mathematics

MDPI

Article

DPDN-YOLOv8: A Method for Dense Pedestrian Detection in Complex Environments

Yue Liu [1], Linjun Xu [1], Baolong Li [1], Zifan Lin [2,*] and Deyue Yuan [1]

[1] College of Electrical and Electronic Engineering, Changchun University of Technology, Changchun 130012, China; liuyue0423@ccut.edu.cn (Y.L.); 2202304030@stu.ccut.edu.cn (L.X.); 2202404020@stu.ccut.edu.cn (B.L.); 2202404070@stu.ccut.edu.cn (D.Y.)

[2] Department of Electrical and Electronic Engineering, School of Engineering, University of Western Australia, Crawley, Perth, WA 6009, Australia

* Correspondence: zifan.lin@uwa.edu.au

Abstract

Accurate pedestrian detection from a robotic perspective has become increasingly critical, especially in complex environments such as crowded and high-density populations. Existing methods have low accuracy due to multi-scale pedestrians and dense occlusion in complex environments. To address the above drawbacks, a dense pedestrian detection network architecture based on YOLOv8n (DPDN-YOLOv8) was introduced for complex environments. The network aims to improve robots' pedestrian detection in complex environments. Firstly, the C2f modules in the backbone network are replaced with C2f_ODConv modules integrating omni-dimensional dynamic convolution (ODConv) to enable the model's multi-dimensional feature focusing on detected targets. Secondly, the up-sampling operator Content-Aware Reassembly of Features (CARAFE) is presented to replace the Up-Sample module to reduce the loss of the up-sampling information. Then, the Adaptive Spatial Feature Fusion detector head with four detector heads (ASFF-4) was introduced to enhance the system's ability to detect small targets. Finally, to accelerate the convergence of the network, the Focaler-Shape-IoU is utilized to become the bounding box regression loss function. The experimental results show that, compared with YOLOv8n, the mAP@0.5 of DPDN-YOLOv8 increases from 80.5% to 85.6%. Although model parameters increase from 3×10^6 to 5.2×10^6, it can still meet requirements for deployment on mobile devices.

Keywords: pedestrian detection; YOLOv8; ODConv; CARAFE

MSC: 37M10

Academic Editors: Marjan Mernik, Juan Manuel Rendón-Mancha and Edgar Roman-Rangel

Received: 24 August 2025
Revised: 27 September 2025
Accepted: 10 October 2025
Published: 18 October 2025

Citation: Liu, Y.; Xu, L.; Li, B.; Lin, Z.; Yuan, D. DPDN-YOLOv8: A Method for Dense Pedestrian Detection in Complex Environments. *Mathematics* **2025**, *13*, 3325. https://doi.org/10.3390/math13203325

1. Introduction

Since service-oriented intelligent robots are developed to realize intelligent interaction with people, such as hotel service robots, medical service robots and so on, all of these robots need to perform their tasks quickly in a human–robot collaborative environment. At the same time, robotic systems developed with existing technologies should strictly prevent unintended physical contact—including collisions and blockages—during human–robot interaction, as these hazardous events may cause unpredictable harm, especially to vulnerable populations such as the elderly, children and persons with disabilities. Thus, the capability to accurately and rapidly detect individuals, groups and crowd densities becomes crucial when robots perform operational tasks. This perceptual capacity not only optimizes navigation routes and ensures efficient task completion, but also represents one

of the most challenging research frontiers in robotics. Furthermore, these advancements in computer vision-based pedestrian detection and analysis capabilities could be applied across multiple domains, including human–robot interaction [1], video surveillance [2], intelligent transportation systems [3], etc.

Recently, plenty of algorithms and network frameworks for automatic identification and localization of pedestrians from images or videos have been published [4], which can be broadly categorized into two aspects [5]: (i) two-stage detection methods based on candidate frames represented by R-CNN [6] and Faster-RCNN [7]; and (ii) single-stage detection methods represented by YOLO [8–12] and SSD [13]. The two-stage detection approach could be described as follows: Firstly, potential regions of interest are explicitly generated within the input image. Subsequently, each candidate region is resized to fixed dimensions based on its image characteristics. These regions then undergo feature extraction through a pre-trained convolutional neural network (CNN) model [14], followed by both final classification and regression operations to achieve the desired detection objectives. This method is slow, although the detection accuracy is high. Subsequently, a single-stage detection method has been proposed. It does not need to pre-calculate the pre-selected regions. Instead, it employs deep neural networks to extract features directly from input images. Then, the extracted feature maps are directly utilized to perform simultaneous object classification and coordinate regression for target localization. Finally, the expected test results are obtained. Nowadays, it serves as the mainstream approach in some modern detection technology fields [15]. It significantly helps to improve processing speed while maintaining optimal performance, particularly in small object detection scenarios. However, it still has limitations: various uncertainty factors occur during detection. This behavior fails to capture the complete feature representation of targets and leads to compromised accuracy, manifesting as both false positives and missed detections. These uncertainty factors mainly come from two aspects: (i) Visible regions with small-scale features often fall below the algorithm's detection threshold; (ii) Occluded targets are frequently overlooked during detection [16].

For example, Fang et al. [17] proposed integrating the concept of receiving field attention into the Conv and C2f modules, introducing a self-designed four-layer adaptive spatial feature fusion module and a small target dynamic head structure (DyHead-S) to improve the performance of pedestrian detection in dense scenarios. Dou et al. [18] proposed introducing a multi-scale feature fusion module and an improved non-maximum suppression (NMS) algorithm based on the YOLOv8 model to enhance the effectiveness and the superiority of the model. Peng et al. [19] proposed FedsNet, a pedestrian detection network based on RT-DETR. They constructed a lightweight backbone ResFastNet to reduce parameters/computation for faster speed, integrated EMA with the backbone for a new ResBlock to improve small target detection, adopted DySample as up-sampling to boost accuracy/robustness and used SIoU as loss to enhance accuracy and speed convergence. Liu et al. [20] proposed to explicitly model the semantic context through context-aware pedestrian detection with self-supervision of visual language semantics and used a self-supervised prototype semantic contrast (PSC) learning method based on the more explicit semantic context obtained from VLS. Li et al. [21] proposed integrating the receptive field attention into the convolution module, partially replacing the C2f module in the backbone network with the MobileViTv3 module, adding a TinyHead for detecting very small objects to the original detection head structure and adopting the boundary box regression loss function power-iouv2 to reduce false detections and missed detections. Ni et al. [22] proposed an improved object detection method based on the SSD framework. The improved ResNet50 network was used to enhance information transmission. The indoor scene context was extracted through the multi-scale Context Information Extraction

(MCIE) module and the indoor occlusion problem was solved with the help of the improved double-threshold non-maximum suppression algorithm (DT-NMS). Liu et al. [23] proposed using MobileViT as the lightweight backbone to reduce parameters, designing an SCNeck-neck network for lossless feature fusion and adopting DEHead for multi-scale target detection. Liu et al. [24] proposed the YOLOv8-CB lightweight multi-scale pedestrian detection algorithm, introducing the CFNet cascaded fusion network and the CBAM concern module to optimize the multi-scale feature semantics and location information representation, superimpose the BIFPN structure to fuse effective features and improve the performance of pedestrian detection.

To sum up, existing pedestrian detection algorithms have made certain progress. For example, the RMTP-YOLO model focuses on the detection of very small targets (e.g., distant pedestrians) and multi-scale feature capture. However, it has obvious shortcomings in terms of robustness in severely occluded scenarios, adaptability to low-light environments and detection performance in extremely dense scenarios. The YOLOv8-CB model focuses on lightweight deployment in in-vehicle scenarios and basic multi-scale fusion. Nevertheless, it has significant deficiencies in the detection accuracy of severely occluded pedestrians, the capture of distant small targets and cross-scenario generalization ability. Thus, existing pedestrian detection algorithms still face several challenges: Pedestrians exhibit diverse poses such as standing and walking, and when their features resemble the background, they are prone to false positives or false negatives. Pedestrians often occlude each other or are partially obscured by vehicles, buildings, etc., making it difficult for models to extract complete features and affecting detection reliability. Variations in the distance between pedestrians and the camera cause significant differences in target size, making small targets easy to overlook while large targets demand higher requirements for feature extraction and resource allocation. Addressing these gaps is critical, as unreliable detection directly compromises robot safety and operational efficiency.

