



FlexIoU: Modularity and Flexibility IoU for Bounding Box Regression in Agricultural Disease Detection

Naphat Keawpibal^{1,2*}, Ajaree Naco¹, Kanida Sinmai¹, and Noppamas Pukkhem¹

¹ Faculty of Science and Digital Innovation, Thaksin University, Phatthalung, 93210, Thailand

² Research Center for Indigenous Plant Species in ASEAN Culture, Thaksin University, Phatthalung, 93210, Thailand

* Correspondence: naphat.k@tsu.ac.th

Citation:

Keawpibal, N.; Naco, A.; Sinmai, K.; Pukkhem, N. FlexIoU: Modularity and flexibility IoU for bounding box regression in agricultural disease detection. *ASEAN J. Sci. Tech. Report.* **2026**, *29*(7), e263179. <https://doi.org/10.55164/ajstr.v29i7.263179>

Article history:

Received: January 9, 2026

Revised: June 3, 2026

Accepted: June 6, 2026

Available online: June 14, 2026

Publisher's Note:

This article is published and distributed under the terms of the Thaksin University.

Abstract: Accurate bounding-box regression plays a crucial role in object detection, particularly in agricultural disease detection, where target objects are often small, irregular, and weakly bounded. Although Intersection over Union (IoU)-based losses such as CIoU and EIoU have improved localization performance, their fixed geometric penalty formulations limit adaptability to domain-specific object characteristics. In this work, we propose FlexIoU, a flexible and lightweight IoU-based regression loss that introduces tunable geometric penalties to enhance localization robustness. FlexIoU provides explicit control over center-distance, width, and height penalties, enabling more adaptive regression behavior for irregular lesion shapes commonly observed in agricultural imagery. The proposed loss integrates seamlessly into the YOLO training pipeline as a drop-in replacement for conventional IoU-based losses, without modifying classification or distribution focal loss components, thereby preserving inference efficiency. We evaluate FlexIoU within a YOLOv11 detection framework on a tomato leaf disease dataset under identical experimental settings and across three random seeds to ensure statistical reliability. Experimental results demonstrate that FlexIoU consistently outperforms the default YOLOv11 regression objective as well as CIoU and EIoU in terms of mAP@0.5:0.95 and overall detection balance, while maintaining stable inference speed comparable to baseline losses. These findings indicate that FlexIoU provides an effective and practical solution for improving bounding-box regression in agricultural disease detection. While the results demonstrate promising robustness across two tomato disease datasets, further validation on additional agricultural and general object detection datasets is required.

Keywords: Object detection; IoU-based loss; FlexIoU; agricultural disease detection; precision agriculture

1. Introduction

Precision agriculture has emerged as a key strategy for increasing crop productivity, reducing resource consumption, and improving early detection of plant diseases [1-3]. Accurate identification and localization of diseased regions on leaves enable timely interventions, minimize pathogen spread, and support automated decision-making in smart farming systems. With the rapid advancement of deep learning, object detection models—especially the YOLO family [4-10], have become widely adopted for real-time object detection. Recent studies have successfully applied YOLO-based models to agricultural disease detection tasks, including tomato disease recognition and plant pest

identification [11-15]. However, the effectiveness of these models depends not only on architectural design but also heavily on the loss function used to guide bounding-box regression during training. In plant-disease imagery, lesions often appear as small, irregular, partially occluded, or non-uniformly shaped regions [3,11]. Leaves vary widely in size, aspect ratio, orientation, and deformation, creating additional challenges for bounding-box localization. These characteristics underscore the importance of a regression loss that effectively penalizes geometric discrepancies between predicted and ground-truth boxes [16]. A loss function that behaves well on general object detection datasets may perform poorly in agricultural settings, where lesion boundaries are subtle and precise localization is crucial for accurate disease diagnosis [3,11,14].

The most fundamental metric for bounding-box regression is the Intersection over Union (IoU), which measures the overlap ratio between predicted and ground-truth boxes [16-18]. Although IoU is widely used, it suffers from a major limitation: the gradient vanishes when boxes do not overlap, leading to unstable early-stage optimization. To address this, several extensions have been proposed. Complete IoU (CIoU) [19] incorporates center-distance and aspect-ratio terms to accelerate convergence, while Enhanced IoU (EIoU) [20] decouples width and height deviations for more stable regression. These methods have improved performance on general object detection tasks, yet they still rely on fixed penalty formulations [21, 22]. Such fixed penalties may not generalize well across agricultural datasets where object geometry varies significantly. Moreover, CIoU and EIoU lack explicit control over the influence of different geometric components, limiting their adaptability to datasets with imbalanced lesion sizes or highly elongated leaf structures.

Beyond distance- and scale-decomposition, recent work incorporates directional or angular components to enhance regression performance. SIoU [23] introduces an angle-based penalty that encourages predicted boxes to move toward the ground-truth direction more efficiently, potentially accelerating convergence. While SIoU improves early-stage regression, its angle-based formulation introduces additional computational cost and is sensitive to noisy boxes. MPDIoU [24] and Minimum-Point-Distance approaches aim to unify area overlap, center-distance, and width–height deviation using geometric projections. These losses demonstrate competitive performance in certain domains; however, they typically involve more complex geometric transformations and do not provide explicit modularity or tunable weighting for different penalty terms. In oriented object detection, such as in aerial imagery and scene text recognition, the limitations of axis-aligned IoU become more pronounced. Losses such as Rotated IoU [25], FPDIoU [26], and EPIoU loss [27] account for rotation angles and corner distances to guide regression. Although effective for rotated bounding boxes, these methods introduce substantial computational overhead, making them unsuitable for mainstream real-time detectors like YOLO unless specifically targeting rotated domains.

Given these limitations, there is a strong need for a flexible, tunable IoU-based regression loss that can adapt to the diverse geometric characteristics of agricultural imagery. A loss function that allows independent control over center distance, aspect ratio, and side-length deviations would provide better localization performance and enable tailored optimization strategies for different plant species or disease types. Such tunability is particularly valuable when working with datasets that exhibit class imbalance, size variation, or high lesion irregularity, all of which are common in real-world farming scenarios.

To address these limitations, this study introduces FlexIoU, a modular, tunable IoU-based regression loss that enhances bounding-box regression for plant-disease detection. FlexIoU decomposes the geometric relationship into interpretable, weighted components, allowing for flexible adaptation. Extensive evaluations on a benchmark tomato leaf disease dataset, along with preliminary validation, demonstrate that FlexIoU achieves superior localization robustness and improved regression performance compared to state-of-the-art IoU-based losses, such as CIoU and EIoU.

The main contributions of this work are as follows:

1. We introduce a modular geometric formulation that enhances regression stability and interpretability compared to existing IoU-based losses.
2. We propose FlexIoU, a tunable IoU-based regression loss that allows explicit control over center-distance, aspect-ratio, width, and height penalties, enabling flexible adaptation to agricultural imagery.

3. FlexIoU integrates seamlessly with the YOLO loss pipeline without modifying the classification or DFL components, enabling a drop-in replacement for CIoU.
4. We evaluate FlexIoU on agricultural datasets, including tomato leaf disease imagery, demonstrating improved localization robustness for small and irregular disease symptoms.
5. We conduct comparative experiments across multiple random seeds, demonstrating consistent regression behavior and providing insights into the impact of tuning geometric weights.

2. Materials and Methods

This section describes the research framework, baseline loss functions, and the proposed FlexIoU formulation. The overall methodology follows a unified training–evaluation pipeline in which each loss function is tested under identical experimental settings and repeated across three random seeds to ensure statistical stability.

2.1 Research Framework

This study follows a unified experimental framework designed to systematically evaluate the impact of different IoU-based loss functions on bounding-box regression performance. The overall pipeline is illustrated in Figure 1 and consists of dataset preparation, model training with interchangeable regression losses, and performance evaluation on held-out test sets.

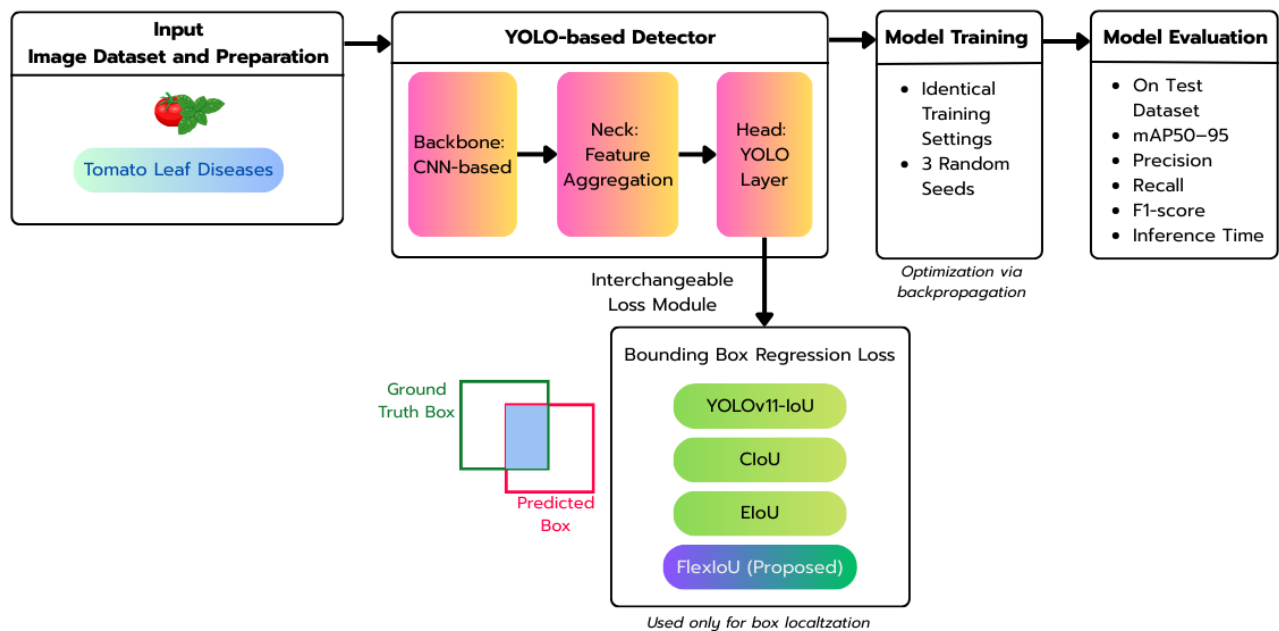


Figure 1. Overview of the research framework.

Given an input image, object detection is performed using a YOLO-based architecture, which jointly predicts class probabilities and bounding-box parameters. The dataset is first divided into training, validation, and test subsets to ensure unbiased evaluation. During training, images from the training set are passed through the YOLO backbone and detection head to generate predicted bounding boxes. These predictions are then compared against the corresponding ground-truth annotations through an IoU-based regression loss.

