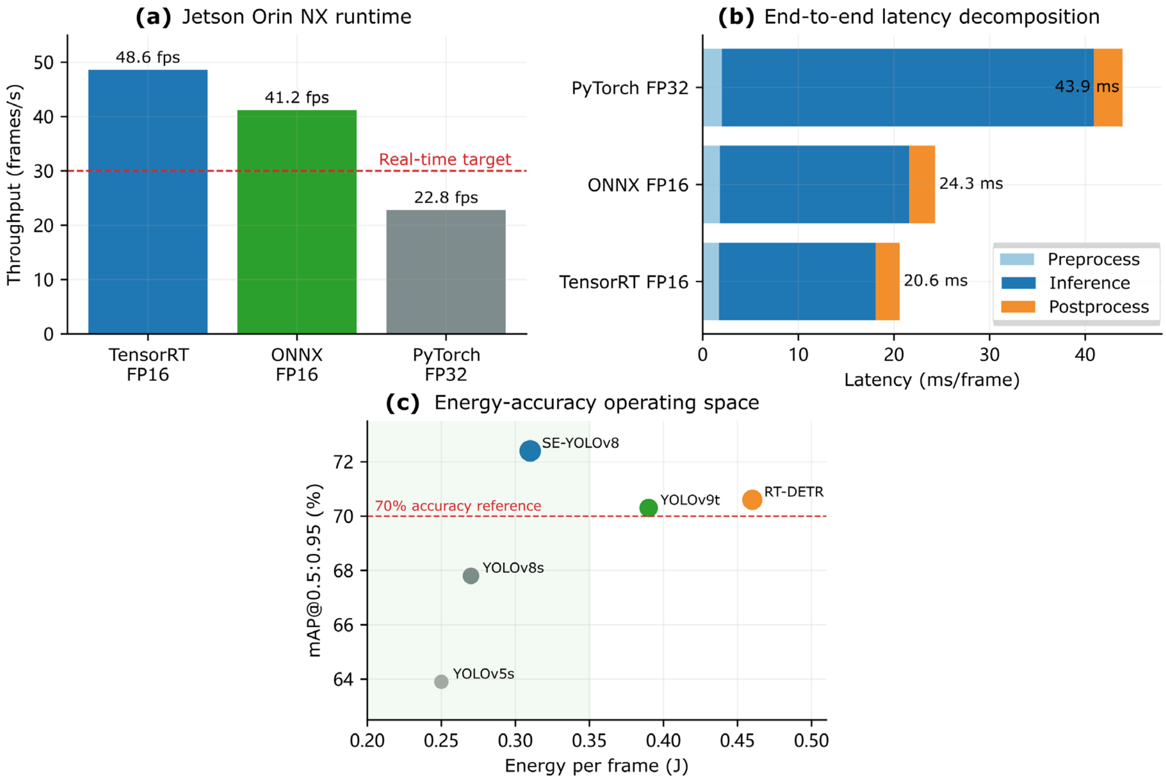

Figure 7. Embedded deployment analysis: (a) throughput by runtime and precision mode; (b) end-to-end latency decomposition; (c) energy-accuracy trade-off across detectors.

Cross-greenhouse testing left one site unseen during training. The model retained 89.7% recall and 69.1% $\mathrm{mAP}_{50:95}$, representing declines of 2.9 and 3.3 points from the mixed-site test. Fine-tuning with 300 labeled images from the new site recovered 2.4 mAP points. Performance was weakest on lettuce downy mildew captured under purple supplemental lighting, indicating a remaining spectral-domain dependency [32]. Simple intensity normalization improved that subset but reduced midday precision, so it was not included in the final pipeline. Future operational trials should associate detections with plant identity and temporal persistence, because repeated evidence can suppress transient reflections without lowering frame-level sensitivity [33].

Threshold behavior was evaluated because a greenhouse system may prioritize sensitivity during an outbreak and precision during routine surveillance. At a confidence threshold of 0.15, SE-YOLOv8 reached 95.1% recall with 90.2% precision. At 0.40, precision increased to 96.7% while recall decreased to 88.4%. The F1 optimum occurred near 0.27, close to the fixed threshold of 0.25. YOLOv8s required a lower threshold to reach comparable recall and consequently generated 1.62 times as many false alerts per 1,000 images. The proposed model therefore provides a wider useful operating interval rather than a gain confined to one threshold.