To tackle these intertwined challenges, a novel dense pedestrian detection framework was proposed and named as Dense Pedestrian Detection Network based on YOLOv8n (DPDN-YOLOv8). Firstly, to enhance multi-scale pedestrian feature perception capabilities and address occlusion and background interference issues, the last three C2f modules in the YOLOv8n backbone were replaced with C2f_ODConv modules with Omni-Dimensional Convolutional layers. Through four-dimensional attention weight learning, introduced C2f_ODConv modules capture more comprehensive pedestrian contextual features. Secondly, to reduce information loss during up-sampling and preserve the semantic details of small targets, the original Up-Sample module is replaced with the Content-Aware Reassembly of Features up-sampling operator. Through the method of "dynamic kernel prediction + feature reassembly", the quality of neck feature extraction and fusion was improved. Simultaneously, to improve the detection accuracy of small-scale pedestrians, reduce missed detections and false detections in occluded and low-light environments and enhance the model's adaptive matching ability for pedestrians of different scales, a secondary innovation based on the original three detection heads of ASFF was presented via adding one dedicated detection head for small targets. Then, the four-detection-head ASFF-4 module was used to dynamically learn feature fusion weights, which reduces missed detections and false detections. Finally, to accelerate the network convergence speed, improve the bounding box regression accuracy of low-IoU hard samples and optimize the model's ability to handle complex samples, the Focaler-IoU and Shape-IoU were integrated into the model to amplify the loss of low-IoU samples and constrain the localization and shape consistency of predicted boxes.

2. Baseline Architecture

YOLOv8 [25], developed by Ultralytics in 2023, represents the next-generation general-purpose object detection framework. As an upgraded iteration of YOLOv5, this architecture achieves enhanced accuracy with reduced computational complexity through three optimized components (that is, backbone, neck and head network). The specific optimizations are as follows: (i) In the backbone [26], the architecture employs the C2f module to replace the C3 module from YOLOv5. This modification achieves dual improvements: one is reducing network depth and redundant parameters, to decrease model size and complexity; the other is enhancing gradient flow diversity and improves feature extraction capability. (ii) The neck [27] structure optimizes feature fusion by eliminating the 1 × 1 convolutional down-sampling layers present in YOLOv5. (iii) The head structure [28] innovatively integrates a decoupled head with anchor-free detection. This advancement enables direct prediction of target center points and aspect ratios, significantly reducing dependency on hyperparameters inherent in traditional anchor-based mechanisms. The design of the YOLOv8 network is shown in Figure 1.

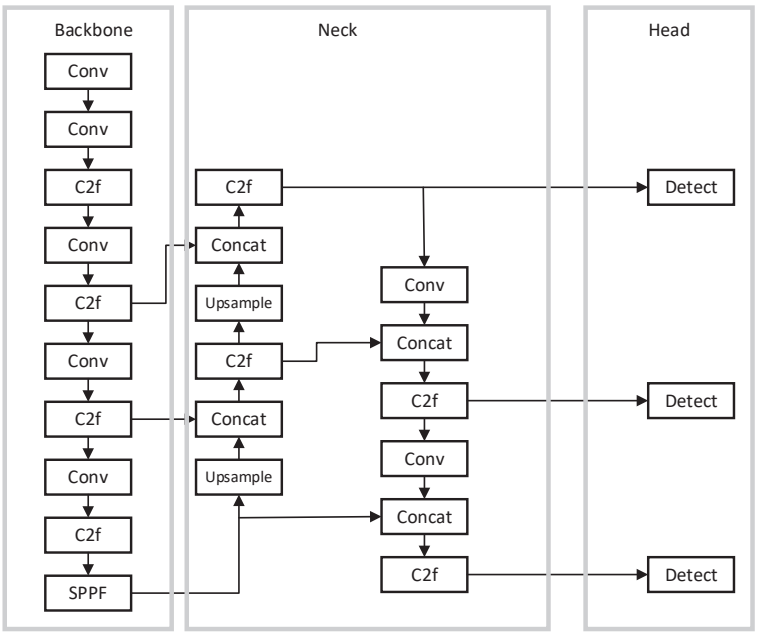

Figure 1. YOLOv8 network structure diagram.

3. Proposed DPDN-YOLOv8 Model

To enhance detection accuracy while reducing both missed detections and false alarms, an improved model based on the YOLOv8 network architecture was proposed and named as DPDN-YOLOv8. The network structure is illustrated in Figure 2. The orange sections indicate the improved modules. Firstly, to enhance the network's target perception capability during feature extraction, we propose the C2f_ODConv module, which replaces the last three C2f modules in YOLOv8's backbone network. To ensure accurate feature extraction across multi-scale input data while meeting the diverse feature requirements for dense pedestrians in complex environments, the model enhances its cross-scale perception and feature extraction capabilities for pedestrian global contextual information. Consequently, the network can carry out accurate feature extraction for different input data characteristics. This, in turn, meets the diversified dense pedestrian environments and enhances the model's ability of multi-scale perception and characterization of the pedestrians' global contextual information. Secondly, the original sampling operator is replaced by the CARAFE up-sampling operator, which exhibits a broader receptive field.

This architectural substitution enables the model to leverage semantic information from feature maps, thereby enhancing the neck network's capability in both feature extraction and fusion. Simultaneously, we replace the detection head with the Adaptive Spatial Feature Fusion detector head with ASFF-4 module, which facilitates adaptive weight learning for spatial feature fusion to address cross-scale discrepancies, ultimately improving detection accuracy. Finally, the Focaler-Shape-IoU is employed as the loss function for bounding box regression. This advanced loss function is designed to address three key aspects: (1) minimizing the influence of low-quality samples on anchor box regression, (2) accelerating model convergence rate and (3) significantly improving detection precision for small-scale targets.

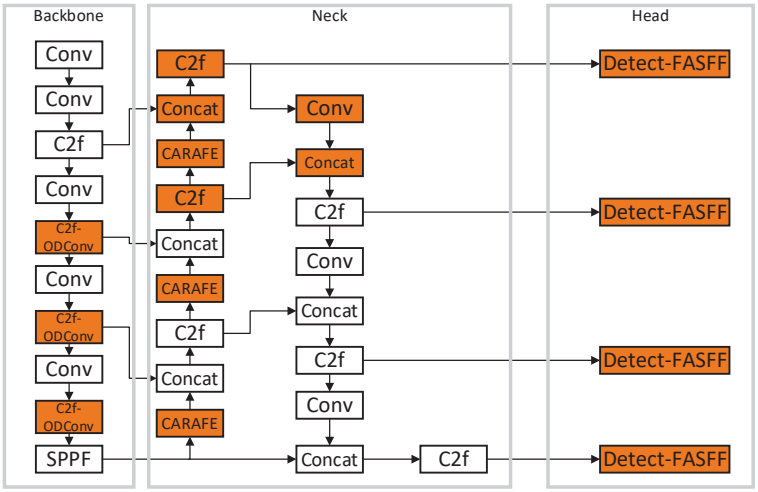

Figure 2. DPDN-YOLOv8 network architecture diagram.

3.1. Omni-Dimensional Dynamic Convolution

In general convolution operations, convolutional layers primarily serve to enhance model performance. Existing approaches, such as adding new layers or stacking existing ones, inevitably increase computational demands and adversely affect detection efficiency. To address this, we integrate ODConv [29] into the C2f module, presenting the novel C2f_ODConv module. This enhanced module enables parallel learning of convolutional kernel features across four dimensions of the kernel space; consequently, it captures more comprehensive contextual information and diverse feature representations, thereby significantly improving detection accuracy. Moreover, this approach achieves enhanced precision in target detection.

Next, the details of the ODConv module are discussed. It employs an advanced multi-attention mechanism with parallel processing strategy to simultaneously capture attention across four dimensions of the convolutional kernel space: (i) kernel spatial size, (ii) input channels, (iii) output channels and (iv) kernel count. This architecture subsequently utilizes the attention-weighted convolutional kernels to achieve high-precision recognition of complex features in dense pedestrian scenarios.

$$Y = (\alpha_{w1} \odot \alpha_{f1} \odot \alpha_{c1} \odot \alpha_{s1} \odot W_1 + \cdots + \alpha_{wn} \odot \alpha_{fn} \odot \alpha_{cn} \odot \alpha_{sn} \odot W_n)^* X \quad (1)$$

where, $Y \in R^{H \times W \times C_{out}}$ denotes the feature dimension of the output channels, while C_{out} represents the spatial dimensions (H × W) of the feature map. The input channel feature dimension is characterized by $X \in R^{H \times W \times C_{in}}$, whereas its spatial extent is defined by a size of H × W. α_{wi} represents the kernel-wise (W_i) attention mechanism; α_{si} stands for the spatial attention mechanism over the k×k convolutional kernel space; and both α_{ci} and α_{fi} represent the input-channel and output-channel attention mechanisms, respectively.

The omni-dimensional dynamic convolution structure is illustrated in Figure 3. The input feature X is compressed into a feature vector through Global Average Pooling (GAP). Subsequently, this vector is mapped to a low-dimensional space via a Fully Connected (FC) layer, with negative values being zeroed out by the ReLU activation function. Finally, the processed feature vector is distributed to four parallel computational branches. The four branches, each processed through a fully connected (FC) layer followed by a Sigmoid activation function, enable the initial three branches to capture attention to scalars for three key convolutional kernel attributes: spatial dimensions, input channels and output channels, respectively. The final branch derives attention to scalars along the kernel's channel-count dimension. Finally, the attention scalars from all four dimensions are convolved with the input features X to generate the output features Y.

