IOP Publishing Measurement Science and Technology
Meas. Sci. Technol. **35** (2024) 016001 (21pp) https://doi.org/10.1088/1361-6501/ad042a

Waste-YOLO: towards high accuracy real-time abnormal waste detection in waste-to-energy power plant for production safety

He Wang[2], Lianhong Wang[1,*], Hua Chen[3], Xiaoyao Li[4], Xiaogang Zhang[1] and Yicong Zhou[5]

[1] College of Electrical and Information Engineering, Hunan University, Changsha 410082, People's Republic of China
[2] Nanning Electric Power Supply Bureau, Guangxi Power Grid Co., Ltd, Nanning 530029, People's Republic of China
[3] College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, People's Republic of China
[4] College of Computer and Information Engineering, Central South University of Forestry and Technology, Changsha 410004, People's Republic of China
[5] Department of Computer and Information Science, University of Macau, Macau 999078, People's Republic of China

E-mail: wanglh@hnu.edu.cn

Received 11 July 2023, revised 4 October 2023
Accepted for publication 17 October 2023
Published 26 October 2023

Abstract

Due to the danger of explosive, oversize and poison-induced abnormal waste and the complex conditions in waste-to-energy power plants (WtEPPs), the manual inspection and existing waste detection algorithms are incapable to meet the requirement of both high accuracy and efficiency. To address the issues, we propose the Waste-YOLO framework by introducing the coordinate attention, convolutional block attention module, content-aware reassembly of features, improved bidirectional feature pyramid network and SCYLLA- intersection over union loss function based on YOLOv5s for high accuracy real-time abnormal waste detection. Through video acquisition, frame-splitting, manual annotation and data augmentation, we develop an abnormal waste image dataset with the four most common types (i.e. gas cans, mattresses, wood and iron sheets) to evaluate the proposed Waste-YOLO. Extensive experimental results demonstrate the superiority of Waste-YOLO to several state-of-the-art algorithms in waste detection effectiveness and efficiency to ensure production safety in WtEPPs.

Keywords: deep learning, object detection, waste-to-energy power plants, abnormal waste detection, YOLOv5, production safety

1. Introduction

Waste-to-energy power plants (WtEPPs) have become a primary method of waste disposal that addresses waste accumulation while generating heat and electricity [1]. In China, due to confusion in waste classification and negligence by waste transfer station workers, some wastes affecting production safety are often transported to WtEPPs along with domestic waste by closed garbage trucks. The wastes are dumped into the garbage dumps at the discharge gate, left to ferment and awaiting incineration. Some of the unsafe

* Author to whom any correspondence should be addressed.

wastes (e.g. gas cans) are explosive, some (e.g. mattresses and wood) are too large and tend to block the incinerator opening, and some (e.g. iron sheets) may melt and bind the grate at high temperatures, causing mechanical failure of the incinerator and even the production of poisonous substances [2]. This paper refers to all unsafe wastes as abnormal waste. Due to the serious impact of abnormal waste on production safety at WtEPPs, it is crucial to detect and remove them as soon as possible. Most WtEPPs in China use manual identification of abnormal waste. Employees need to constantly watch for the abnormal waste and control the waste crane to pick them out. Since the abnormal waste is heavy and often buried by other lighter waste, it is more suitable to detect during the falling process rather than in garbage dumps. However, due to the high similarity of garbage objects, fast falling speed and long sight distance, it is difficult for employees to identify abnormal waste with the naked eye. Prolonged gazing can also cause eye fatigue. As a result, the manual inspection often leads to a high false detection rate and missed rate of abnormal waste and cannot locate abnormal waste in time.