To enable a fair comparison, the regression loss component is treated as a modular element within the YOLO training pipeline. Specifically, four different loss functions—IoU, CIoU, EIoU, and the proposed FlexIoU—are independently integrated into the bounding-box regression branch while keeping all other components unchanged. Classification loss and objectness loss follow the default YOLO configuration, ensuring that observed performance differences are primarily due to the regression loss design.

Each model is trained using identical hyperparameters and optimization settings, and the training process is repeated using three fixed random seeds. This multi-seed strategy reduces stochastic variance and

provides a more reliable assessment of loss stability and convergence behavior. After training, the final models are evaluated on the test set, where detection performance is measured using standard object detection metrics such as mean Average Precision (mAP) and inference time. This framework allows the proposed FlexIoU loss to be evaluated under controlled, reproducible conditions, while enabling direct comparisons with established IoU-based losses. By isolating the effect of the regression loss within the YOLO architecture, the framework provides clear insight into how different geometric penalty formulations influence localization accuracy, convergence stability, and generalization performance.

2.2 Baseline IoU-based Loss Functions

Bounding-box regression is a core component of modern object detection frameworks, determining how effectively a detector localizes objects of interest. In YOLO-based models, this regression traditionally relies on IoU-driven losses because Intersection over Union remains the most intuitive and widely adopted measure of spatial overlap between predicted and ground-truth bounding boxes. However, IoU alone is insufficient for ensuring stable optimization, motivating the development of extended formulations such as CIoU and EIoU. In this section, we describe these baseline losses in detail to establish a clear foundation for evaluating the proposed FlexIoU.

2.2.1. IoU Loss

Intersection over Union (IoU) evaluates the geometric overlap between two bounding boxes:

$$IoU = \frac{|B_p \cap B_g|}{|B_p \cup B_g|} \quad (1)$$

where $B_p \cap B_g$ refers to the intersection region between the two boxes and $B_p \cup B_g$ is their union.

The loss function corresponding to this measure is:

$$\mathcal{L}_{IoU} = 1 - IoU \quad (2)$$

This loss directly penalizes low overlap and reaches its minimum when the predicted and ground-truth boxes coincide exactly. However, despite its conceptual simplicity, IoU suffers from two well-known limitations. First, when B_p and B_g do not intersect, $|B_p \cap B_g| = 0$, leading to $IoU = 0$ and a zero gradient with respect to box coordinates. In such cases, the model receives no meaningful signal to guide the predicted box toward the target, which can severely slow convergence during early training. Second, IoU depends solely on the overlapping area and does not encode any information about the distance between box centers, their relative position, or their shape similarity. As a result, two pairs of boxes may yield the same IoU value even if one pair is well-aligned but slightly shifted, while the other is far apart but happens to overlap marginally. These weaknesses often lead to slow and sometimes unstable optimization, especially in datasets containing small or sparsely distributed objects, such as agricultural disease lesions, where slight misalignments can drastically affect detection quality.

2.2.2. CIoU Loss

To overcome the limitations of IoU, Zheng et al. introduced the Complete IoU (CIoU) loss, which augments the original formulation with additional geometric constraints. While IoU considers only the overlapping area, CIoU integrates (1) the distance between bounding-box centers and (2) the discrepancy in aspect ratio. These additional terms enable more stable and discriminative gradients during optimization.

Let $p = (x_p, y_p)$ and $g = (x_g, y_g)$ denote the center coordinates of the predicted and ground-truth bounding boxes, respectively. Let $\rho(p, g)$ represent the Euclidean distance between the two centers:

$$\rho(p, g) = \sqrt{(x_p - x_g)^2 + (y_p - y_g)^2} \quad (3)$$

Let c be the diagonal length of the smallest enclosing box that contains both the predicted and ground-truth boxes. CIoU is defined as:

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(p,g)}{c^2} + \alpha v, \quad (4)$$

where v represents the aspect-ratio penalty term, defined as

$$v = \frac{4}{\pi^2} \left(\tan^{-1} \frac{w_g}{h_g} - \tan^{-1} \frac{w_p}{h_p} \right)^2 \quad (5)$$

Here,

- w_p, h_p : width and height of the predicted box,
- w_g, h_g : width and height of the ground-truth box

The first additional term, ρ^2/c^2 , directly penalizes deviations in box location, providing meaningful gradients even when IoU is zero (i.e., when boxes do not overlap). The second term, αv , penalizes aspect-ratio discrepancies, with α dynamically balancing the contribution of the shape penalty to ensure stable and proportional regression.

Despite its advantages, CIoU employs fixed, hard-coded balancing between its distance and aspect-ratio components. Models trained on domains with highly variable object shapes—such as diseased leaves, which exhibit irregular lesions—may require different penalization strengths, but CIoU does not allow such adaptation. This motivates the development of more flexible formulations, such as the proposed FlexIoU.

2.2.3. EIoU Loss

Extended IoU (EIoU) addresses another limitation of CIoU: its aspect-ratio penalty indirectly influences width and height without explicitly optimizing them. In many real-world cases—especially agricultural images where lesions vary dramatically in width, elongation, and local deformation—separating width and height regression can significantly improve convergence.

Let (w_p, h_p) and (w_g, h_g) denote the width-height pairs of the predicted and ground-truth boxes, respectively. Let w_c and h_c be the width and height of the smallest enclosing box containing both boxes. EIoU is defined as:

$$\mathcal{L}_{EIoU} = 1 - IoU + \frac{(w_p - w_g)^2}{w_c^2} + \frac{(h_p - h_g)^2}{h_c^2} \quad (6)$$

The additional terms explicitly penalize width and height differences:

- $\frac{(w_p - w_g)^2}{w_c^2}$: width deviation
- $\frac{(h_p - h_g)^2}{h_c^2}$: height deviation

This decomposition enables direct supervision of the predicted box's shape, generally yielding faster convergence and more stable regression than CIoU, especially for narrow, elongated, or irregularly shaped targets. However, similar to CIoU, EIoU employs fixed penalty strengths, applying the same geometric constraints uniformly across datasets regardless of object morphology or scale variation. Thus, although EIoU improves optimization behavior, it still lacks the adaptability required for domain-specific tuning—an issue that the proposed FlexIoU addresses by introducing user-configurable weighting for each geometric component.

2.3 Proposed FlexIoU Loss

Accurate bounding box regression is fundamental to YOLO-based object detectors, as localization quality directly influences both classification confidence and final detection accuracy. Although IoU-based loss functions have been widely adopted to improve localization by incorporating geometric constraints, most existing formulations employ fixed penalty terms that treat all geometric components equally across different datasets. In practice, however, object characteristics such as scale, shape, and spatial distribution can vary substantially, particularly in agricultural imagery where objects often exhibit irregular shapes, elongated structures, or small target regions. Motivated by this observation, we propose FlexIoU, a flexible IoU-based loss function that allows explicit and independent control over key geometric components involved in

bounding box regression. Rather than enforcing a rigid formulation, FlexIoU enables adaptive weighting of center alignment and shape-related penalties, thereby improving robustness across diverse detection scenarios while preserving the efficiency of the original YOLO training pipeline.

2.3.1 Formulation of FlexIoU

Let $B_p = (x_p, y_p, w_p, h_p)$ denote the predicted bounding box and $B_g = (x_g, y_g, w_g, h_g)$ denote the corresponding ground-truth bounding box, where (x, y) represent the box center coordinates, and w and h denote width and height, respectively. FlexIoU is defined as a similarity score composed of the standard IoU term and multiple weighted geometric penalties:

$$\text{FlexIoU}(B_p, B_g) = \text{IoU} - (\lambda_c \mathcal{L}_{\text{center}} + \lambda_a \mathcal{L}_{\text{aspect}} + \lambda_w \mathcal{L}_{\text{width}} + \lambda_h \mathcal{L}_{\text{height}}) \quad (7)$$

where λ_c , λ_a , λ_w and λ_h are tunable weight coefficients that regulate the relative importance of each penalty term. This formulation decomposes bounding box regression into interpretable components, allowing each term to contribute independently to the optimization process.

2.3.2 IoU Term

The Intersection over Union (IoU) between B_p and B_g is computed as

$$\text{IoU} = \frac{|B_p \cap B_g|}{|B_p \cup B_g|} \quad (8)$$

The term IoU serves as the primary overlap-driven objective, directly encouraging the predicted box to maximize its intersection with the ground truth. However, IoU alone provides limited gradient information when predicted and ground-truth boxes do not overlap, which often occurs during early training stages. Consequently, additional geometric constraints are required to better guide the regression process.

2.3.3 Center Distance Penalty

To address spatial misalignment between bounding boxes, FlexIoU incorporates a center distance penalty defined as

$$\mathcal{L}_{\text{center}} = \frac{(x_p - x_g)^2 + (y_p - y_g)^2}{c^2} \quad (9)$$

where (x_p, y_p) and (x_g, y_g) denote the center coordinates of the predicted and ground-truth boxes, respectively. The normalization term c^2 represents the squared diagonal length of the smallest enclosing (convex) box that covers both B_p and B_g , and is defined as $c^2 = w_c^2 + h_c^2$, where w_c and h_c denote the width and height of the smallest enclosing box, respectively.

This normalization ensures scale invariance and prevents excessively large gradients for objects of different sizes. By explicitly penalizing large center offsets, this term provides meaningful gradient signals even when the predicted and ground-truth boxes do not overlap. The influence of spatial alignment can be adjusted via the weighting coefficient λ_c , which is particularly beneficial for small or sparsely distributed objects commonly found in agricultural imagery.

2.3.4 Aspect Ratio Penalty

To account for shape discrepancies caused by aspect-ratio differences, FlexIoU adopts an aspect-ratio penalty inspired by CIoU.

$$\mathcal{L}_{\text{aspect}} = \alpha \cdot \frac{4}{\pi^2} \left(\tan^{-1} \frac{w_g}{h_g} - \tan^{-1} \frac{w_p}{h_p} \right)^2 \quad (10)$$

where α is a dynamic balancing factor following the original CIoU formulation, designed to regulate the contribution of the aspect-ratio term relative to IoU, this penalty encourages consistent proportional scaling between predicted and target boxes, reducing oscillation caused by shape mismatch.

Although the CIoU formulation inspires the aspect-ratio term, FlexIoU extends its role through an independent weighting mechanism rather than introducing a new aspect-ratio equation. Unlike CIoU, where the aspect-ratio penalty is incorporated as a fixed component of the regression loss, FlexIoU allows the contribution of the aspect-ratio term to be explicitly adjusted through λ_a . This design enables the aspect-ratio

penalty to be strengthened, reduced, or disabled entirely ($\lambda_a = 0$) based on object characteristics and dataset requirements. Consequently, the novelty of FlexIoU lies not in redefining the aspect-ratio penalty itself, but in transforming the fixed CIoU penalty structure into a flexible weighting framework that allows center-distance, aspect-ratio, width, and height penalties to be independently controlled. This provides greater adaptability than the fixed-weighting strategy used by CIoU.