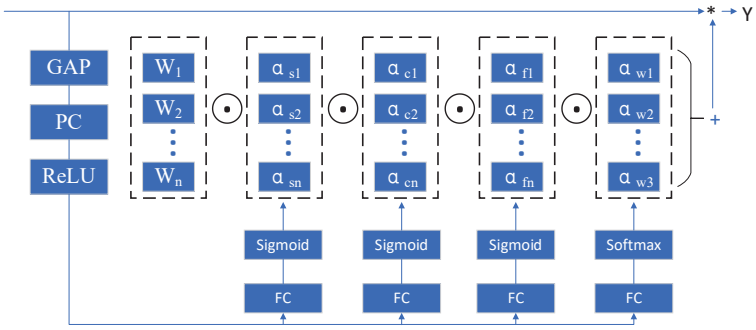

Figure 3. ODConv structure diagram. The symbol "∗" represents convolution operation; the symbol "+" represents element-wise addition.

3.2. CARAFE Up-Sampling Operator

The incorporation of the CARAFE [30] up-sampling operator into the neck network represents the second innovation of this study. In the complex scenario of pedestrian detection, the foundational model exhibits deficiencies in semantic information and insufficient size of sensory regions during the super-resolution process. These deficiencies lead to suboptimal feature fusion performance. To address this issue, the original up-sampling operator in the YOLOv8 baseline model was replaced with the CARAFE up-sampling operator. The architecture of the new network is illustrated in Figure 4. As shown in Figure 4, the diagram visually presents the CARAFE network architecture, with the upper section illustrating the Up-Sampling Kernel Prediction Module (showing the process from feature compression to kernel normalization) and the lower section depicting the Feature Reorganization Module (demonstrating how input features are mapped and recombined).

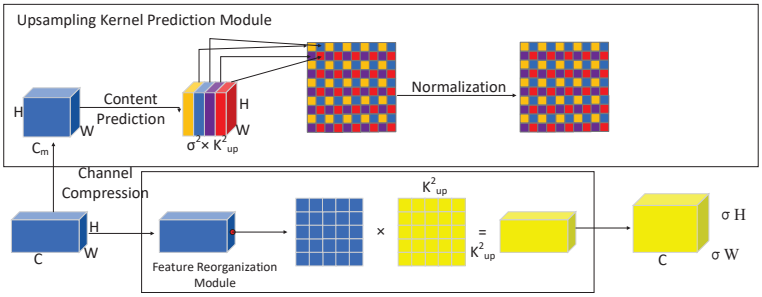

Figure 4. CARAFE network architecture. In the structural diagram of the CARAFE upsampling operator, different colors are used for visual differentiation of feature channels or feature groups. CARAFE upsampling involves operations of the upsampling kernel prediction module and the feature reconstruction module, and the color coding clarifies how features from different channels are processed and fused during the kernel prediction procedure.

The CARAFE up-sampling operator consists of a kernel prediction module and a feature reassembly module. In the kernel prediction module, the input feature map has dimensions of $C \times H \times W$, where C represents the number of channels, while H and W denote the height and width of the feature map, respectively. In the initial stage, the feature map undergoes convolutional compression to a specified channel dimension $C_m \times H \times W$ to reduce computational requirements. Subsequently, content prediction yields an up-sampling kernel with dimensions $\sigma H \times \sigma W \times K_{\text{up}}^2$. This is followed by normalization to obtain convolution kernels with weights summing to 1, ensuring the feature map's magnitude remains unaffected. Within the feature reassembly module, information from the output feature map is mapped back to the input feature map. A region centered at $K_{\text{up}} \times K_{\text{up}}$ is then extracted and subjected to dot product operations with the up-sampling kernel, ultimately producing an output feature map with dimensions $\sigma H \times \sigma W \times C$.

3.3. ASFF-4 Detection Heads

The introduction of the ASFF-4 module to YOLOv8's detection head constitutes the third innovation. In the YOLOv8 object detection algorithm, the Head module is primarily responsible for final object detection and bounding box prediction. It employs a Detect structure as its Head module, which has demonstrated strong detection performance across numerous specific scenarios. However, in scenarios involving complex tasks such as occluded or overlapping pedestrian detection, it faces significant challenges. This is primarily attributed to the diversity in pedestrian feature scales, substantial feature variations and overlapping annotation regions, all of which contribute to suboptimal detection outcomes. Under such circumstances, certain objects may obstruct the neck network from extracting critical feature information, resulting in target information loss. This fundamentally explains the persistently high false alarm rate observed in pedestrian detection systems.

To solve the above problems, the improved structure based on ASFF [31] was introduced, which adds a novel detection head structure named ASFF-4 to the baseline three-head detection architecture of ASFF. The newly added small-object detection head is designed to achieve secondary extraction of small targets. This structural design achieves three key improvements: one is to enhance multi-scale object detection capability for pedestrian recognition, the second is extraction of more profound hierarchical features leading to notable accuracy gains, and the third is effective mitigation of feature degradation in multi-scale fusion processes.

The process of calculating the ASFF-4 detection head is shown in Equation (2):

$$y_{ij}^{l} = \alpha_{ij}^{l} \cdot x_{ij}^{1 \to l} + \beta_{il}^{l} \cdot x_{ij}^{2 \to l} + \gamma_{ij}^{l} x_{ij}^{3 \to l} + \delta_{ij}^{l} x_{ij}^{4 \to l} \tag{2}$$

where the term is used to denote the fusion features generated on layer l that are ultimately employed for prediction. refers to the input value of the feature vector at (x, y) in the n-th layer feature map prior to the fusion of features on layer l. The term signifies the learnable weight parameters of the feature maps of four distinct levels up to the lth layer. These parameters correspond to the weight parameters of the feature maps of Level 0, Level 1, Level 2 and Level 3, which are obtained through back-propagation convolution of 1 × 1 during the training process. The feature maps are obtained through 1 × 1 back-propagation convolution during the training phase. The values of the control weight parameters are constrained to the interval [0, 1] and the sum of is normalized to 1. The main schematic structure of the ASFF-4 module is shown in Figure 5 below.

As shown in the figure, it presents the structure of the ASFF-4 module, which enables adaptive spatial feature fusion for multi-scale pedestrian detection. On the left side, the feature maps are processed to generate hierarchical feature levels — Level 3 preserves fine-

grained details for small or occluded pedestrians, while Level 0 captures global contextual information for large or long-distance targets. Subsequently, four ASFF modules take these multi-level features as inputs and each module adaptively learns feature weights and optimal fusion ratios based on spatial positions and target sizes to resolve the inconsistency of cross-scale features. Finally, the fused features from each ASFF module are fed into prediction heads with matching resolutions, thereby achieving accurate detection of pedestrians at different scales, enhancing the model's multi-scale information integration capability, reducing missed detections and false detections and thus improving the robustness of dense pedestrian detection.

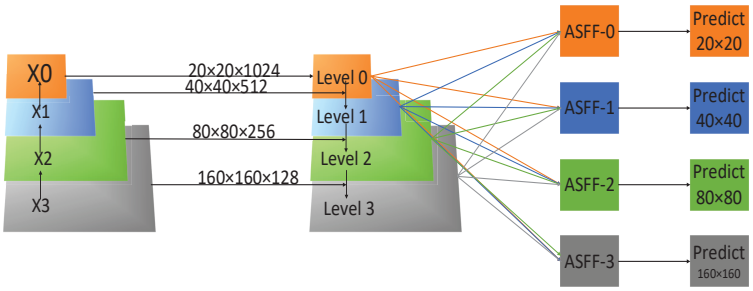

Figure 5. ASFF-4 module structure.

The demonstrated ASFF-4 module adaptively adjusts feature weights to ensure the model autonomously selects the most effective features across different spatial locations, thus preventing inadequate feature representation. Moreover, it dynamically assigns optimal fusion ratios based on actual scenarios and target sizes to minimize both missed detections and false alarms.

3.4. Loss Function

The YOLOv8n architecture initially employs CIoU loss [32] for bounding box prediction, which accounts for both overlap area and aspect ratio differences. However, its inability to adaptively weight samples based on detection difficulty results in the aspects of delayed optimization convergence, compromised model generalization and ultimately constrained detection performance enhancement. Therefore, the insights from Focaler-IoU [33] and Shape-IoU [34] were incorporated to propose an enhanced loss function. The implementation is illustrated in Figure 6.