To address the issues of high cost, low efficiency and low accuracy of manual inspection, waste detection techniques have been widely applied in urban and natural scenarios. These techniques primarily include traditional methods and deep learning-based methods. Traditional methods employ classical image classification algorithms [3–6] to manually extract features at the geometric, edge and texture levels, and then classify waste by classifiers [7]. However, these methods are characterized by a complex process, low efficiency and accuracy. The superior performance of deep learning has led to its widespread adoption in computer vision [8, 9]. Convolutional neural networks (CNNs) have been widely applied for waste detection. Existing object detection algorithms can be classified into two categories: two-stage and single-stage. The former first generates candidate regions and then performs classification and regression on them. The latter directly regresses the bounding boxes and class labels. Among the two-stage algorithms, Faster region-based convolutional neural networks (R-CNN) [10] is a representative one that combines region proposal network [10] and Fast R-CNN [11]. Nowakowski and Pamuła [12] used Faster R-CNN for the e-waste detection with an average accuracy of over 90%. Faster R-CNN achieves high accuracy but suffers from low speed and poor performance on small objects. To address these issues, single-stage algorithms such as single shot multibox detector (SSD) [13] and you only look once: unified, real-time object detection (YOLO) [14–19] have been developed with higher speed but lower accuracy than two-stage algorithms. YOLOv5 [18] is the most popular one due to its excellent overall performance. Recently, some new variants of YOLO such as YOLOv6 [20], YOLOv7 [21], DAMO-YOLO [22] and YOLOv8 [23] have been proposed and improved the detection accuracy on the COCO [24] dataset. However, their performance on waste detection has not been validated. Patel *et al* [25] compared the waste detection capability of several detectors in street scenes and found that YOLOv5 outperformed the others. Mao *et al* [26] designed a YOLOv3-based detector on a Taiwanese recycled waste dataset with high accuracy. Li *et al* [27] proposed a multimodel cascaded CNN based on SSD, YOLOv4 [17] and Faster R-CNN to reduce false positive predictions of domestic waste. However, these detectors are mostly evaluated on datasets with simple backgrounds and single objects with fixed shapes. In contrast, the waste detection in WtEPPs faces more challenges due to the complex backgrounds, overlapping objects and varying shapes and poses of the waste in the falling process. These issues may lead to the incapability of existing detectors in waste detection in WtEPPs. Moreover, some of these detectors have a large number of parameters and thus may not ensure real-time performance.

To meet the need for data-driven evaluation benchmarks for deep detection networks, several studies have focused on image dataset construction for waste recognition. TrashNet [28] is the first public waste dataset and contains images taken at Stanford University and residential areas with simple white background. The corresponding image augmentation was achieved by techniques such as random rotation and brightness adjustments. GINI [29] dataset contains 2561 waste images obtained through a combination of Bing search engine queries and other sources. TACO [30] is an open image dataset for waste classification, detection and segmentation tasks with images captured in outdoor scenes. Panwar *et al* [31] proposed the AquaTrash dataset, which consists of 369 manually annotated images covering four types of waste. Although these datasets provide a sample base and evaluation benchmark for the development of waste detection, there are still very few available waste image datasets because of the limitations of relatively small size, specific scenes and non-open source. Therefore, these datasets cannot be applied to the task of abnormal waste detection in this paper due to the differences in waste categories and scenes.

Since there are few studies on the abnormal waste detection, we propose a high accuracy real-time abnormal waste detection algorithm for WtEPPs, named Waste-YOLO. Additionally, we analyzed and collected four types of abnormal waste (i.e. mattresses, gas cans, wood and iron sheets) with the highest frequency and quantity according to the requirements of a WtEPP in Changsha, China, from July 2020 to July 2022. We captured the abnormal waste images during falling in the dumping pool to construct a dataset for neural network training and testing. Our main contributions are summarized as follows:

(1) An abnormal waste image dataset is constructed with 6283 images, including the four most common types of abnormal waste, i.e. mattresses, gas cans, wood, and iron sheets.
(2) Different from the typical modules in YOLOv5s, we introduce the coordinate attention (CA) and convolutional block attention module (CBAM) to extract the key features and suppress the irrelevant information, and the content-aware reassembly of features (CARAFE) to enhance the image definition of feature maps after upsampling and retain more details. Besides, we also introduce and improve the bidirectional feature pyramid network (BiFPN) to enhance deep feature fusion. To further

enhance object localization accuracy and speed up bounding box regression convergence, SCYLLA-intersection over union (SIoU) is applied as the bounding box regression loss function.

(3) Based on the improved modules, we propose a novel YOLOv5s-based framework named Waste-YOLO for high accuracy real-time abnormal waste detection.

(4) We provide extensive experiments to evaluate our method. Experimental results show that the proposed Waste-YOLO framework achieves state-of-the-art detection performance in both effectiveness and efficiency on the proposed abnormal waste dataset and several public datasets.