2.3.5 Width and Height Penalties

In addition to spatial alignment, accurate shape estimation is essential for precise localization. FlexIoU therefore introduces independent penalties for width and height discrepancies:

$$\mathcal{L}_{width} = \frac{(w_p - w_g)^2}{w_c^2}, \mathcal{L}_{height} = \frac{(h_p - h_g)^2}{h_c^2} \quad (11)$$

where w_c and h_c represent the width and height of the smallest enclosing box, respectively.

Unlike aspect-ratio-based penalties that couple width and height into a single term, this decomposition allows FlexIoU to handle anisotropic shape variations more effectively. In agricultural detection tasks, objects such as leaves or disease regions often exhibit elongated or irregular geometries, where width and height errors may have different semantic importance. The independent weighting coefficients λ_w and λ_h enable targeted shape refinement without imposing unnecessary constraints.

2.3.6 Integration into YOLO Training

During training, FlexIoU replaces only the bounding-box regression component of the default YOLOv11 localization loss. The Distribution Focal Loss (DFL) remains unchanged and follows the standard YOLOv11 implementation. The overall training objective is expressed as

$$\mathcal{L}_{Total} = \lambda_{box}(1 - FlexIoU) + \lambda_{cls}\mathcal{L}_{cls} + \lambda_{dfl}\mathcal{L}_{dfl} \quad (12)$$

In the baseline YOLOv11 implementation, the localization term is computed using the default IoU-based regression loss together with DFL. In the proposed approach, only the localization term is replaced with FlexIoU, while all other training components and hyperparameters are kept identical for a fair comparison.

No additional prediction branches or inference-time operations are introduced. As a result, it preserves the computational efficiency and inference speed of the original YOLO architecture, making it suitable for deployment on resource-constrained and edge devices.

2.3.7 Practical Interpretation

FlexIoU's flexibility allows the loss function to be tailored to specific detection scenarios. Emphasizing the center penalty improves convergence stability for small objects, while increasing width or height penalties enhances shape accuracy for elongated targets. This tunable design enables FlexIoU to adapt across both generic benchmarks and domain-specific datasets without architectural modifications. To facilitate practical deployment and reproducibility, the proposed FlexIoU loss defined in Section 2.3 is summarized in Algorithm 1. The algorithm explicitly follows the mathematical formulation presented in Eqs. (7) - (11), detailing the computation of overlap, spatial alignment, and shape discrepancy terms used during bounding box regression. This algorithm is applied during the training phase of YOLO-based detectors to compute the localization loss for each matched prediction-ground-truth pair.

Algorithm 1. FlexIoU Bounding Box Regression

Input: predicted box $B_p = (x_p, y_p, w_p, h_p)$, ground-truth box $B_g = (x_g, y_g, w_g, h_g)$, weights $\lambda_c, \lambda_a, \lambda_w, \lambda_h$
 Output: FlexIoU similarity score and bounding-box regression loss ($FlexIoU$ and $\mathcal{L}_{box}$)

```

function FlexIoU( $B_p, B_g, \lambda_c, \lambda_a, \lambda_w, \lambda_h$ )
  # Step 1: Convert (center, w, h) → box corners for overlap computation
   $x_{p1} \leftarrow x_p - w_p/2; y_{p1} \leftarrow y_p - h_p/2;$ 
   $x_{p2} \leftarrow x_p + w_p/2; y_{p2} \leftarrow y_p + h_p/2;$ 
   $x_{g1} \leftarrow x_g - w_g/2; y_{g1} \leftarrow y_g - h_g/2;$ 
   $x_{g2} \leftarrow x_g + w_g/2; y_{g2} \leftarrow y_g + h_g/2;$ 

  # Step 2: Compute intersection area
   $x_{i1} \leftarrow \max(x_{p1}, x_{g1}); y_{i1} \leftarrow \max(y_{p1}, y_{g1});$ 
   $x_{i2} \leftarrow \min(x_{p2}, x_{g2}); y_{i2} \leftarrow \min(y_{p2}, y_{g2});$ 
   $i_w \leftarrow \max(0, x_{i2} - x_{i1})$ 
   $i_h \leftarrow \max(0, y_{i2} - y_{i1})$ 
   $A_{int} \leftarrow i_w * i_h$ 

  # Step 3: Compute union area and IoU
   $A_p \leftarrow w_p * h_p$ 
   $A_g \leftarrow w_g * h_g$ 
   $A_{union} \leftarrow A_p + A_g - A_{int} + \varepsilon$ 
   $IoU \leftarrow A_{int}/A_{union}$ 

  # Step 4: Smallest enclosing box (for normalization terms)
   $x_{c1} \leftarrow \min(x_{p1}, x_{g1}); y_{c1} \leftarrow \min(y_{p1}, y_{g1});$ 
   $x_{c2} \leftarrow \max(x_{p2}, x_{g2}); y_{c2} \leftarrow \max(y_{p2}, y_{g2});$ 
   $w_c \leftarrow (x_{c2} - x_{c1}) + \varepsilon$ 
   $h_c \leftarrow (y_{c2} - y_{c1}) + \varepsilon$ 
   $d^2 \leftarrow w_c^2 + h_c^2 + \varepsilon$ 

  # Step 5: Center distance penalty
   $d_x \leftarrow x_p - x_g$ 
   $d_y \leftarrow y_p - y_g$ 
   $\mathcal{L}_{center} \leftarrow (d_x^2 + d_y^2)/d^2$ 

  # Step 6: Aspect ratio penalty
   $v \leftarrow 4/\pi^2 \times (\tan^{-1}(w_g/h_g) - \tan^{-1}(w_p/h_p))^2$ 
   $\alpha \leftarrow v/(v - IoU + (1 + \varepsilon))$ 
   $\mathcal{L}_{aspect} \leftarrow \alpha \times v$ 

  # Step 7: Width/height penalties
   $\mathcal{L}_{width} \leftarrow (w_p - w_g)^2/w_c^2$ 
   $\mathcal{L}_{height} \leftarrow (h_p - h_g)^2/h_c^2$ 

  # Step 7: FlexIoU similarity
   $FlexIoU \leftarrow IoU - (\lambda_c \times \mathcal{L}_{center} + \lambda_a \times \mathcal{L}_{aspect} + \lambda_w \times \mathcal{L}_{width} + \lambda_h \times \mathcal{L}_{height})$ 

  # Step 8: Final FlexIoU loss
   $\mathcal{L}_{box} \leftarrow 1 - FlexIoU$ 

  return  $FlexIoU, \mathcal{L}_{box}$ 
end function

```

* ε is a small constant (e.g., 10^{-7} to avoid division by zero)

Algorithm 1 is invoked for each positive–ground-truth pair during training and used exclusively to compute the bounding-box regression loss within the YOLO detection head. The resulting FlexIoU loss is then combined with classification and DFL as described in Eq. (12) and optimized via backpropagation.

2.4 Experimental Setup

This section describes the datasets, data preparation, training configuration, and implementation details used to evaluate the proposed FlexIoU loss under controlled and reproducible conditions. All experiments were designed to ensure a fair comparison between baseline IoU-based losses and the proposed method by keeping the detection architecture and training settings identical across all configurations.

2.4.1 Datasets

Experiments were conducted on the Tomato Leaf Disease dataset, which is publicly available on Kaggle under the CC0 (Public Domain) license at: <https://www.kaggle.com/datasets/farukalam/tomato-leaf-diseases-detection-computer-vision>. This dataset represents a practical agricultural object detection scenario, consisting of images of tomato leaves affected by multiple disease types. The diseased regions exhibit irregular lesion shapes, varying object scales, and fine-grained visual patterns, which pose significant challenges for accurate bounding-box localization. Such characteristics make the dataset particularly suitable for evaluating the effectiveness of IoU-based regression losses under complex geometric variations. In particular, small lesion regions, non-uniform shapes, and ambiguous boundaries highlight the limitations of fixed geometric penalties and motivate the need for more flexible loss formulations. The dataset contains a total of 1,082 images, which were divided into 938 training images, 96 validation images, and 48 test images using a consistent split strategy to ensure unbiased evaluation. The test set was completely excluded from both training and validation and used exclusively for final performance assessment. This separation allows the reported results to reflect true generalization performance rather than memorization of training data. Detailed dataset statistics, including the number of images per class and their distribution across training, validation, and test splits, are summarized in Table 1.

Table 1. Dataset statistics of the tomato leaf disease dataset

Class	Train	Validation	Test	Total
Bacterial Spot	45	3	2	50
Black Spot	191	21	9	221
Early Blight	321	31	15	367
Healthy	162	19	7	188
Late Blight	144	13	9	166
Leaf Mold	45	4	4	53
Target Spot	30	5	2	37
Total	938	96	48	1,082

It should be noted that the dataset exhibits class imbalance across disease categories. For example, the Early Blight class contains substantially more training images than the Target Spot class. No class weighting, oversampling, or resampling techniques were applied in this study, as the objective was to evaluate the effect of the proposed regression loss within the standard YOLO training pipeline.

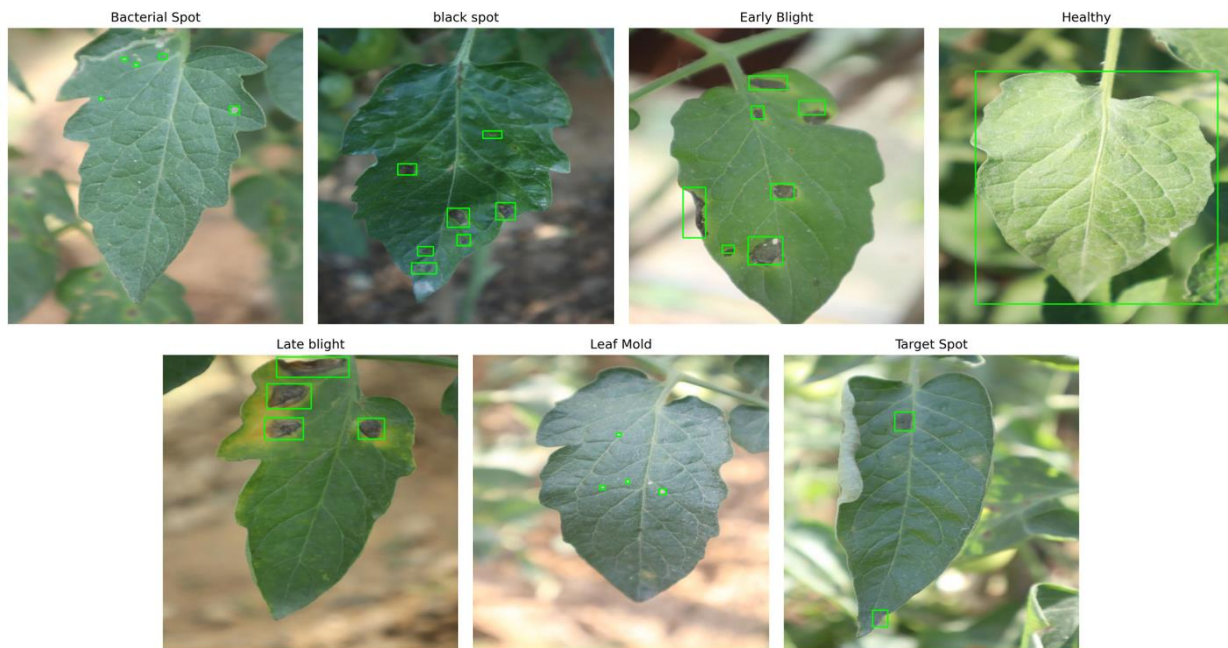


Figure 2. Samples of the tomato leaf disease dataset with ground-truth annotations.