Focaler-IoU focuses on low-IoU challenging samples (e.g., occluded or small-scale pedestrians) by amplifying the loss contribution of hard-to-detect samples via focal weighting, while simultaneously constraining spatial localization accuracy. Shape-IoU achieves "object shape and scale matching" by imposing shape penalty terms to constrain the consistency between bounding boxes and the actual shape and scale of pedestrians. The specific formulas are as follows:

$$L_{\text{Focaler-IoU}} = 1 - \text{IoU}^F + d^{\text{Shape}} \tag{3}$$

$$L_{\text{Shape-IoU}} = 1 - \text{IoU} + \Omega^{\text{shape}} \tag{4}$$

The traditional intersection-to-parallel ratio IoU is defined as

$$IoU = \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} \tag{5}$$

where B denotes the prediction frame, while B^{gt} signifies the real frame. IoU is a metric that quantifies the degree of overlap between the prediction frame and the real frame. As demonstrated, higher IoU values correspond to greater prediction accuracy. In dense

pedestrian detection scenarios, failure to adequately account for sample difficulty and bounding box shape/size characteristics may adversely affect regression performance.

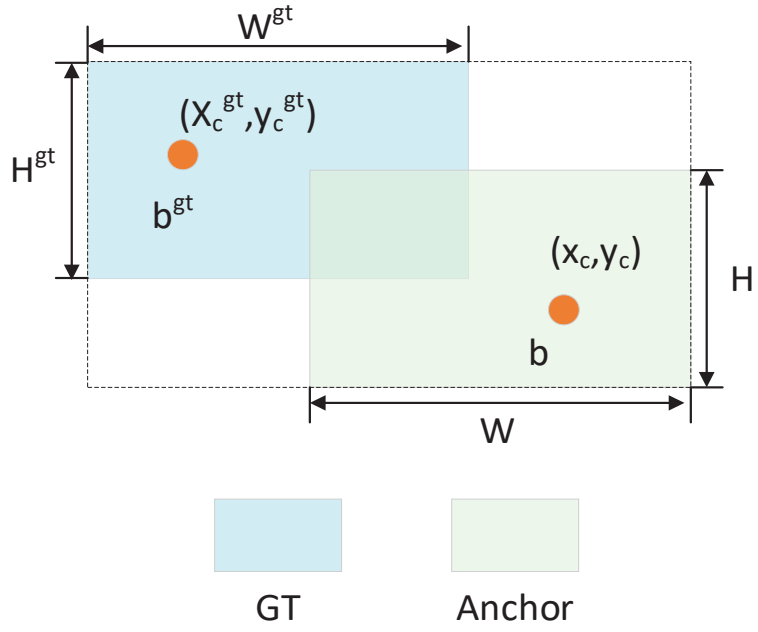

Figure 6. Schematic diagram of Focaler-Shape-IoU parameters.

In order to achieve the defined objectives, two learnable parameters (dd and uu) were employed into the Focaler-IoU module. One is to increase the loss weight of low IoU samples. The other is designed to enhance the model's focus on small objects and low-overlap samples; the details are elaborated as follows:

$$\mathrm{IoU}^F = \begin{cases} 0 & \text{if IoU} < d \\ \frac{\mathrm{IoU}-d}{u-d} & \text{if } d \le \text{IoU} \le u \\ 1 & \text{if IoU} > u \end{cases} \tag{6}$$

d^{Shape} is a penalty term for the center point distance between the predicted bounding box and the ground-truth bounding box, constraining the accuracy of spatial localization.

$$d^{Shape} = \frac{HH \times (x_c - x_c^{gt})^2}{c^2} + \frac{WW \times (y_c - y_c^{gt})^2}{c^2} \tag{7}$$

where x_c and y_c are the coordinates of the center of the prediction box; x_c^{gt} and y_c^{gt} are the coordinates of the center of the real box; and c is the diagonal length of the minimum circumscribed rectangle that contains the prediction box and the true box. WW and HH are the weight coefficients calculated based on the target aspect ratio. The calculation formula for the weight coefficient is as follows:

$$WW = \frac{2 \times (W^{gt})^{scale}}{(W^{gt})^{scale} + (H^{gt})^{scale}} \tag{8}$$

$$HH = \frac{2 \times (H^{gt})^{scale}}{(W^{gt})^{scale} + (H^{gt})^{scale}} \tag{9}$$

where W^{gt} and H^{gt} are the width and height of the real frame, respectively, and $scale$ is the scaling factor related to the size of the target in the dataset. Additionally, Ω^{shape} is a

shape loss term that adjusts the discrepancy between width and height through learnable weighting factors, formulated as

$$\Omega^{shape} = \sum_{t=H,W} \left(1 - e^{-\omega_t}\right)^4 \tag{10}$$

where both ω_W and ω_H could be thought of as the shape loss factors between width and height, respectively. Their calculation formulas could be described as follows:

$$\begin{cases} \omega_W = HH \times \frac{|W - W^{gt}|}{max(W, W^{gt})} \\ \omega_H = WW \times \frac{|H - H^{gt}|}{max(H, H^{gt})} \end{cases} \tag{11}$$

Finally, by combining the Focaler-IoU loss function with the Shape-IoU loss function, to maintain sensitivity for challenging samples, the core components $1 - \text{IoU}^F$ and d^{Shape} of Focaler-IoU are directly retained. Shape-IoU is introduced to impose shape/scale constraints, incorporating Ω^{shape} into the loss function with a weighting coefficient of 0.5 for the shape penalty term. The Focaler-Shape-IoU loss function could be obtained as follows:

$$L_{Focaler-ShapeIou} = 1 - \text{IoU}^F + d^{Shape} + 0.5 \times \Omega^{shape} \tag{12}$$

In summary, within pedestrian detection scenarios—where targets are predominantly distributed in dense, non-uniform clusters—the Focal-Shape-IoU method achieves superior performance. Firstly, it adaptively tunes sampling focus through a linear-interval mapping mechanism. Also, it precisely optimizes shape and scale features for complex samples. Secondly, the dual approach enhances accuracy and recognition reliability, demonstrating measurable improvements in both precision and robustness.

4. Results

To verify the effectiveness of the DPDN-YOLOv8 model, we conducted both ablation studies and comparative experiments. These experiments systematically evaluated the feasibility and applicability of the proposed network.

4.1. Experimental Environment

In this subsection, to experimentally validate the effectiveness of the proposed network, the model was trained for 300 epochs using a batch size of 8, the stochastic gradient descent (SGD) optimizer and images of size 640 × 640. The weight decay factor was set to 0.0005, the initial learning rate to 0.01 and the learning rate momentum factor to 0.937. Detailed experimental environment configurations are presented in Table 1.

Table 1. Experimental environment configuration.

Configuration	Version
Operating system	Ubuntu 18.04
CPU	Intel Xeon Silver 4314
GPU	RTX 3090
Computing platform	CUDA11.6
RAM	24G
Deep learning framework	Pytoch 1.8.2
Programming language	Python 3.8

4.2. Dataset

A series of experiments were conducted using the CrowdHuman [35] dataset. The dataset comprises 15,000 images in the training set, 5000 images in the test set and 4370 images in the validation set, totaling 24,370 images. It contains approximately

470,000 annotated instances, with each image averaging 23 individuals and exhibiting various occlusions.

These images cover various existing dense crowd scenarios in complex environments. Although each pedestrian instance is annotated as either head, visible body parts or full body, only full-body labels were adopted for the focus of this study. Some of the images in the dataset and their annotations are shown in Figure 7.

Figure 7. Dataset and annotations.

4.3. Assessment of Indicators

This experiment evaluates model performance using six metrics: *Precision* (*P*), *Recall* (*R*), mAP@50, mAP@50:95, model size and floating-point operations per second (GFLOPs).

Precision measures the proportion of instances predicted as positive samples by the model that are actually positive. It reflects the accuracy of the model in predicting positive samples. *Precision* can be calculated by the following equation:

$$Precision = \frac{TP}{TP + FP} \tag{13}$$

where TP denotes true positive and FP denotes false positive.

The recall rate is a metric that measures a model's ability to correctly predict all positive samples, also known as the total inspection rate. Its calculation formula is as follows:

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

where FN denotes false negatives and TP denotes true positives.

The $F1$ score is a measure of the harmonic mean of precision and recall, providing a comprehensive evaluation of these two metrics. A higher $F1$ score indicates better performance in both precision and recall. The formula for calculating the $F1$ score is

$$F1 = \frac{2 \times P \times R}{P + R} \tag{15}$$

mAP50 refers to the mean average precision of the model's predictions when IoU (Intersection over Union) is no less than 0.5. This metric is commonly used for benchmarking. mAP50-95 refers to calculating AP (Average Precision) when IoU is in the range of [0.5, 0.95] at intervals of 0.05 and then averaging the obtained AP values.

GFLOPs: a commonly used metric to reflect the computational complexity of a model. If GFLOPs are too high, it may restrict the deployment and operational efficiency of the model on devices with limited computing power.