Occlusion analysis used visible-fraction bins estimated by reviewers on a 900 -lesion subset. Recall was 96.0% above 80% visibility, 92.8% at $60-80\%$, 87.5% at $40-60\%$, and 71.9% below 40%. Corresponding baseline values were 93.1%, 87.6%, 78.3%, and 63.0%. Contextual fusion is helpful when lesion texture is incomplete, yet performance still drops sharply in the most severe bin. False negatives in that bin often have no distinctive central texture, so recovering them may require adjacent frames or alternate viewpoints rather than further enlargement of the detector.

The confusion heat map also exposed asymmetric class errors. Early blight was mislabeled as bacterial spot more often than the reverse because small early-blight lesions may not yet show concentric rings. Confusion between downy mildew and leaf mold increased under supplemental lighting, which altered their yellow-green contrast. Healthy hard negatives were rarely assigned to powdery mildew after coordinate attention was added, but wet specular patches remained a source of bacterial-spot false positives. These observations support collecting temporally paired wet and dry images rather than applying stronger generic color jitter.

Deployment profiling included memory and thermal stability. The TensorRT engine occupied 39.8 MB, and peak device memory during inference was 1.21 GB. Over a two-hour continuous run, throughput varied between 47.9 and 49.1 frames/s, and no thermal throttling occurred under active cooling. The baseline engine required 0.96 GB and reached 57.8 frames /s, while RT-DETR-R18 required 1.74 GB. For a camera sampled at five frames /s, every model exceeds the acquisition rate; the practical difference is that SE-YOLOv8 leaves sufficient capacity for tracking, alert aggregation, and image logging on the same device.

End-to-end field latency also depends on communication and scheduling. With local inference, the median time from image capture to an alert record was 47 ms, including camera transfer and database writing. A server pipeline operating over greenhouse Wi-Fi showed a median of 126 ms and a 95th percentile of 418 ms during network contention. Although both configurations satisfy minute-scale scouting requirements, local execution avoids intermittent upload failure and limits the transmission of high-resolution crop imagery. The selected model therefore targets edge inference, with optional synchronization of compact detections and selected evidence frames.

A practical alert simulation grouped detections by camera zone over ten-minute windows. Requiring two detections of the same class in a zone reduced isolated false alerts by 61% while delaying the first notification by a median of 18 seconds. This temporal rule was intentionally excluded from detector metrics, but it illustrates how calibrated framelevel confidence can support stable operational decisions. Because the proposed model preserves confidence ranking under corruption, its outputs are better suited to such aggregation than those of an equally accurate but poorly calibrated detector.

Model quantization was explored as a deployment option. Post-training INT8 quantization increased throughput to 61.7 frames/s and reduced the engine size to 24.6 MB, but $\mathrm{mAP}_{50:95}$ fell by 1.6 points and AP_S fell by 2.8 points. Calibration with a small-lesion-enriched image set recovered $0.7\mathrm{AP}_S$ points, showing that representative activation ranges are especially important for the high-resolution branch. Because FP16 already met the target rate with a smaller accuracy penalty, it was selected for the principal system. INT8 remains attractive for passively cooled devices if calibration data cover glare, dark leaves, and dense tiny lesions.

Edge-device comparisons were again conducted with a batch size of one and dynamic frequency adjustment disabled. SE-YOLOv8, YOLOv8s and RT-DETR-R18 had average power consumption of 15.2 W, 14.7 W and 16.8 W, respectively, during prolonged inference. The extra P2 computation in the proposed network raises memory traffic more than arithmetic utilisation; thus, its latency increase is relatively larger than what would be expected from an increase in parameters alone. Kernel fusion in TensorRT reduces this cost by combining projection, normalisation and activation operations. Profiling shows that the P2 refinement block is a more promising target for future structured pruning.

Conclusion

This paper developed SE-YOLOv8 for the detection of small crop disease spots in greenhouse vegetable pictures. A high-resolution prediction path, adaptive bidirectional fusion, lightweight coordinate attention, and scale-balanced optimisation are used to form a small one-stage detector. The resulting structure can keep the fine lesion information, add contextual discrimination, and is still compatible with edge-oriented export and real-time inspection.

Only RGB observations of a small number of crops, greenhouses, and camera positions are available in this study. Very small or heavily occluded lesions can be mistaken for leaf texture, and abnormal artificial light introduces a domain shift. Box annotations do not describe the extent of lesions as precisely as masks, and single-frame evaluation does not take into account temporal continuity in routine monitoring.

Future work will expand the detector to support multiple spectra and videos, introduce uncertainty-aware active learning, and perform temporal association at the plant level. Domain-general training across seasons and greenhouse materials should reduce reliance on local illumination. Jointly detect and set the severity for the alarms using these segmentation labels to link them with corresponding treatment measures.