The rest of the paper is organized as follows. Section 2 will describe the difficulties in abnormal waste image detection. Section 3 will briefly review the YOLOv5s network. Section 4 will introduce the Waste-YOLO framework based on YOLOv5s. Section 5 will present the experimental settings and results. Section 6 will summarize the paper.

2. Difficulties in abnormal waste image detection

In the early stages of unloading waste, abnormal waste may be hidden in the garbage heap and obscured by other objects. The conditions for capturing images are complex and various unrelated domestic garbage can interfere with detecting specific targets. This means that the detection model needs to focus on learning the features of the object to be inspected while ignoring irrelevant features as much as possible. Detection must occur during the dynamic process of waste dumping and falling, where the shape and size of waste constantly change. This requires that the detection model have good feature extraction and fusion capabilities. However, due to limitations in camera hardware and installation position, images may be blurred and targets may occupy a relatively small portion of images. The environmental brightness at the waste dump is also rather dark, which adds further difficulty to the detection task.

Owing to the intricate environmental variables present in waste disposal sites and the distinct morphological properties of waste, difficulties arise in the detection of abnormal waste imagery. Figure 1 illustrates the difficulties of our detection task. The images have intricate backgrounds and contain numerous small targets such as gas cans. Accurate detection of these small targets within a complex background and dynamic falling process is essential. Upon zooming in, figure 1(a) reveals apparent noise related to the camera's focus, lens distance, and lens surface contamination. Figure 1(b) shows motion blur during the fall of the red mattress, resulting in indistinct texture at its edges. In figure 1(c), the top of the wood is obscured by the waste truck's tailgate and overlaps with surrounding garbage. The iron sheet in figure 1(d) appears as a horizontal strip with severe deformation during its fall, losing its sheet-like characteristics. Note that images in figure 1 are acquired by our designed abnormal waste dataset, which is introduced in section 4. Initially, we selected the YOLOv5s algorithm for study in this paper. Our experimental results indicate that there is potential for improvement in both the mAP and prediction box localization. To address this, we have improved YOLOv5s by focusing on the four difficulties mentioned above. Subsequently, the baseline algorithm YOLOv5s will be introduced.

3. YOLOv5s network

YOLOv5 has gained widespread use in both industry and academia due to its lightweight design, high speed and accuracy. There are four versions of YOLOv5 [8]: YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. The smallest version YOLOv5s has the least network depth and width with a parameter size of only 7.2 MB. The other three models are deeper and wider than YOLOv5s and the largest model YOLOv5x has a parameter size of 86.7 MB. Due to the real-time requirements for abnormal waste detection in actual scenes, we choose YOLOv5s as our baseline network for further development.

Figure 2 illustrates the structure of YOLOv5s comprising four components: input, backbone, neck, and prediction. The input component includes mosaic data enhancement, automatic anchor box calculation and adaptive image scaling. The backbone network consists of focus downsampling, convolution + batch normalization + sigmoid linear unit (CBS), cross-stage partial (C3) module and spatial pyramid pooling (SPP). Firstly, the focus module takes each 640 $\times$ 640 RGB image to generate a 320 $\times$ 320 $\times$ 12 feature map through two times downsampling, channel concatenation and convolution operation with a 3 $\times$ 3 kernel. This module retains the most information while using reshape operations to reduce floating point operations (FLOPs) caused by convolution and accelerate the network. Next, CBS and C3 blocks are stacked to extract low-level visual features with downsampling used to generate feature maps at different scales. CBS comprises convolutional layers followed by batch normalization and sigmoid linear unit activation functions to extract image features through convolutional operations. Based on ResNet's residual connection [32], C3 consists of three CBS modules and shortcut branches, performing better in feature extraction than traditional convolutional layers. Finally, SPP uses a convolutional module to halve the number of feature channels before applying maximum pooling with three different kernels to expand the receptive field at a relatively low cost. In the neck component, C3 further extracts high-level semantic features before combining feature pyramid network (FPN) [33] and path aggregation network (PANet) [34] for feature fusion through upsampling and concatenation transmitting features at different scales. In the prediction component, three detection heads with size 80 $\times$ 80, 40 $\times$ 40, and 20 $\times$ 20 are generated using binary cross-entropy loss with a sigmoid layer (BCEWithLogitsLoss) as the loss function for confidence and classification, generalized intersection over union (GIoU) [35] as the loss function for bounding box regression; and non-maximum suppression (NMS) algorithm to select the best prediction box.