4.4. Ablation Experiments

To verify the effectiveness of various improvements in DPDN-YOLOv8, ablation experiments were conducted on the CrowdHuman dataset to comprehensively analyze the performance differences of different improvement methods. Using YOLOv8n as the baseline model, a series of experiments were progressively enhanced for the C2f module in the backbone, up-sampling operator, detection head and loss function via a series of experiments, demonstrating the impact of each module on model performance. The "√" indicates the application of corresponding improvement methods to the YOLOv8n network model. The experimental results are shown in Table 2.

Table 2. Ablation experiment table.

Experiment	ODConv	CARAFE	ASFF-4	Focaler-Shape-IoU	F1 (%)	mAP_{50} (%)	mAP_{50-95} (%)	Parameters/10^6	GFLOPs
1	YOLOv8n				76.9	80.5	49.5	3.0	8.2
2	√				77.3	82.2	51.4	3.0	6.9
3		√			78.0	82.8	52.2	3.1	8.4
4			√		79.5	84.5	54.6	4.3	15.2
5				√	78.0	82.8	52.0	3.0	8.1
6	√		√		79.6	84.6	54.8	5.0	14.7
7	√	√			77.3	82.9	51.7	3.2	7.2
8		√	√		79.8	84.7	54.9	4.5	16.3
9	√	√	√		79.8	84.8	55.0	5.2	15.4
10	√	√	√	√	79.8	85.6	55.1	5.2	15.7

From Table 2, in Experiment 2, based on the YOLOv8 foundation model, the last three C2f modules in the backbone network were replaced with C2f_ODConv. This modification resulted in respective improvements of 0.3%, 0.4%, 1.7% and 1.9% in P, R, mAP@0.5 and mAP@0.5:0.95 metrics. The results demonstrate that the feature extraction capability of the novel model could be enhanced via C2f_ODConv module to make sure more effective processing of occlusions and interfering features have been obtained in dense pedestrian detection tasks.

In Experiment 3, the traditional up-sampling operator is also replaced by the CARAFE up-sampling operator, which could improve the quality of the up-sampled feature maps via their semantic information and enhance the ability of the necking network to extract and fuse image features. Compared with the original model, the improvements were 0.9%, 1.1%, 2.3% and 2.7% in P, R, mAP@0.5 and mAP@0.5:0.95, respectively. Thus, the effectiveness of the replaced up-sampling operator has been validated.

In Experiment 4, the introduction of the ASFF-4 detection head enabled the model to learn adaptive spatial fusion weights at different spatial locations. This enhancement improved both the localization accuracy and feature extraction capability for heavily occluded targets and ensured more precise object localization. As shown in Table 2, the improved algorithm achieved significant gains of 1.1%, 3.6%, 4% and 5.1% in P, R, mAP@0.5 and mAP@0.5:0.95 metrics, respectively. These results demonstrate that the modified component can adaptively integrate pedestrian features from different network levels; thus, the accuracy has been enhanced.

In Experiment 5, the improved Focaler-Shape-IoU loss function was adopted as the bounding box regression function. Compared with the baseline algorithm, all evaluation metrics showed improvements, enhancing the network's detection accuracy. Then, the results demonstrate that further optimization of the loss function can significantly enhance the overall performance of the improved network, with respective improvements of 0.9%, 1%, 2.3% and 2.5% observed in P, R, mAP@0.5 and mAP@0.5:0.95 metrics. These findings indicate that the Focaler-Shape-IoU loss function could effectively improve the capability to handle complex samples.

As shown in Table 2, Experiments 6, 7 and 8, incorporating multiple improvements, demonstrate significant enhancements in P, R, mAP@0.5 and mAP@0.5:0.95 compared to Experiment 1. Then, Experiment 9 combines the first three improvements, achieving respective increases of 0.9%, 4.2%, 4.3% and 5.5% in these metrics relative to Experiment 1. Moreover, Experiment 10 integrates all proposed enhancements. Notably, the mAP@0.5 value reaches 85.6%, representing a 5.1% gain over the original algorithm, with other metrics showing commensurate improvements. The comprehensive analysis of the ablation study results indicates that despite an increase in parameters, our approach achieved optimal performance across all key metrics: the F1 score, mAP_{50} and $mAP_{50\text{-}95}$ all reached the highest values in this ablation experiment. The above description indicates that the improved algorithm strikes a favorable balance between model complexity and detection accuracy, validating its superiority in overall performance.

4.5. Loss Function Comparison Experiment

A comparative analysis with existing methods(e.g., EIou [36], SIou [37], GIou [38], FocalerioU, ShapeioU, Focal-SIou [39] and Focaler-EIou [40] was conducted to determine the most suitable loss function and verify the effectiveness of Focaler-Shape-IoU. The experimental results in Table 3 demonstrate that Focaler-Shape-IoU emerges as the most effective approach, which has achieved improvements of 1.3% in mAP@0.5 and 0.9% in Precision. These findings further validate the superiority of the proposed loss function over the baseline CIoU loss function. Meanwhile, the visualization results (see Figure 8) more intuitively show that the curve corresponding to Focaler-Shape-IoU has a faster decreasing speed of bounding box loss during the training process; also, with the increase in the number of training epochs, in the later stage of training the loss value is significantly lower than that of other compared loss functions. As is well known, the lower the bounding box loss, the higher the accuracy of the model's prediction for the target bounding box. From the perspective of gradient stability, Focaler-Shape-IoU effectively avoids the issues of gradient explosion and vanishing by optimizing the mathematical form of the loss function, thereby ensuring smoother gradient updates during the training process. In terms of sample difficulty modeling, a dynamic sample difficulty perception mechanism is introduced. It can accurately distinguish the difficulty and ease levels of samples, assigning higher loss weights to deal with the difficult samples and enhance the model's ability for the learn objects in complex scenarios. Therefore, Figure 8 intuitively verifies that Focaler-Shape-IoU could more effectively optimize bounding box regression performance during training, fully demonstrating its advantages over other loss functions.

Table 3. Comparison results of different loss functions.

Loss Function	P/(%)	R/(%)	mAP@0.5/(%)	mAP@0.5:0.95/(%)
base	85.1	70.2	80.5	49.5
GIou	85.6	71.1	82.2	51.7
SIou	85.8	71.0	82.2	51.6
EIou	85.7	71.0	82.5	51.9
Focaleriou	85.8	71.0	82.7	52.0
Shapeiou	86.0	71.2	82.6	52.0
Focal_SIou	86.0	71.1	82.6	52.0
Focaler_EIou	85.7	70.8	82.4	51.7
Focaler-Shape-IoU	86.0	71.2	82.8	52.0

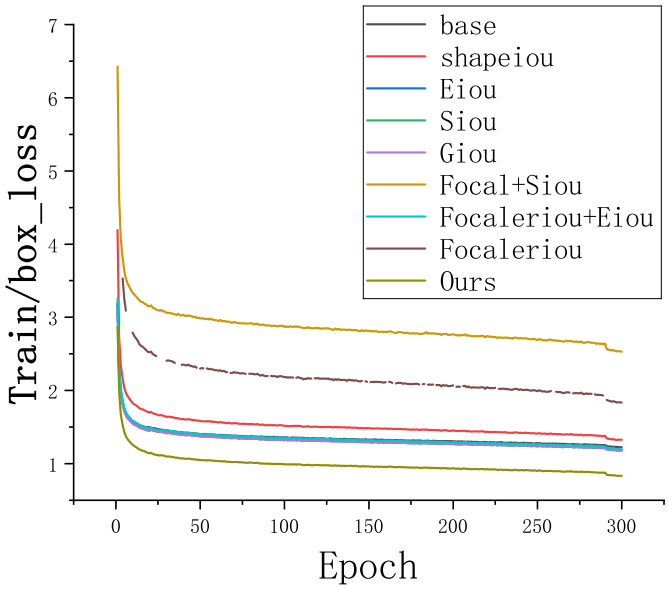

Figure 8. Visual comparison of different loss functions.

4.6. Comparative Experiments of Different Algorithms

Here, we conduct a comprehensive comparison between our proposed network (DPDN-YOLOv8) and other state-of-the-art object detection models, including two-stage algorithms (e.g., Faster R-CNN), single-stage YOLO series (SSD, RetinaNet [41], RTDETR, SOLIDER,YOLOv5 [42], YOLOv7 [43], YOLOv10 [44], YOLOv11 [45]), and the baseline YOLOv8n model. The experimental results are systematically presented in Table 4.

The experimental results demonstrate that the proposed network achieves an mAP@0.5 of 85.6%, representing a 5.1% improvement over the baseline YOLOv8n model. Furthermore, comparative analysis reveals its significant advantages in both parameter efficiency and detection accuracy when benchmarked against other state-of-the-art networks.

Table 4. Comparative experiments of different object detection algorithms in the CrowdHuman dataset.