To provide a qualitative understanding of the dataset, Figure 2 presents representative training samples with ground-truth annotations. Each image includes a green bounding box indicating the annotated disease region. The examples illustrate substantial variability in lesion size, shape, and visual appearance across different disease categories, further emphasizing the challenges of bounding-box regression in agricultural disease detection.

2.4.2 Data Preprocessing and Annotation

All images were resized to 640×640 pixels before training and evaluation, following the default YOLOv11 preprocessing pipeline. Ground-truth annotations were converted to the standard YOLO bounding-box format, representing each object as normalized center coordinates, width, and height. No dataset-specific geometric augmentation was introduced to avoid biasing the regression behavior toward a particular loss formulation.

2.4.3 Training Configuration

All models were trained using the same YOLOv11 detector architecture, with the bounding-box regression loss being the only interchangeable component. The classification loss and Distribution Focal Loss (DFL) formulations remained unchanged throughout the experiments. Training was performed with identical hyperparameters across all experiments, including the learning rate schedule, batch size, optimizer settings, image resolution, and number of epochs, as shown in Table 2.

Table 2. Training hyperparameters used in all experiments

Hyperparameter	Value
Input Resolution	640×640
Epochs	100
Batch Size	16
Optimizer	MuSGD
Initial Learning Rate	0.01
Momentum	0.90
Random Seeds	0, 1, 2
Device	NVIDIA L4

To investigate the tunability of the proposed FlexIoU loss, a sensitivity analysis was conducted by varying the geometric penalty coefficients λ_c , λ_a , λ_w , and λ_h . Eight coefficient combinations were evaluated, as summarized in Table 3. In addition, different loss-weighting settings for λ_{cls} and λ_{box} were investigated to examine the influence of balancing classification and localization objectives during training, as shown in Table 4. To reduce stochastic variance and improve statistical reliability, each experiment was repeated with three fixed random seeds, yielding three independently trained models per configuration. Results were reported as the mean performance across the three runs. The final FlexIoU configuration was selected based on validation performance and subsequently evaluated on the independent test set.

Table 3. FlexIoU coefficient configurations evaluated in the sensitivity analysis

Weight ID	λ_c	λ_a	λ_w	λ_h
W1	1.0	0.0	1.0	1.0
W2	1.0	0.5	1.0	1.0
W3	1.0	0.5	1.0	1.5
W4	1.0	0.5	1.5	1.0
W5	1.0	0.5	1.5	1.5
W6	1.0	1.0	1.0	1.0
W7	1.5	0.5	1.0	1.0
W8	1.5	0.5	1.5	1.5

Table 4. Loss-weighting configurations evaluated for FlexIoU

Loss ID	λ_{cls}	λ_{box}
C1	1.0	1.0
C2	1.0	1.5
C3	1.0	2.0
C4	1.2	1.5
C5	1.2	2.0

2.4.4 Implementation Details

All experiments were implemented using the Ultralytics YOLO framework. Training and evaluation were conducted on Google Colab with an Nvidia L4 GPU-enabled system to ensure consistent computational performance. During inference, identical post-processing steps, including non-maximum suppression, were applied to all models to ensure a fair comparison of both detection accuracy and inference efficiency. This experimental setup ensures that any observed performance differences among YOLOv11-IoU, CIoU, EIoU, and the proposed FlexIoU losses can be attributed primarily to the loss formulation itself rather than to architectural or training discrepancies.

2.5 Evaluation Metrics

To comprehensively assess the effectiveness of the proposed FlexIoU loss, detection performance was evaluated using standard metrics widely adopted in object detection research. These metrics capture both localization accuracy and computational efficiency, enabling a balanced comparison between baseline IoU-based losses and the proposed method. Detection accuracy was primarily measured using mean Average Precision (mAP), which evaluates the quality of predicted bounding boxes and classification confidence across different Intersection over Union thresholds. Specifically, mAP@0.5:0.95 was adopted, where Average Precision is computed at IoU thresholds from 0.5 to 0.95 with a step size of 0.05, then averaged. This metric provides a stringent evaluation of localization performance, as higher IoU thresholds require more precise alignment between predicted and ground-truth boxes. In addition to mAP, Precision and Recall were reported to analyze detection behavior further. Precision reflects the proportion of correctly predicted objects among all detections, while Recall measures the proportion of ground-truth objects that are successfully detected. Together, these metrics provide insight into the trade-off between false positives and false negatives and help interpret the impact of different regression losses on detection confidence. Beyond accuracy, inference efficiency is a critical consideration, particularly for real-time and edge-device applications. Inference

performance was evaluated by measuring the average inference time per image during the testing phase under identical hardware conditions. All models were evaluated using the same input resolution and post-processing configuration to ensure fair comparison. By jointly considering detection accuracy and inference efficiency, this evaluation protocol enables a comprehensive assessment of how different IoU-based loss functions affect both localization quality and practical deployment performance.

3. Results and Discussion

This section presents the experimental results of evaluating the proposed FlexIoU loss within a YOLOv11 detection framework on the tomato leaf disease dataset. The objective of this analysis is to examine whether introducing tunable geometric penalties can improve bounding-box regression performance within a controlled training pipeline while maintaining practical efficiency. To ensure a fair comparison, all models were trained using identical settings, and only the bounding-box regression loss was varied across four configurations: the default YOLOv11 regression (hereafter referred to as YOLOv11-IoU), CIoU, EIoU, and the proposed FlexIoU. All experiments were repeated across three random seeds, and performance is reported as mean values with corresponding standard deviations to reflect both effectiveness and stability

3.1 Hyperparameter Sensitivity Analysis of FlexIoU

Since FlexIoU introduces four tunable geometric penalty coefficients, an additional sensitivity analysis was conducted to identify promising coefficient configurations before performing the final evaluation. Eight coefficient combinations were evaluated under identical training conditions using $\lambda_{cls} = 1.0$ and $\lambda_{box} = 1.0$. To avoid selecting configurations solely based on localization accuracy, a weighted ranking score was used to consider localization quality, precision, and recall jointly. Specifically, a weight of 0.5 was assigned to mAP@0.5:0.95, while precision and recall were assigned weights of 0.2 and 0.3, respectively. The larger weight assigned to mAP@0.5:0.95 reflects its role as the primary evaluation metric for object detection, as it summarizes localization performance across multiple IoU thresholds. Precision was included to penalize excessive false-positive detections, whereas recall was given slightly higher importance to ensure that diseased regions were not overlooked. This weighting strategy was designed to favor configurations that achieve a balanced trade-off between localization accuracy and detection reliability rather than optimizing a single metric. The corresponding results are summarized in Table 5. Configurations ranked among the top five by weighted score are highlighted in bold and were subsequently selected for further investigation under different λ_{cls} and λ_{box} settings.

Table 5. Sensitivity analysis of FlexIoU geometric penalty coefficients using $\lambda_{cls} = 1.0$ and $\lambda_{box} = 1.0$

Weight	mAP@0.5:0.95	Precision	Recall	F1-score
W1	0.504 ± 0.005	0.801 ± 0.033	0.725 ± 0.003	0.761 ± 0.016
W2	0.491 ± 0.006	0.760 ± 0.023	0.732 ± 0.036	0.746 ± 0.030
W3	0.491 ± 0.014	0.775 ± 0.071	0.743 ± 0.036	0.756 ± 0.015
W4	0.494 ± 0.001	0.719 ± 0.048	0.730 ± 0.043	0.723 ± 0.023
W5	0.493 ± 0.008	0.754 ± 0.051	0.781 ± 0.033	0.766 ± 0.022
W6	0.495 ± 0.006	0.767 ± 0.029	0.738 ± 0.031	0.751 ± 0.004
W7	0.492 ± 0.017	0.768 ± 0.016	0.768 ± 0.017	0.768 ± 0.007
W8	0.498 ± 0.001	0.784 ± 0.051	0.728 ± 0.046	0.753 ± 0.008

As shown in Table 5, different coefficient combinations led to noticeable variations in detection performance, confirming that FlexIoU's behavior is sensitive to the choice of geometric penalties. Although several configurations achieved similar mAP@0.5:0.95 values, differences in precision and recall led to different weighted scores. This observation indicates that evaluating coefficient configurations using a single metric may overlook important trade-offs between localization accuracy and detection reliability. Therefore, the weighted ranking strategy was adopted to identify configurations that provide a more balanced overall performance. Interestingly, configurations with different λ_a values produced different performance trends, suggesting that the contribution of the aspect-ratio penalty is dataset-dependent and may not always be

beneficial when applied with a fixed weight. This observation further supports the motivation behind FlexIoU, where the aspect-ratio penalty can be flexibly adjusted or completely disabled depending on object characteristics and localization requirements. The top-ranked configurations were subsequently evaluated under additional λ_{cls} and λ_{box} settings to investigate the interaction between geometric penalties and loss-weight balancing. The results of this second-stage sensitivity analysis are presented in Table 6.

Table 6. Performance of selected FlexIoU configurations under different λ_{cls} and λ_{box} settings.