Author Contributions

Milorad Vlahović contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision, project administration, and funding acquisition. Gojko Vasiljev contributes to validation, analysis, investigation, data collection, draft preparation. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Deshpande, R., & Patidar, H. (2022). Tomato plant leaf disease detection using generative adversarial network and deep convolutional neural network. The Imaging Science Journal, 70(1), 1-9. https://doi.org/10.1080/13682199.2022.2161696

[2] Kaur, P., Harnal, S., Gautam, V., Singh, M. P., & Singh, S. P. (2022). An approach for characterization of infected area in tomato leaf disease based on deep learning and object detection technique. Engineering Applications of Artificial Intelligence, 115, 105210. https://doi.org/10.1016/j.engappai.2022.105210

[3] Zeng, W., & Li, M. (2020). Crop leaf disease recognition based on Self-Attention convolutional neural network. Computers and Electronics in Agriculture, 172, 105341. https://doi.org/10.1016/j.compag.2020.105341

[4] Sujatha, R., Chatterjee, J. M., Jhanjhi, N., & Brohi, S. N. (2021). Performance of deep learning vs machine learning in plant leaf disease detection. Microprocessors and Microsystems, 80, 103615. https://doi.org/10.1016/j.micpro.2020.103615

[5] Lakshmi, R. K., & Savarimuthu, N. (2022). PLDD—A Deep Learning-Based Plant Leaf Disease Detection. IEEE Consumer Electronics Magazine, 11(3), 44-49. https://doi.org/10.1109/mce.2021.3083976

[6] Bhattania, Y., Singhal, P., & Agarwal, T. (2022). Plant Leaf Disease Detection Using Deep Learning. International Journal for Research in Applied Science and Engineering Technology, 10(5), 2518-1523. https://doi.org/10.22214/ijraset.2022.42892

[7] Rashwan, S. A., & Elteir, M. K. (2022). Plant leaf disease detection using deep learning on mobile devices. International Journal of Computational Vision and Robotics, 12(2), 156. https://doi.org/10.1504/ijcvr.2022.121151

[8] Saleem, M. H., Potgieter, J., & Arif, K. M. (2022). A Performance-Optimized Deep Learning-Based Plant Disease Detection Approach for Horticultural Crops of New Zealand. IEEE Access, 10, 89798-89822. https://doi.org/10.1109/access.2022.3201104

[9] Sai Reddy, B., & Neeraja, S. (2022). Plant leaf disease classification and damage detection system using deep learning models. Multimedia Tools and Applications, 81(17), 24021-24040. https://doi.org/10.1007/s11042-022-12147-0

[10] Musa, A., Hassan, M., Hamada, M., & Aliyu, F. (2022). Low-Power Deep Learning Model for Plant Disease Detection for Smart-Hydroponics Using Knowledge Distillation Techniques. Journal of Low Power Electronics and Applications, 12(2), 24. https://doi.org/10.3390/jlpea12020024

[11] S, B., R, S., R, A., M, T., & Lingisetti, R. B. (2022). Disease Detection in Plant Leaf using LNet Based on Deep Learning. International Journal of Engineering Trends and Technology, 70(9), 64-75. https://doi.org/10.14445/22315381/ijett-v70i9p207

[12] Sakkarvarthi, G., Sathianesan, G. W., Murugan, V. S., Reddy, A. J., Jayagopal, P., & Elsisi, M. (2022). Detection and Classification of Tomato Crop Disease Using Convolutional Neural Network. Electronics, 11(21), 3618. https://doi.org/10.3390/electronics11213618

[13] Pandian, J. A., Kanchanadevi, K., Kumar, V. D., Jasińska, E. B., Goňo, R. R., Leonowicz, Z., & Jasiński, M. (2022). A Five Convolutional Layer Deep Convolutional Neural Network for Plant Leaf Disease Detection. Electronics, 11(8), 1266. https://doi.org/10.3390/electronics11081266

[14] Paymode, A. S., & Malode, V. B. (2022). Transfer Learning for Multi-Crop Leaf Disease Image Classification using Convolutional Neural Network VGG. Artificial Intelligence in Agriculture, 6, 23-33. https://doi.org/10.1016/j.aiia.2021.12.002

[15] Jadhav, S., & Lal, A. M. (2022). Multi-Class Plant Leaf Disease Detection Using a Deep Convolutional Neural Network. International Journal of Information System Modeling and Design, 13(1), 1-14. https://doi.org/10.4018/ijismd.315126

[16] Tarek, H., Aly, H., Eisa, S., & Abul-Soud, M. (2022). Optimized Deep Learning Algorithms for Tomato Leaf Disease Detection with Hardware Deployment. Electronics, 11(1), 140. https://doi.org/10.3390/electronics11010140