Model	mAP@0.5/(%)	mAP@0.5:0.95/(%)	Parameters/(10^6)	GFLOPS
Faster RCNN	73.5	43.0	45.1	—
SSD	68.4	45.6	26.3	—
RetinaNet	81.8	49.8	45.7	—
RTDETR	83.5	54.1	19.7	56.9
SOLIDER	82.2	52.9	43.9	259.5
YOLOv5	79.8	46.5	1.8	4.1
YOLOv7-tiny	82.0	49.4	6.0	13.2
YOLOv8n	80.5	49.5	3.0	8.2
YOLOv10	80.0	50.2	2.6	8.2
YOLOv11	80.2	48.9	2.6	6.3
DPDN-YOLOv8	85.6	55.1	5.2	15.7

The comparative analysis with other YOLO variants demonstrates that DPDN-YOLOv8 achieves significant performance improvements, observed as follows:

(i) 5.8% (mAP@0.5) and 8.6% (mAP@0.5:0.95) over YOLOv5;

(ii) 3.6% (mAP@0.5) and 5.7% (mAP@0.5:0.95) versus YOLOv7-tiny, with an additional 0.8×10^6 parameter reduction;

(iii) 5.6% (mAP@0.5) and 4.9% (mAP@0.5:0.95) compared to YOLOv10;

(iv) 5.4% (mAP@0.5) and 6.2% (mAP@0.5:0.95) against YOLOv11.

To sum up, the key accuracy metrics of DPDN-YOLOv8—specifically mAP@0.5 and mAP@0.5-0.95—achieve the highest values among all compared models, demonstrating

significant advantages over classical two-stage models and RetinaNet. Compared with the baseline models in the YOLO series, the proposed algorithm also demonstrates a remarkably significant improvement in accuracy. In terms of model size, DPDN-YOLOv8 has 5.2×10^6 parameters, which can be regarded as characteristic of a lightweight model. This is because its size is significantly smaller than other models, such as RTDETR and SOLIDER. Meanwhile, its accuracy far surpasses that of smaller lightweight models like YOLOv5. In terms of computational complexity, its GFLOPS is 15.7, balancing accuracy with reasonable computational speed while remaining far below SOLIDER and RTDETR. The parameter count and computational cost of DPDN-YOLOv8 are both within the manageable range of mobile devices, and its accuracy far surpasses that of other models. Therefore, even with a relatively large number of parameters, this method can still be successfully deployed on mobile devices.

To validate the statistical reliability of the performance improvements, key metrics across multiple replicate experiments were analyzed: an independent samples *t*-test compared metric differences between DPDN-YOLOv8 and the baseline model YOLOv8n, while one-way ANOVA assessed performance variations between DPDN-YOLOv8 and models such as YOLOv5, YOLOv7-tiny and RTDETR. Results indicate that DPDN-YOLOv8 exhibits statistically significant differences from comparison models across key metrics, confirming that the performance gains are not random fluctuations but rather effective contributions from the proposed method.

4.7. Generalization Experiment

To validate the generalization capability of the DPDN-YOLOv8 algorithm, this section conducts comparative experiments on the Cityperson dataset. It comprises 5000 images captured by vehicle-mounted cameras across 27 European cities. The images are divided into training (2975 images), validation (500 images) and test (1575 images) sets. Each image contains an average of 7 people, featuring various small-scale objects and occlusions. Annotations include visible regions and full-body regions. The experimental results are shown in Table 5:

Table 5. Generalizability experimental results for Cityperson.

Model	mAP@0.5/(%)	mAP@0.5:0.95/(%)	Parameters/(10^6)	GFLOPS
YOLOv8n	60.5	37.3	3.0	8.2
DPDP-YOLOv8	65.9	40.7	5.2	15.7

As shown in Table 5's comparative experiments on the Cityperson dataset, DPDP-YOLOv8 achieves significantly improved detection accuracy compared to YOLOv8n, indicating superior pedestrian detection performance. Simultaneously, the model's parameter count increased from 3×10^6 to 5.2×10^6 and FLOPs rose from 8.1 to 15.7, reflecting a moderate growth in computational complexity. Despite this increase, the model parameters remain suitable for deployment in mobile object detection applications.

4.8. Visualization Results

To further validate the performance of the DPDN-YOLOv8 model, both YOLOv8n and DPDN-YOLOv8 models were trained on the CrowdHuman dataset. A comparative analysis was conducted on the training results, evaluating four key metrics: precision, recall, mean Average Precision at IoU = 0.5 (mAP@0.5) and mean Average Precision over IoU thresholds from 0.5 to 0.95 (mAP@0.5:0.95). Figure 9 provides visual representations that offer more intuitive comparisons of these metrics. Thus, the proposed algorithm consistently outperformed the baseline model.

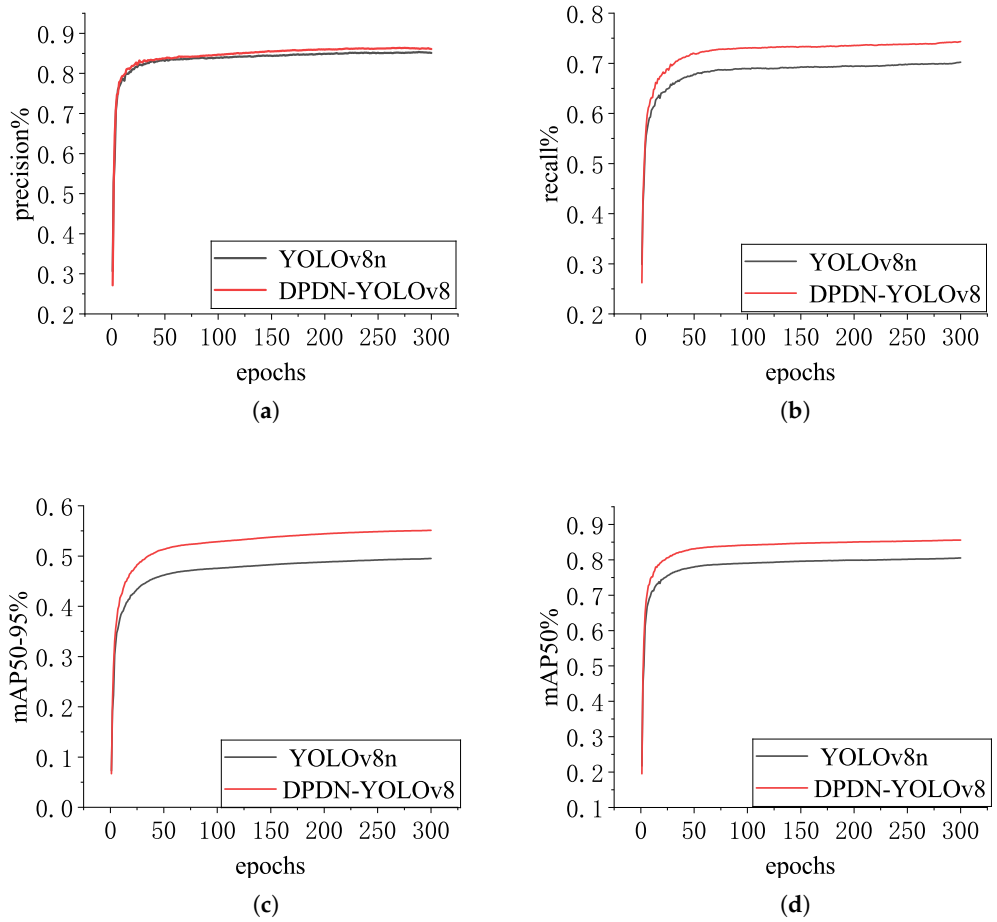

Figure 9. Metrics comparison between YOLOv8 and DPDN-YOLOv8. (**a**) Precision Comparison Chart. (**b**) Recall Comparison Chart. (**c**) mAP50-95 Comparison Chart. (**d**) mAP50 Comparison Chart.

Figure 9 presents the comparison results between YOLOv8 and DPDN-YOLOv8 in four core detection metrics: precision, recall, mAP@0.5:0.95 and mAP@0.5, clearly illustrating the performance evolution and differences between them during the training process. In terms of precision, as the number of training iterations increases, two models rise rapidly and converge—with DPDN-YOLOv8's accuracy consistently remaining higher than that of YOLOv8n—and eventually stabilize above 85%. This indicates that the improved model exerts stricter control over the correctness of detection results and generates fewer false positives. In terms of Recall, two models exhibit an upward convergence trend. Then, the proposed DPDN-YOLOv8 demonstrates a significantly higher recall than the other model, eventually stabilizing above 75%. This indicates that it can more effectively capture real pedestrian targets while maintaining a significantly lower missed detection rate. In terms of mAP@0.5:0.95, the proposed model outperforms YOLOv8n throughout the entire training process, eventually stabilizing above 55%. Moreover, as training progresses, the gap between them gradually widens. Therefore, it proves that the proposed model possesses stronger robustness and significantly enhances fine-grained localization capability under varying localization precision requirements. On the mAP@0.5 curve, DPDN-YOLOv8 outperforms YOLOv8n significantly, eventually stabilizing above 85%. This further validates its advantages in fundamental detection capabilities. Overall, DPDN-YOLOv8 consistently and comprehensively surpasses the benchmark model YOLOv8n across four key metrics. These results indicate that the proposed method effectively enhances the model's detection performance for dense pedestrians, reduces missed detections and false positives, and boosts robustness in scenarios with varying localization precision requirements.