Weight	λ_{cls}	λ_{box}	mAP@0.5:0.95	Precision	Recall	F1-score
W1	1.0	1.0	0.504 ± 0.005	0.801 ± 0.033	0.725 ± 0.003	0.761 ± 0.016
	1.0	1.5	0.504 ± 0.011	0.782 ± 0.042	0.746 ± 0.032	0.763 ± 0.007
	1.0	2.0	0.491 ± 0.003	0.774 ± 0.015	0.747 ± 0.009	0.760 ± 0.010
	1.2	1.5	0.492 ± 0.005	0.760 ± 0.040	0.739 ± 0.047	0.748 ± 0.013
	1.2	2.0	0.496 ± 0.016	0.753 ± 0.046	0.737 ± 0.015	0.744 ± 0.022
W3	1.0	1.0	0.491 ± 0.014	0.775 ± 0.071	0.743 ± 0.035	0.756 ± 0.015
	1.0	1.5	0.494 ± 0.011	0.765 ± 0.014	0.737 ± 0.026	0.751 ± 0.018
	1.0	2.0	0.482 ± 0.007	0.748 ± 0.042	0.739 ± 0.030	0.742 ± 0.005
	1.2	1.5	0.484 ± 0.004	0.738 ± 0.038	0.734 ± 0.009	0.736 ± 0.022
	1.2	2.0	0.510 ± 0.010	0.784 ± 0.058	0.752 ± 0.046	0.765 ± 0.006
W5	1.0	1.0	0.493 ± 0.008	0.754 ± 0.051	0.781 ± 0.033	0.766 ± 0.022
	1.0	1.5	0.499 ± 0.016	0.783 ± 0.018	0.732 ± 0.041	0.756 ± 0.014
	1.0	2.0	0.488 ± 0.009	0.750 ± 0.044	0.735 ± 0.015	0.742 ± 0.028
	1.2	1.5	0.495 ± 0.005	0.798 ± 0.010	0.721 ± 0.049	0.757 ± 0.023
	1.2	2.0	0.498 ± 0.008	0.721 ± 0.034	0.761 ± 0.020	0.740 ± 0.014
W7	1.0	1.0	0.492 ± 0.017	0.768 ± 0.016	0.768 ± 0.017	0.768 ± 0.007
	1.0	1.5	0.498 ± 0.017	0.751 ± 0.018	0.755 ± 0.007	0.753 ± 0.013
	1.0	2.0	0.487 ± 0.008	0.734 ± 0.015	0.735 ± 0.024	0.734 ± 0.005
	1.2	1.5	0.511 ± 0.012	0.786 ± 0.015	0.724 ± 0.035	0.754 ± 0.025
	1.2	2.0	0.503 ± 0.002	0.799 ± 0.012	0.742 ± 0.021	0.770 ± 0.007
W8	1.0	1.0	0.498 ± 0.001	0.784 ± 0.051	0.728 ± 0.046	0.753 ± 0.008
	1.0	1.5	0.489 ± 0.005	0.743 ± 0.035	0.715 ± 0.000	0.728 ± 0.016
	1.0	2.0	0.492 ± 0.005	0.756 ± 0.022	0.737 ± 0.020	0.746 ± 0.017
	1.2	1.5	0.503 ± 0.006	0.781 ± 0.018	0.755 ± 0.017	0.768 ± 0.017
	1.2	2.0	0.498 ± 0.002	0.769 ± 0.022	0.739 ± 0.008	0.754 ± 0.014

The results indicate that the optimal performance of FlexIoU depends not only on the geometric penalty coefficients but also on the balance between localization and classification objectives. Increasing λ_{box} generally places greater emphasis on bounding-box regression, whereas increasing λ_{cls} encourages improved classification performance. However, the influence of these parameters varies across coefficient configurations, suggesting that the best-performing setting cannot be determined solely from a single evaluation metric. To provide a more balanced assessment, the final configuration was selected using the weighted ranking score defined in Section 3.1, which combines mAP@0.5:0.95, precision, and recall with weights of 0.5, 0.2, and 0.3, respectively. Based on this criterion, configuration W3 ($\lambda_c = 1.0$, $\lambda_a = 0.5$, $\lambda_w = 1.0$, $\lambda_h = 1.5$, $\lambda_{cls} = 1.2$, and $\lambda_{box} = 2.0$) achieved the highest overall score and was therefore selected for final evaluation on the independent test set.

3.2 Quantitative Results on the Tomato Leaf Disease Dataset

Table 7 reports the quantitative performance of the evaluated bounding-box regression losses on the tomato leaf disease test set. Overall, the proposed FlexIoU achieves the highest mean mAP@0.5:0.95 among all compared methods, indicating improved localization accuracy across multiple IoU thresholds. This improvement is particularly relevant for tomato disease detection, where disease symptoms often appear as small, irregular, and weakly bounded regions that are challenging to localize accurately.

Table 7. Quantitative performance comparison on the tomato leaf disease test set

Loss	mAP@0.5:0.95	Precision	Recall	F1-score	Inference Time (ms)	FPS
YOLOv11-IoU	0.473 ± 0.029	0.690 ± 0.044	0.739 ± 0.064	0.711 ± 0.018	14.117 ± 0.133	70.842 ± 0.669
CIoU	0.464 ± 0.006	0.695 ± 0.082	0.742 ± 0.021	0.715 ± 0.035	14.147 ± 0.112	70.691 ± 0.563
EIoU	0.464 ± 0.015	0.632 ± 0.066	0.755 ± 0.029	0.687 ± 0.038	14.211 ± 0.102	70.372 ± 0.507
FlexIoU	0.488 ± 0.005	0.751 ± 0.012	0.768 ± 0.005	0.759 ± 0.003	14.116 ± 0.083	70.844 ± 0.419

Compared with the default YOLOv11 regression loss, FlexIoU improves mAP@0.5:0.95, precision, recall, and F1-score, indicating that introducing tunable geometric penalties can enhance localization performance. Among all evaluated methods, FlexIoU achieves the highest mean values across all reported detection metrics, suggesting a favorable balance between localization accuracy and classification reliability. While CIoU and EIoU provide alternative geometric formulations for bounding-box regression, their overall mAP@0.5:0.95 remains lower than both YOLOv11-IoU and FlexIoU on this dataset. As shown in Table 7, EIoU achieves the highest recall among the baseline methods, indicating a tendency to detect a larger proportion of disease regions. However, this behavior is accompanied by reduced precision, resulting in a lower overall F1-score. In contrast, FlexIoU simultaneously achieves the highest precision, recall, and F1-score, indicating a more balanced trade-off between false-positive and false-negative detections. Such a balance is particularly important in agricultural disease detection, where missed lesions may delay treatment decisions, while excessive false detections may lead to unnecessary interventions. Importantly, the performance improvements achieved by FlexIoU are not limited to a single training run. The relatively low standard deviations reported in Table 7 indicate stable performance across three independent random seeds. This observation suggests that the proposed loss function produces consistent detection behavior under different initialization conditions and does not rely on a specific random seed to achieve improved performance. Table 8 presents the statistical significance analysis of mAP@0.5:0.95 for FlexIoU relative to the baseline regression losses. FlexIoU achieved higher mean mAP@0.5:0.95 values than all evaluated baselines, with mean improvements of +0.0152, +0.0244, and +0.0248 over YOLOv11-IoU, CIoU, and EIoU, respectively.

Table 8. Statistical significance analysis of mAP@0.5:0.95 between FlexIoU and baseline losses across three random seeds

Comparison	FlexIoU mean	Baseline mean	Mean difference	Paired t-test p-value	Wilcoxon p-value
FlexIoU vs YOLOv11-IoU	0.4884	0.4732	+0.0152	0.5135	1.0000
FlexIoU vs CIoU	0.4884	0.4640	+0.0244	0.0218	0.2500
FlexIoU vs EIoU	0.4884	0.4636	+0.0248	0.0980	0.2500

The paired t-test revealed a statistically significant improvement of FlexIoU over CIoU ($p = 0.0218$). In contrast, the comparisons against YOLOv11-IoU ($p = 0.5135$) and EIoU ($p = 0.0980$) did not reach statistical significance under the current experimental setting. Similarly, the Wilcoxon signed-rank test did not indicate significant differences between FlexIoU and the baseline methods. These results should be interpreted in the context of the limited number of repeated runs used in this study. Although statistical significance was observed only for the comparison with CIoU, FlexIoU consistently achieved the highest mean values for mAP@0.5:0.95, precision, recall, and F1-score among all evaluated methods. Therefore, the overall experimental evidence suggests that FlexIoU provides a competitive and reliable improvement in localization performance on the tomato leaf disease dataset, while additional experiments with more repeated runs and a wider range of datasets would be beneficial for further validation. To further analyze the quantitative results, Figure 3 presents the convergence behavior of each regression loss across three independent random seeds in terms of validation mAP@0.5:0.95 over training epochs. By visualizing all three runs for each method, the figure provides a clearer view of training stability and reproducibility under different initialization conditions. Overall, the convergence trajectories exhibit similar trends across seeds, indicating that a single favorable run does not drive the reported performance.