[17] Aggarwal, A. K. (2022). Biological Tomato Leaf Disease Classification using Deep Learning Framework. International Journal of Biology and Biomedical Engineering, 16, 241-244. https://doi.org/10.46300/91011.2022.16.30

[18] Anandhakrishnan, T., & Jaisakthi, S. M. (2022). Deep Convolutional Neural Networks for image based tomato leaf disease detection. Sustainable Chemistry and Pharmacy, 30, 100793. https://doi.org/10.1016/j.scp.2022.100793

[19] Ashwinkumar, S., Rajagopal, S., Manimaran, V., & Jegajothi, B. (2022). Automated plant leaf disease detection and classification using optimal MobileNet based convolutional neural networks. Materials Today: Proceedings, 51, 480-487. https://doi.org/10.1016/j.matpr.2021.05.584

[20] Min, K., Lee, G. H., & Lee, S. W. (2022). Attentional feature pyramid network for small object detection. Neural Networks, 155, 439-450. https://doi.org/10.1016/j.neunet.2022.08.029

[21] Zhang, H., Du, Q., Qi, Q., Zhang, J., Wang, F., & Gao, M. (2022). A recursive attention-enhanced bidirectional feature pyramid network for small object detection. Multimedia Tools and Applications, 82(9), 13999-14018. https://doi.org/10.1007/s11042-022-13951-4

[22] Jiao, L., Kang, C., Dong, S., Chen, P., Li, G., & Wang, R. (2022). An attention-based feature pyramid network for single-stage small object detection. Multimedia Tools and Applications, 82(12), 18529-18544. https://doi.org/10.1007/s11042-022-14159-2

[23] S, P., & A, F. S. D. (2021). Image based Plant leaf disease detection using Deep learning. International Journal of Computer Communication and Informatics, 3(1), 53-65. https://doi.org/10.34256/ijcci2115

[24] Ramacharan, D. S. (2021). A 3-Stage Method for Disease Detection of Cotton Plant Leaf using Deep Learning CNN Algorithm. International Journal for Research in Applied Science and Engineering Technology, 9(VII), 2503-2510. https://doi.org/10.22214/ijraset.2021.36913

[25] Shrimali, S. (2021). PlantifyAI: A Novel Convolutional Neural Network Based Mobile Application for Efficient Crop Disease Detection and Treatment. Procedia Computer Science, 191, 469-474. https://doi.org/10.1016/j.procs.2021.07.059

[26] Singh, A. K., & Chaurasia, B. (2021). Plant Leaf Disease Detection using Convolutional Neural Network and Random Forest Classifier. International Journal of Innovations in Engineering and Science, 6(10), 204. https://doi.org/10.46335/ijies.2021.6.10.42

[27] Wu, Y., Xu, L., & D. Goodman, E. (2021). Tomato Leaf Disease Identification and Detection Based on Deep Convolutional Neural Network. Intelligent Automation & Soft Computing, 28(2), 561-576. https://doi.org/10.32604/iasc.2021.016415

[28] Wagle, S. A., & R, H. (2021). A Deep Learning-Based Approach in Classification and Validation of Tomato Leaf Disease. Traitement du Signal, 38(3), 699-709. https://doi.org/10.18280/ts.380317

[29] Kumar, N. (2021). Double Channel CNN Based Tomato Plant Leaf Disease Detection. Quaid-e-Awam University Research Journal of Engineering Science & Technology, 19(2), 1. https://doi.org/10.52584/qrj.1902.01

[30] Tiwari, V., Joshi, R. C., & Dutta, M. K. (2021). Dense convolutional neural networks based multiclass plant disease detection and classification using leaf images. Ecological Informatics, 63, 101289. https://doi.org/10.1016/j.ecoinf.2021.101289

[31] Mathew, M. P., & Mahesh, T. Y. (2021). Leaf-based disease detection in bell pepper plant using YOLO v5. Signal, Image and Video Processing, 16(3), 841-847. https://doi.org/10.1007/s11760-021-02024-y
[32] Kim, M., Jeong, J., & Kim, S. (2021). ECAP-YOLO: Efficient Channel Attention Pyramid YOLO for Small Object Detection in Aerial Image. Remote Sensing, 13(23), 4851. https://doi.org/10.3390/rs13234851
[33] Chakali, R. (2020). Effective pomegranate plant leaf disease detection using deep learning. International Journal of Circuit, Computing and Networking, 1(2), 08-10. https://doi.org/10.33545/27075923.2020.v1.i2a.13