To visually demonstrate the detection performance of DPDN-YOLOv8 in complex scenarios, we selected three challenging scenarios—partial occlusion, severe occlusion and low-light conditions—and conducted comparative visualization experiments between YOLOv8n and DPDN-YOLOv8. The results are shown in Figure 10. The detection results of YOLOv8n and DPDN-YOLOv8 in three challenging scenarios are compared: partial occlusion, heavy occlusion and low light. In partial occlusion, DPDN-YOLOv8 can more accurately detect occluded targets with precise bounding boxes and high confidence. For heavy occlusion, it outperforms YOLOv8n by effectively identifying pedestrians with only partial features exposed, reducing missed detections. In low-light environments, DPDN-YOLOv8 also demonstrates strong adaptability, stably detecting pedestrians with reliable bounding boxes and confidence scores. Overall, DPDN-YOLOv8 exhibits excellent advantages: superior occlusion resistance, strong adaptability to low-light conditions and higher detection accuracy and confidence, making it more robust in complex scenarios.

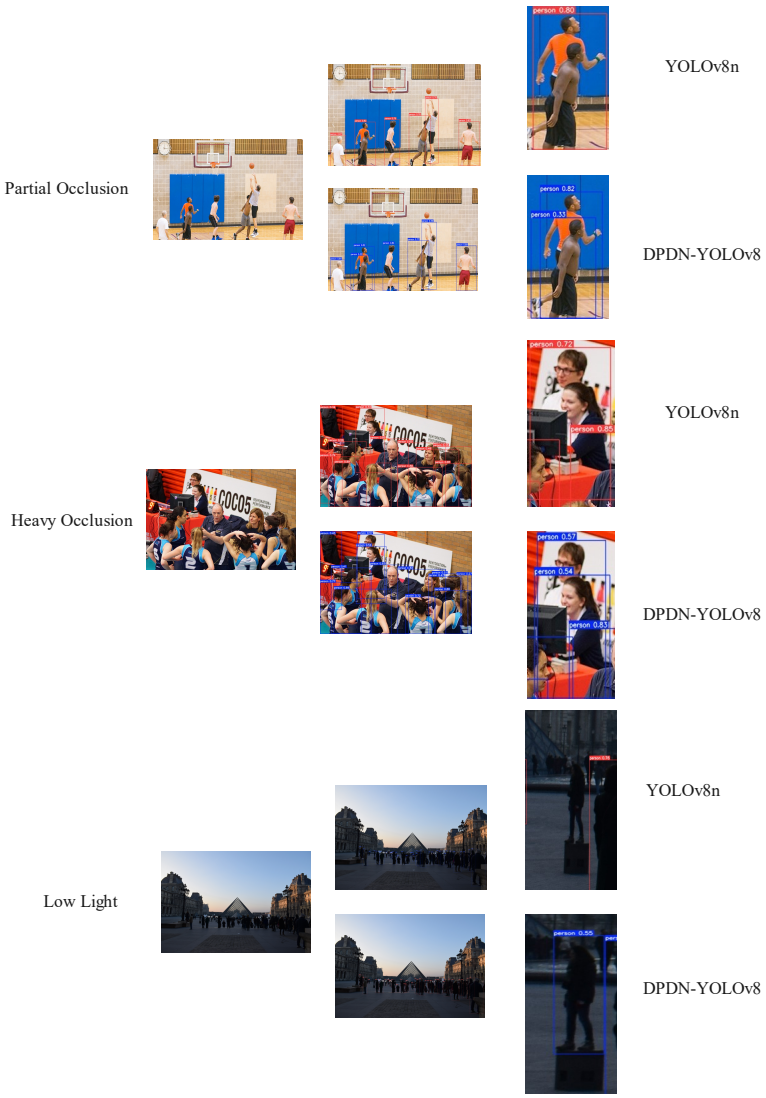

Figure 10. Test results of YOLOv8 and DPDN-YOLOv8.

Figure 11 presents a visual comparison between the DPDN-YOLOv8 model and the original dataset annotations across various challenging scenarios, including motion scenes, urban streets, crowded indoor environments and outdoor pedestrian flows. Green bounding boxes indicate correctly detected targets, blue boxes represent missed detections and red boxes denote false positives. It can be observed that DPDN-YOLOv8 effectively detects the majority of pedestrian targets in complex scenarios involving partial occlusion, severe

crowding and complex lighting conditions. This demonstrates the enhanced robustness of the improved model in accurately identifying and localizing pedestrian targets under various challenging conditions.

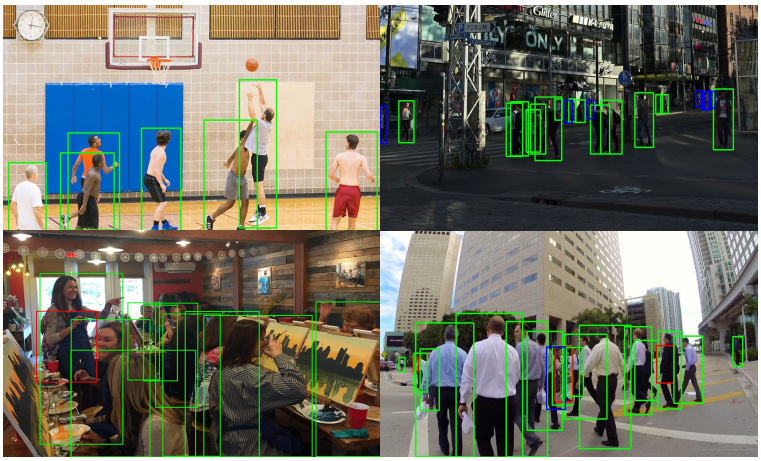

Figure 11. Comparison of detection results with ground truth annotations.

5. Conclusions

To address dense pedestrian detection tasks from a robotic perspective, a novel DPDN-YOLOv8 network model is proposed in this paper, which has four key innovations. First, for enhanced feature extraction, the last three C2f modules in the backbone are replaced with C2f_ODConv modules; these leverage omni-dimensional dynamic convolution to substantially boost the model's multidimensional perception of complex pedestrian features in crowded scenarios. Second, optimized up-sampling is achieved by substituting conventional up-sampling with CARAFE operators, which use semantic-guided feature reorganization strategies to effectively reduce information loss during resolution enhancement. Third, adaptive multi-scale fusion is introduced via a quadruple-head ASFF (Adaptive Spatial Feature Fusion) detection head, which resolves feature inconsistency across scales through dynamic weight learning and thus improves small-target detection accuracy. Fourth, an advanced loss function—the Focaler-Shape-IoU loss—is implemented, integrating dynamic sample focusing with shape-matching optimization to refine bounding box regression precision under heavy occlusion. Extensive validation on the CrowdHuman dataset demonstrates the model's effectiveness: precision of 86.1%, recall of 74.4%, mAP@0.5 of 85.6% and mAP@0.5:0.95 of 55.1%. However, two limitations remain: occasional missed detections in extreme-density scenarios (for which further optimized strategies are recommended) and an increased parameter count (5.2×10^6), which necessitates developing improved feature fusion strategies and lightweight network architectures—priorities for future research in our laboratory.