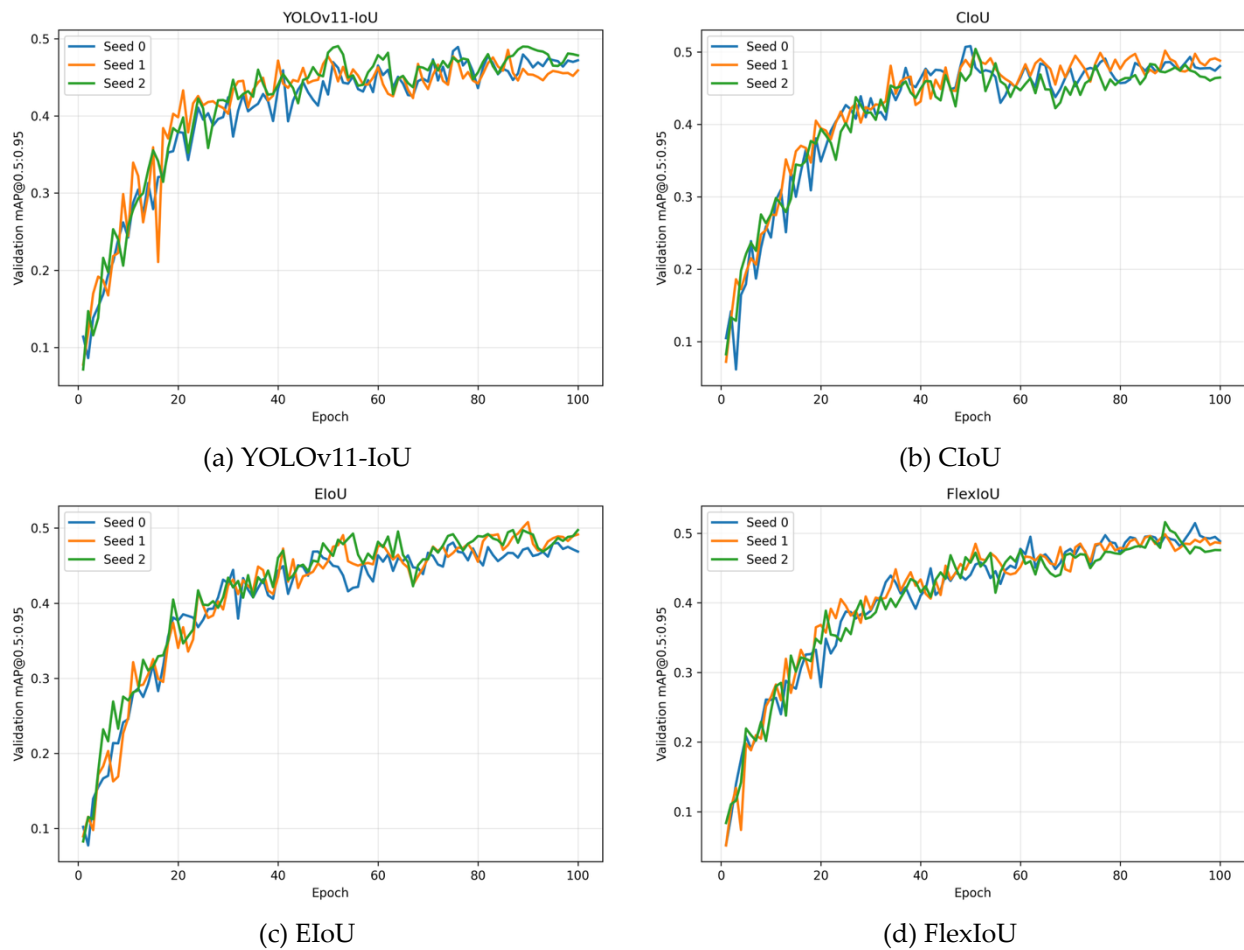


Figure 3. Convergence behavior of YOLOv11-IoU, CIoU, EIoU, and FlexIoU across three independent random seeds in terms of validation mAP@0.5:0.95.

Although minor variations can be observed among individual runs, all evaluated regression losses converge to stable performance levels during later training epochs. FlexIoU maintains competitive validation performance throughout training and demonstrates consistent convergence behavior across different random seeds. This observation is consistent with the statistical results reported in Table 8, where FlexIoU achieves the highest mean mAP@0.5:0.95 among the evaluated methods while maintaining relatively low variation across repeated runs. These findings suggest that the proposed tunable geometric penalties do not introduce unstable optimization dynamics and can maintain reliable training behavior under different initialization conditions. The convergence analysis therefore, provides additional evidence that the performance improvements achieved by FlexIoU are not dependent on a particular random seed and are consistently observed across repeated experiments.

3.3 Comparison with Baseline Loss Functions

To further analyze the behavior of the proposed FlexIoU loss beyond aggregate performance metrics, this section compares the precision-recall curves and confusion matrices of YOLOv11-IoU, CIoU, EIoU, and FlexIoU on the tomato leaf disease test set. While Table 7 summarizes the overall quantitative performance, Figures 4 and 5 provide additional insight into class-wise prediction behavior, precision-recall characteristics, and error distribution across disease categories. Figure 4 presents the precision-recall curves of all evaluated regression losses. Overall, the four methods exhibit similar precision-recall curves, indicating that the underlying detector architecture remains effective across different regression loss formulations. Although the differences among the curves are relatively moderate, variations can still be observed for individual disease categories. In particular, classes such as Early Blight, Late Blight, and Target Spot maintain strong precision

across a wide range of recall values, whereas classes with fewer samples, such as Bacterial Spot and Leaf Mold, exhibit greater performance fluctuations. These observations are consistent with the class imbalance characteristics discussed in Section 2.4.

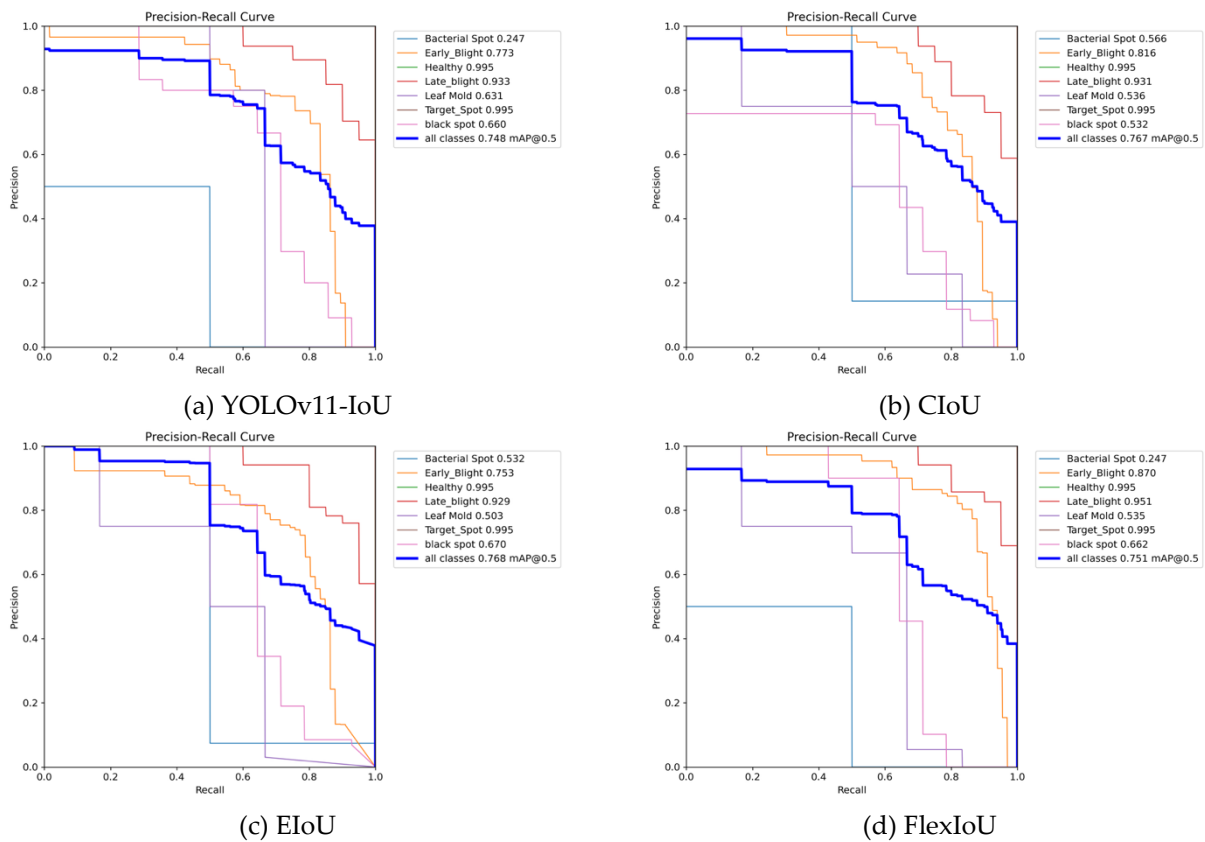


Figure 4. Precision–recall curves of YOLOv11-IoU, CIoU, EIoU, and FlexIoU on the tomato test set.

Figure 5 presents the confusion matrices of the evaluated methods. Across all regression losses, most predictions are concentrated along the diagonal, indicating that the detector correctly identifies the majority of disease instances. The remaining errors are primarily due to visually similar disease categories and background confusion, as expected in tomato leaf disease detection given overlapping lesion textures, color patterns, and disease morphology. Compared with the baseline losses, FlexIoU exhibits prediction patterns that are generally consistent with the quantitative results reported in Table 7. The confusion matrix shows strong diagonal dominance across major disease categories, while maintaining relatively limited confusion with background regions. These observations suggest that the proposed tunable geometric penalties improve localization quality without introducing excessive false-positive detections. When considered together with the quantitative results, the qualitative analyses help explain the performance differences observed among the evaluated losses. CIoU incorporates center-distance and aspect-ratio penalties, whereas EIoU further introduces explicit width and height penalties. FlexIoU extends this concept by allowing independent control of multiple geometric penalty terms through tunable coefficients. This flexibility enables the regression objective to be adapted to dataset-specific characteristics and may contribute to the improved localization performance observed in the tomato leaf disease dataset. Overall, the qualitative evidence provided by Figures 4 and 5 complements the quantitative results reported in Section 3.2. While the differences among the evaluated losses are not dramatic at the visual level, the combined quantitative and qualitative analyses suggest that FlexIoU provides a favorable balance between localization accuracy, detection reliability, and prediction consistency.

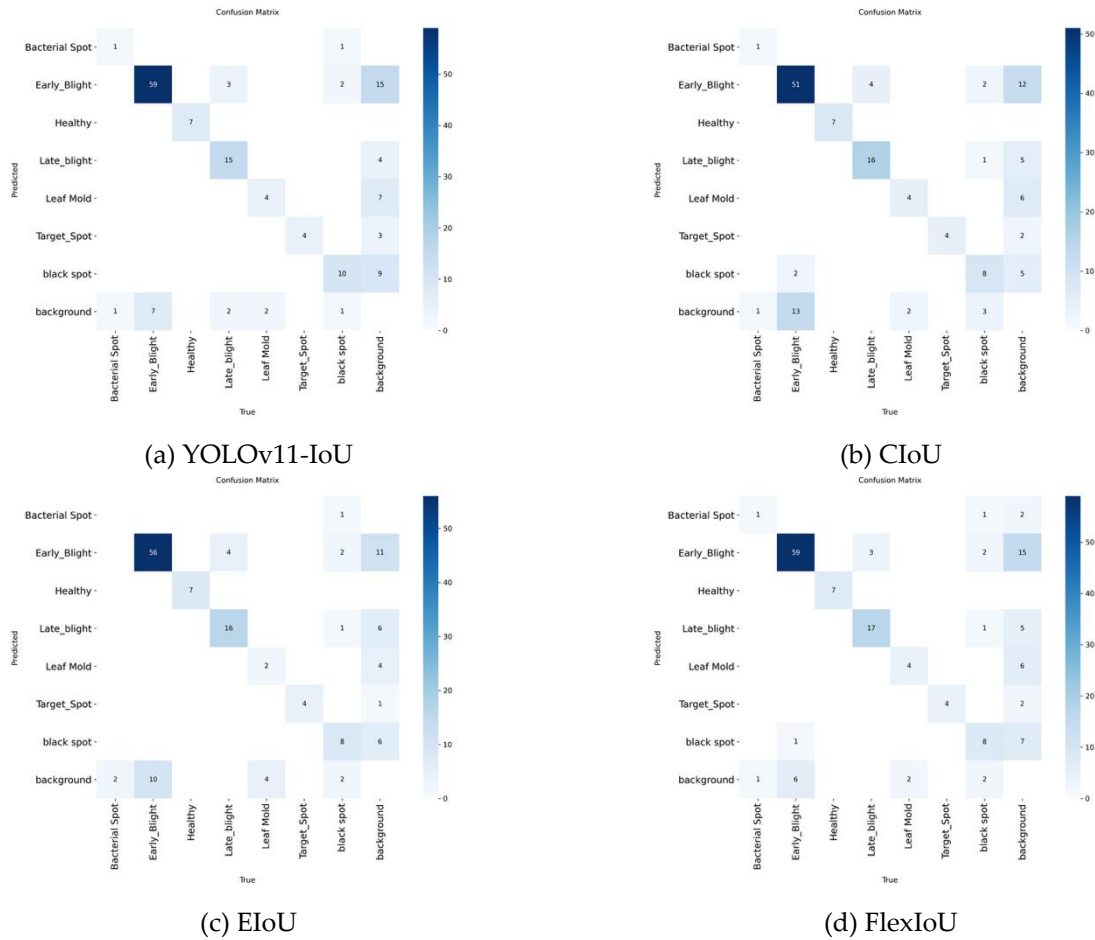


Figure 5. Confusion matrices of YOLOv11-IoU, CIoU, EIoU, and FlexIoU on the tomato test set