Author Contributions: Conceptualization, Y.L. and L.X.; methodology, Y.L., L.X. and B.L.; software, L.X., B.L. and D.Y.; validation, Y.L., L.X. and Z.L.; formal analysis, Y.L.; investigation, L.X.; resources, Y.L. and L.X.; data curation, Y.L., L.X. and Z.L.; writing—original draft preparation, Y.L. and L.X.; writing—review and editing, B.L. and D.Y.; visualization, B.L., Z.L. and D.Y.; supervision, Y.L. and Z.L.; project administration, Y.L.; funding acquisition, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: Enterprise-University Cooperation Project (20250041), (20250039).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DPDN-YOLOv8	Dense pedestrian detection network architecture based on YOLOv8n
ODConv	Omni-dimensional dynamic convolution
CARAFE	Content-Aware Reassembly of Features
ASFF	Adaptive Spatial Feature Fusion
YOLO	You Only Look Once

References

1. Patalas-Maliszewska, J.; Dudek, A.; Pajak, G.; Pajak, I. Working toward solving safety issues in human–robot collaboration: A case study for recognising collisions using machine learning algorithms. *Electronics* **2024**, *13*, 731. [CrossRef]
2. Nimma, D.; Al-Omari, O.; Pradhan, R.; Ulmas, Z.; Krishna, R.V.V.; El-Ebiary, T.Y.A.B.; Rao, V.S. Object detection in real-time video surveillance using attention based transformer-YOLOv8 model. *Alex. Eng. J.* **2025**, *118*, 482–495.
3. Tang, J.; Ye, C.; Zhou, X.; Xu, L. YOLO-Fusion and Internet of Things: Advancing object detection in smart transportation. *Alex. Eng. J.* **2024**, *107*, 1–12. [CrossRef]
4. Bobyr, M.; Milostnaya, N.; Khrapova, N. Approach to Detecting Pedestrian Movement Using the Method of Histograms of Oriented Gradients. *Autom. Doc. Math. Linguist.* **2024**, *58* (Suppl. S4), S169–S176.
5. Li, S.; Wang, Q.; Li, R.; Gao, J. Lightweight Neural Network Model and Algorithm for Pedestrian Detection. *SAE Int. J. Connect. Autom. Veh.* **2024**, *8*, 411–424. [CrossRef]
6. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
7. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems, Proceedings of the Conference on Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015*; NeurIPS Foundation: La Jolla, CA, USA, 2015; Volume 28.
8. 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, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
9. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
10. Liu, L.; Song, X.; Song, H.; Sun, S.; Han, X.; Akhtar, N.; Mian, A. Yolo-3DMM for simultaneous multiple object detection and tracking in traffic scenarios. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 9467–9481. [CrossRef]
11. Cheng, T.; Song, L.; Ge, Y.; Liu, W.; Wang, X.; Shan, Y. YOLO-World: Real-Time Open-Vocabulary Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 16–22 June 2024; pp. 16901–16911.
12. Fu, H.; Wang, S.; Duan, P.; Gao, C.; Dian, R.; Li, S.; Li, Z. Lraf-net: Long-range attention fusion network for visible–infrared object detection. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *35*, 13232–13245.
13. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.; Berg, A. Ssd: Single shot multibox detector. In Proceedings of the 14th European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016; Part I 14; Springer International Publishing: Cham, Switzerland, 2016; pp. 21–37.
14. Krizhevsky, A.; Sutskever, I.; Hinton, G. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems, Proceedings of the Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012*; NeurIPS Foundation: La Jolla, CA, USA, 2012; Volume 25.
15. Sun, J.; Wang, Z. Vehicle and pedestrian detection algorithm based on improved YOLOv5. *IAENG Int. J. Comput. Sci.* **2023**, *50*, 1401–1409.
16. Oussouaddi, M.; Bouazizi, O.; Ismaili, Z.; Attaoui, Y.; Chentouf, M. DSR-YOLO: A lightweight and efficient YOLOv8 model for enhanced pedestrian detection. *Cogn. Robot.* **2025**, *5*, 152–165. [CrossRef]
17. Fang, Y.; Pang, H. An improved pedestrian detection model based on YOLOv8 for dense scenes. *Symmetry* **2024**, *16*, 716. [CrossRef]

18. Dou, H.; Chen, S.; Xu, F.; Liu, Y.; Zhao, H. Analysis of vehicle and pedestrian detection effects of improved YOLOv8 model in drone-assisted urban traffic monitoring system. *PLoS ONE* **2025**, *20*, e0314817.
19. Peng, H.; Chen, S. FedsNet: The Real-Time Network for Pedestrian Detection Based on RT-DETR. *J. Real-Time Image Process.* **2024**, *21*, 142.
20. Liu, M.; Jiang, J.; Zhu, C.; Yin, X. Vlpd: Context-aware pedestrian detection via vision-language semantic self-supervision. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 6662–6671.
21. Li, G.; Luo, H.; Huang, H.; Yu, J.; Huang, C.; Xu, X.; Cai, J. RMTP-YOLO: An improved dense pedestrian detection algorithm based on YOLOv8. *J. Electron. Imaging* **2025**, *34*, 013037. [CrossRef]
22. Ni, J.; Shen, K.; Chen, Y.; Yang, S. An improved ssd-like deep network-based object detection method for indoor scenes. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 1–15. [CrossRef]
23. Liu, Q.; Li, Z.; Zhang, L.; Deng, J. MSCD-YOLO: A Lightweight Dense Pedestrian Detection Model with Finer-Grained Feature Information Interaction. *Sensors* **2025**, *25*, 438. [CrossRef]
24. Liu, Q.; Ye, H.; Wang, S.; Xu, Z. YOLOv8-CB: Dense pedestrian detection algorithm based on in-vehicle camera. *Electronics* **2024**, *13*, 236.
25. Terven, J.; Córdova-Esparza, D.; Romero-González, J. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. *Mach. Learn. Knowl. Extr.* **2023**, *5*, 1680–1716. [CrossRef]
26. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
27. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Lake City, UT, USA, 18–22 June 2018; pp. 8759–8768.
28. Feng, C.; Zhong, Y.; Gao, Y.; Scott, M.; Huang, W. Tood: Task-aligned one-stage object detection. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; IEEE Computer Society: Alamitos, CA, USA, 2019; pp. 3490–3499.
29. Huang, C.; Cui, J.; Li, Y.; Lu, Y.; Yang, C. Fusion YOLOv8s and Dynamic Convolution Algorithm for Steel Surface Defect Detection. *Symmetry* **2025**, *17*, 701. [CrossRef]
30. Wang, J.; Chen, K.; Xu, R.; Liu, Z.; Loy, C.; Lin, D. Carafe: Content-aware reassembly of features. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3007–3016.
31. Xu, Y.; Pan, H.; Wang, L.; Zou, R. MC-ASFF-ShipYOLO: Improved Algorithm for Small-Target and Multi-Scale Ship Detection for Synthetic Aperture Radar (SAR) Images. *Sensors* **2025**, *25*, 2940.
32. 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.* **2021**, *52*, 8574–8586. [CrossRef] [PubMed]
33. Cui, Y.; Ren, W.; Cao, X.; Knoll, A. Focal network for image restoration. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 13001–13011.
34. Wang, Z.; Zhang, S.; Chen, Y.; Xia, Y.; Wang, H.; Jin, R.; Wang, C.; Fan, Z.; Wang, Y.; Wang, B. Detection of small foreign objects in Pu-erh sun-dried green tea: An enhanced YOLOv8 neural network model based on deep learning. *Food Control* **2025**, *168*, 110890. [CrossRef]
35. Li, Z.; Luo, N.; Zhang, X.; Guo, Z.; Fang, X.; Qiao, Y. Crowdassign: A Label Assignment Scheme for Pedestrian Detection in Crowded Scenes. In Proceedings of the 2024 IEEE International Conference on Image Processing (ICIP), Abu Dhabi, United Arab Emirates, 27–30 October 2024; pp. 326–331.
36. Zhang, Y.; Ren, W.; Zhang, Z.; Jia, Z.; Wang, L.; Tan, T. Focal and efficient IOU loss for accurate bounding box regression. *Neurocomputing* **2022**, *506*, 146–157. [CrossRef]
37. Ren, F.; Fei, J.; Li, H.; Doma, B. Steel surface defect detection using improved deep learning algorithm: ECA-SimSPPF-SIoU-Yolov5. *IEEE Access* **2024**, *12*, 32545–32553. [CrossRef]
38. Rezatofighi, H.; Tsoi, N.; Gwak, J.; 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, Long Beach, CA, USA, 16–20 June 2019; pp. 658–666.
39. Geng, X.; Su, Y.; Cao, X.; Li, H.; Liu, L. YOLOFM: An improved fire and smoke object detection algorithm based on YOLOv5n. *Sci. Rep.* **2024**, *14*, 4543. [CrossRef]
40. Liu, Y.; Shen, S. Vehicle detection and tracking based on improved YOLOv8. *IEEE Access* **2025**, *13*, 24793–24803. [CrossRef]
41. Lin, T.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
42. Thuan, D. Evolution of Yolo Algorithm and Yolov5: The State-of-the-Art Object Detention Algorithm. Bachelor's Thesis, Oulu University of Applied Science, Oulu, Finland, 2021.

43. Wang, C.; Bochkovskiy, A.; Liao, H. 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, Vancouver, BC, Canada, 18–22 June 2023; pp. 7464–7475.
44. Mao, M.; Lee, A.; Hong, M. Efficient Fabric Classification and Object Detection Using YOLOv10. *Electronics* **2024**, *13*, 3840. [CrossRef]
45. Wang, Q.; Wang, Q. BT-YOLO11: Automatic Driving Road Target Detection in Complex Scenarios. *IEEE Access* **2025**, *13*, 72364–72374. [CrossRef]