3.4 Additional Evaluation on an Independent Tomato Disease Dataset

To further assess the robustness of the proposed FlexIoU loss, an additional evaluation was conducted on an independent tomato leaf disease dataset obtained from a different public source. Unlike the primary dataset used in Sections 3.2 and 3.3, this dataset exhibits distinct class distributions, lesion characteristics, and annotation properties, providing a complementary evaluation setting to assess whether FlexIoU's effectiveness can be maintained under varying data conditions. The dataset was obtained from a publicly available Kaggle repository from <https://www.kaggle.com/datasets/vasantharank/tomato-leaf-disease-detection-yolov8-dataset> and is distributed under the CC BY-NC-SA 4.0 license. To ensure a balanced and reproducible evaluation protocol, the original dataset was re-partitioned using a stratified class-wise split of 70%, 20%, and 10% for training, validation, and testing, respectively. The resulting dataset contains 10 tomato disease categories: Tomato Bacterial Spot, Early Blight, Late Blight, Leaf Mold, Septoria Leaf Spot, Spider Mites, Target Spot, Yellow Leaf Curl Virus, Healthy, and Mosaic Virus. After re-splitting, the dataset consisted of 7,476 training images, 2,131 validation images, and 1,080 testing images. The purpose of this experiment was not to perform an additional hyperparameter search, but rather to evaluate whether the coefficient configuration selected in Section 3.1 remains effective when applied to a different tomato disease dataset. Therefore, the best-performing FlexIoU configuration identified from the primary dataset was directly transferred to this independent dataset without further coefficient optimization. All baseline methods were trained using identical experimental settings to ensure a fair comparison. Based on the sensitivity analysis presented in Section 3.2, the selected FlexIoU configuration employed $\lambda_c = 1.0$, $\lambda_a = 0.5$, $\lambda_w = 1.0$, $\lambda_h = 1.5$, $\lambda_{cls} = 1.2$, and $\lambda_{box} = 2.0$. This configuration was transferred directly to the independent dataset without further tuning to evaluate its robustness across different data distributions. Table 9 presents the quantitative

performance comparison between YOLOv11-IoU, CIoU, EIoU, and the proposed FlexIoU on the independent tomato disease dataset.

Table 9. Quantitative performance comparison on the independent tomato disease dataset

Loss	mAP@0.5:0.95	Precision	Recall	F1-score	Inference Time (ms)	FPS
YOLOv11-IoU	0.883 ± 0.005	0.970 ± 0.014	0.954 ± 0.004	0.962 ± 0.006	14.715 ± 0.131	67.962 ± 0.608
CIoU	0.873 ± 0.003	0.954 ± 0.008	0.956 ± 0.002	0.955 ± 0.003	15.022 ± 0.251	66.579 ± 1.104
EIoU	0.843 ± 0.013	0.906 ± 0.023	0.960 ± 0.013	0.932 ± 0.009	14.955 ± 0.146	66.872 ± 0.653
FlexIoU	0.858 ± 0.007	0.944 ± 0.025	0.928 ± 0.012	0.936 ± 0.017	14.895 ± 0.140	67.139 ± 0.628

As shown in Table 9, the performance trends observed on the independent dataset differ from those reported on the primary tomato leaf disease dataset. While FlexIoU achieved the best overall performance on the primary dataset, YOLOv11-IoU obtained the highest mAP@0.5:0.95, precision, and F1-score on the independent dataset. Nevertheless, FlexIoU remained competitive and achieved performance comparable to the evaluated IoU-based regression losses, indicating that the proposed formulation can maintain reasonable detection effectiveness under different dataset characteristics. It is important to note that the independent dataset differs substantially from the primary dataset in terms of class distribution, lesion composition, and annotation characteristics. In particular, the rebalanced split provides a more uniform distribution across disease categories and generally contains larger lesion regions than those observed in the primary dataset. Under these conditions, the advantage of emphasizing width and height penalties may become less pronounced, which could partially explain the reduced performance gap between FlexIoU and the baseline methods. These observations suggest that the effectiveness of individual geometric penalty terms depends on dataset-specific object characteristics. Despite not achieving the highest performance on this dataset, FlexIoU continued to produce competitive results while maintaining inference efficiency comparable to the baseline losses. This behavior is consistent with the findings of the sensitivity analysis in Section 3.2, which demonstrated that the contribution of geometric penalty components depends on the underlying data distribution. Therefore, the additional evaluation provides useful evidence that the proposed weighting framework remains applicable across different tomato disease datasets, although the optimal coefficient configuration may vary depending on lesion size, shape, and annotation properties. Overall, the results indicate that FlexIoU is not universally superior across all tomato disease datasets but remains a competitive regression loss under varying dataset conditions. These findings highlight both the potential and limitations of tunable geometric penalties, underscoring the importance of dataset-specific coefficient selection when applying FlexIoU across different detection scenarios.

3.5 Inference Speed and Computational Overhead

In addition to detection accuracy, computational efficiency is a key factor for real-world deployment, particularly in agricultural monitoring systems where real-time or near-real-time inference is often required. To evaluate whether the proposed FlexIoU loss incurs additional runtime cost, inference latency, and throughput, these metrics were measured under controlled benchmarking conditions. To reduce measurement variability caused by model initialization and GPU stabilization effects, all timing experiments were conducted after a dedicated warm-up phase before latency measurement. The resulting inference latency and throughput are summarized in Table 7. As shown in Table 7, FlexIoU achieves inference time and FPS comparable to those of CIoU and EIoU. Across all evaluated configurations, the average inference latency remains nearly identical, with only negligible variations. This result is expected, as IoU-based regression losses are applied exclusively during training, whereas inference relies solely on forward propagation through the trained detection network. Consequently, modifying the regression loss formulation does not alter the inference-time computational graph. Following the controlled warm-up procedure, all evaluated methods exhibited low variance in both latency and FPS measurements. The resulting consistency indicates that the reported inference performance is stable and largely independent of runtime initialization effects. Furthermore, the comparable latency observed across YOLOv11-IoU, CIoU, EIoU, and FlexIoU confirms that modifications to the regression loss do not introduce additional computational overhead during inference.

Overall, these results confirm that FlexIoU enhances localization robustness without introducing additional inference-time overhead. The absence of measurable computational cost makes FlexIoU suitable for deployment in resource-constrained environments and practical agricultural applications, where maintaining efficiency is as critical as improving detection accuracy.

3.6 Discussion

The experimental results demonstrate that improvements in bounding-box localization can be achieved without introducing additional architectural complexity or inference-time overhead. On the primary tomato leaf disease dataset, FlexIoU achieved the highest mAP@0.5:0.95, precision, recall, and F1-score among the evaluated methods while maintaining inference efficiency comparable to YOLOv11-IoU, CIoU, and EIoU. Furthermore, the relatively low variation observed across three independent random seeds suggests that the proposed loss provides stable optimization behavior under different initialization conditions. A key finding of this study is that the effectiveness of bounding-box regression depends on how geometric penalties are weighted. The sensitivity analysis showed that different combinations of λ_c , λ_a , λ_w , λ_h produced noticeably different detection outcomes, indicating that the contribution of individual geometric constraints is not uniform across all datasets. This observation supports the central motivation of FlexIoU: to provide a flexible weighting framework rather than relying on a fixed penalty structure. The findings are generally consistent with previous studies showing that incorporating additional geometric information beyond conventional IoU can improve localization performance. Similar to CIoU, EIoU, SIoU, and MPDIoU, the proposed method leverages geometric relationships between predicted and ground-truth boxes. However, unlike these approaches, FlexIoU emphasizes adaptability through independently tunable penalty terms, allowing the regression objective to be adjusted according to dataset characteristics and application requirements. Figure 6 presents representative detection results obtained using the FlexIoU-based detector. The qualitative results are consistent with the quantitative findings reported in Section 3.2, demonstrating that the model can successfully localize disease regions across multiple tomato leaf disease categories while maintaining visually coherent bounding-box predictions.

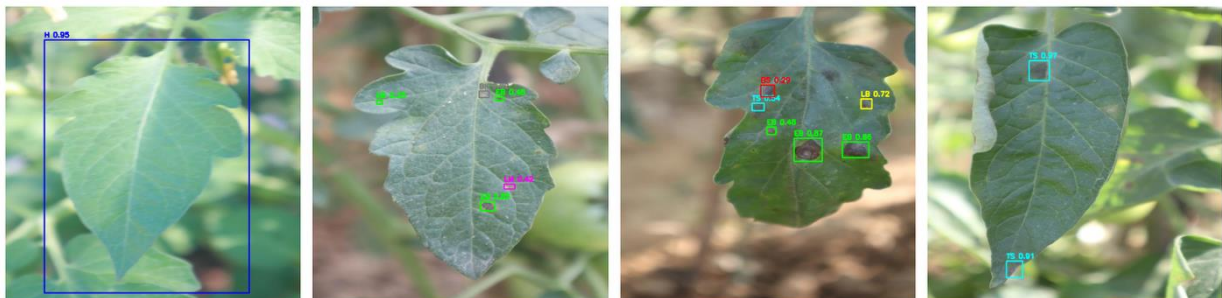


Figure 6. Detection results produced by the FlexIoU on the test dataset.

Another important observation is that the localization improvements achieved by FlexIoU do not come at the expense of increased computational cost. Since the proposed modifications affect only the training objective and do not alter the inference-time computation graph, latency and throughput remain comparable to those of the baseline losses. This characteristic is particularly important for agricultural monitoring systems and edge-device deployments where computational resources are limited. The additional evaluation on an independent tomato disease dataset further highlights the importance of dataset-specific coefficient selection. Although FlexIoU achieved the best performance on the primary dataset, YOLOv11-IoU achieved slightly higher mAP@0.5:0.95 on the secondary dataset. This result suggests that no single coefficient configuration is universally optimal and reinforces the need to introduce tunable geometric penalties that can be adapted to different data distributions. Finally, it should be noted that recent regression losses such as SIoU [23] and MPDIoU [24] were not included in the experimental comparison. Therefore, the reported improvements should be interpreted relative to the evaluated baselines (YOLOv11-IoU, CIoU, and EIoU) rather than all existing IoU-based regression losses. Additional comparisons with more recent loss formulations and evaluations across diverse datasets would provide a more comprehensive assessment of FlexIoU's strengths and limitations.

4. Conclusion

This study introduced FlexIoU, a flexible and lightweight IoU-based regression loss designed to improve bounding-box localization in YOLO-based object detection, with a particular focus on agricultural disease imagery. By incorporating tunable geometric penalties into the regression objective, FlexIoU enables independent control of center-distance, aspect-ratio, width, and height constraints, providing greater flexibility than conventional fixed-penalty IoU-based losses. Experimental results on the tomato leaf disease dataset demonstrated that FlexIoU achieved the highest mean mAP@0.5:0.95, precision, recall, and F1-score among the evaluated regression losses. Furthermore, the proposed loss maintained stable performance across three independent random seeds while preserving inference efficiency comparable to the baseline methods. These findings suggest that introducing tunable geometric penalties can improve localization performance without sacrificing computational efficiency. An additional contribution of this work is the introduction of a configurable regression framework that enables geometric penalty terms to be adjusted based on dataset characteristics. The sensitivity analysis showed that different coefficient configurations produce different performance behaviors, highlighting the importance of adaptive geometric weighting in bounding-box regression. This flexibility represents a practical advantage over fixed-penalty formulations such as CIoU and EIoU. Despite these promising results, several limitations should be acknowledged. First, the experimental evaluation was conducted on a tomato leaf disease dataset, which may limit the generalizability of the findings to other agricultural domains and object detection tasks. Second, the comparative experiments were restricted to YOLOv11-IoU, CIoU, and EIoU, while more recent IoU-based losses such as SIoU [23] and MPDIoU [24] were not included. Third, although statistical significance was observed when comparing FlexIoU with CIoU, the limited number of repeated runs may restrict the strength of statistical conclusions for some comparisons. Furthermore, the dataset exhibits noticeable class imbalance across disease categories. Although all methods were trained under identical conditions, class imbalance may have influenced category-specific detection performance and represents an additional limitation of the current study. Future work will focus on evaluating FlexIoU on additional agricultural and general-purpose object detection datasets, as well as extending comparisons to more recent IoU-based regression losses. Further research will also investigate adaptive and data-driven approaches for automatic coefficient selection. In addition, extending the proposed formulation to support rotated and oriented bounding-box regression represents a promising direction for improving localization performance in more complex detection scenarios.

5. Acknowledgments

The authors would like to thank all individuals who provided valuable support and feedback during the development of this study.

Author Contributions: Conceptualization, N.K.; methodology, N.K.; software, N.K.; validation, N.K., A.N., K.S. and N.P.; formal analysis, N.K.; investigation, N.K.; resources, N.K.; data curation, N.K.; writing — original draft preparation, N.K.; writing — review and editing, N.K., K.S. and N.P.; visualization, N.K., K.S.; supervision, A.N. and N.P.; project administration, N.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study, in the collection, analysis, or interpretation of data, in the writing of the manuscript, or in the decision to publish the results.

References

- [1] Nagababu, P.; Nageena, S.; Dharani, V.; Naveen, D. Plant Disease Detection and Diagnosis. In *Proceedings of the 2024 5th International Conference for Emerging Technology (INCET)*; Belgaum, India, 2024; pp 1–6. <https://doi.org/10.1109/INCET61516.2024.10593371>

- [2] Lokhande, N.; Thool, V.; Vikhe, P. Comparative Analysis of Different Plant Leaf Disease Classification and Detection Using CNN. In *Proceedings of the 2024 International Conference on Recent Innovation in Smart and Sustainable Technology (ICRISST)*; Bengaluru, India, 2024; pp 1–4. <https://doi.org/10.1109/ICRISST59181.2024.10921975>
- [3] Kusuma, S.; Jothi, K. R. Early Betel Leaf Disease Detection Using Vision Transformer and Deep Learning Algorithms. *Int. J. Inf. Technol.* **2024**, *16*(1), 169–180. <https://doi.org/10.1007/s41870-023-01647-3>
- [4] Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; Las Vegas, NV, USA, 2016; pp 779–788. <https://doi.org/10.1109/CVPR.2016.91>
- [5] Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y. M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
- [6] Jocher, G. *Ultralytics YOLOv5*, Version 7.0; Ultralytics, 2020. <https://github.com/ultralytics/yolov5> (accessed 2025-12-13). <https://doi.org/10.5281/zenodo.3908559>
- [7] Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y. M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE, 2023; pp 7464–7475. <https://doi.org/10.1109/CVPR52729.2023.00721>
- [8] Varghese, R.; Sambath, M. YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance and Robustness. In *Proceedings of the 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS)*; Chennai, India, 2024; pp 1–6. <https://doi.org/10.1109/ADICS58448.2024.10533619>
- [9] Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J. YOLOv10: Real-Time End-to-End Object Detection. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 107984–108011
- [10] Jocher, G.; Qiu, J. *Ultralytics YOLO11*, Version 11.0.0; Ultralytics, 2024. <https://github.com/ultralytics/ultralytics> (accessed 2025-12-13).
- [11] Ramesh, M.; Sherin, S. D.; Deepak A, V.; Dhanin, G.; Kodeeswari, K.; Nathiya, T. Tomato Plant Disease Classification with Pesticide Suggestion Using YOLOv8 Algorithm. In *Proceedings of the 2025 3rd International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA)*; Coimbatore, India, 2025; pp 1–6. <https://doi.org/10.1109/ICAECA63854.2025.11012147>
- [12] Wei, Y.; Pan, X. Improvement Algorithm Analysis and Implementation of Plant Disease and Pest Recognition Based on YOLO. In *Proceedings of the 5th International Conference on Computer Vision, Image and Deep Learning (CVIDL 2024)*; Zhuhai, China, 2024; pp 442–446. <https://doi.org/10.1109/CVIDL62147.2024.10603557>
- [13] Pathan, F.; Sakalle, V.; Munir, S. An Efficient Machine Learning Technique for Agricultural Plant Leaf Disease Prediction Using Image Processing. In *Proceedings of the 2025 International Conference on Engineering Innovations and Technologies (ICoEIT)*; Bhopal, India, 2025; pp 612–617. <https://doi.org/10.1109/ICoEIT63558.2025.11211659>
- [14] David, F.; Mukunthan, M. A. Betel Leaf Diseases Classification Using Machine Learning Algorithm: A Feasible Approach. *J. Adv. Res. Appl. Sci. Eng. Technol.* **2024**, *40*(1), 74–86. <https://doi.org/10.37934/araset.40.1.7486>
- [15] Mamart, S.; Sangamuang, S.; Sugunnasil, P. Deep Learning-Based Classification of Apple Leaf Diseases under Field Conditions. *ASEAN J. Sci. Technol. Rep.* **2026**, *29*(3), e261221. <https://doi.org/10.55164/ajstr.v29i3.261221>
- [16] Yu, J.; Jiang, Y.; Wang, Z.; Cao, Z.; Huang, T. UnitBox: An Advanced Object Detection Network. In *Proceedings of the 24th ACM International Conference on Multimedia*; ACM: New York, NY, 2016; pp 516–520. <https://doi.org/10.1145/2964284.2967274>
- [17] Rezatofighi, H.; Tsoi, N.; Gwak, J. Y.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized Intersection over Union: A Metric and a Loss for Bounding Box Regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; Long Beach, CA, USA, 2019; pp 658–666. <https://doi.org/10.1109/CVPR.2019.00075>

- [18] Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. *Proc. AAAI Conf. Artif. Intell.* **2020**, 34(7), 12993–13000. <https://doi.org/10.1609/aaai.v34i07.6999>
- [19] Zheng, Z.; Wang, P.; Ren, D.; Liu, W.; Ye, R.; Hu, Q.; Zuo, W. Enhancing Geometric Factors in Model Learning and Inference for Object Detection and Instance Segmentation. *IEEE Trans. Cybern.* **2022**, 52(8), 8574–8586. <https://doi.org/10.48550/arXiv.2005.03572>
- [20] Yang, Z.; Wang, X.; Li, J. EIoU: An Improved Vehicle Detection Algorithm Based on VehicleNet Neural Network. *J. Phys.: Conf. Ser.* **2021**, 1924 (1), 012001. <https://doi.org/10.1088/1742-6596/1924/1/012001>.
- [21] Sun, Y.; Wang, J.; Wang, H.; Zhang, S.; You, Y.; Yu, Z.; Peng, Y. Fused-IoU Loss: Efficient Learning for Accurate Bounding Box Regression. *IEEE Access* **2024**, 12, 37363–37377. <https://doi.org/10.1109/ACCESS.2024.3359433>
- [22] Xiao, Z.; Zhao, H.; Guo, J.; Yu, H.; Li, J. CRIoU: A Complete and Relevant Bounding Box Regression Method. In *Proceedings of the 2022 International Symposium on Advances in Informatics, Electronics and Education (ISAIEE)*; IEEE: Piscataway, NJ, 2022; pp 511–518. <https://doi.org/10.1109/ISAIEE57420.2022.00111>
- [23] Gevorgyan, Z. SIoU Loss: More Powerful Learning for Bounding Box Regression. *arXiv* **2022**, arXiv:2205.12740.
- [24] Ma, S.; Xu, Y. MPDIoU: A Loss for Efficient and Accurate Bounding Box Regression. *arXiv* **2023**, arXiv:2307.07662.
- [25] Zheng, Y.; Zhang, D.; Xie, S.; Lu, J.; Zhou, J. Rotation-Robust Intersection over Union for 3D Object Detection. In *Computer Vision – ECCV 2020*; Springer: Cham, Switzerland, 2020; pp 464–480. https://doi.org/10.1007/978-3-030-58565-5_28
- [26] Ma, S.; Xu, Y. FPDIoU Loss: A Loss Function for Efficient Bounding Box Regression of Rotated Object Detection. *Image Vis. Comput.* **2025**, 154, 105381. <https://doi.org/10.1016/j.imavis.2024.105381>.
- [27] Li, W.; Shang, R.; Ju, Z.; Feng, J.; Xu, S.; Zhang, W. Ellipse IoU Loss: Better Learning for Rotated Bounding Box Regression. *IEEE Geosci. Remote Sens. Lett.* **2024**, 21, 1–5. <https://doi.org/10.1109/LGRS.2023.334588>