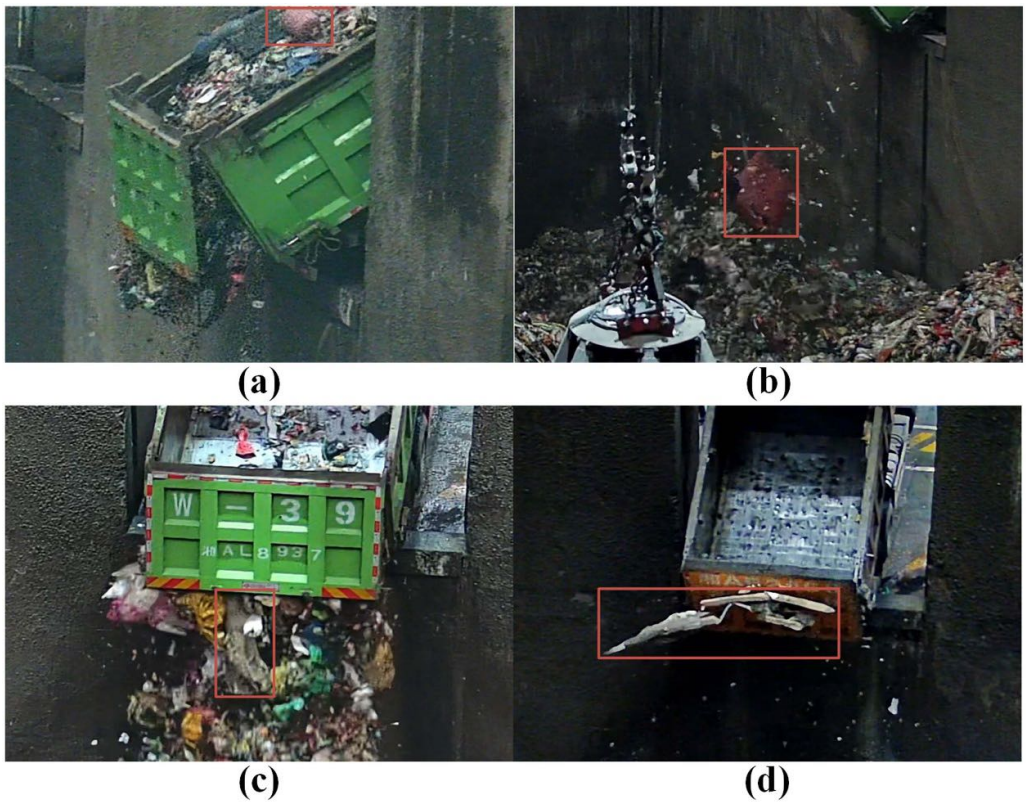


Figure 1. Difficulties in our detection task. The red rectangles represent detection targets. (a) Image noise. (b) Motion blur. (c) Occlusion and overlap. (d) Severe deformation.

4. Waste-YOLO framework

To address the difficulties in abnormal waste sorting for WtEPPs, we introduce an enhanced model structure of YOLOv5s, named Waste-YOLO. Figure 3 illustrates its network structure, with detailed descriptions provided below. The red rectangles represent our improvements compared to YOLOv5s-v5.0.

4.1. C3 with attention mechanisms

4.1.1. C3CA. While most image classification networks utilize convolutional layers to extract image features, these layers primarily analyze local relationships and often fail to establish long-distance dependencies or focus on the most relevant targets. This becomes particularly challenging when low-level convolutional layers concentrate on features with local deviation, making it difficult for high-level convolutional layers to capture effective features representing local information through continuous downsampling and network deepening. To mitigate this issue, we introduce the C3CA module, which embeds a CA [36] block into the C3 module.

As depicted in figure 4, the execution of CA consists of two stages: coordinate information embedding and CA generation [36]. Initially, the feature map undergoes coordinate information embedding. An image X of size $C \times H \times W$ is inputted, and average pooling is employed to encode each channel separately according to horizontal and vertical coordinate directions. Consequently, the output of the C_{th} channel with height h is given by equation (1):

$$z_c^h(h) = \frac{1}{W} \sum_{0 \leqslant i < W} x_c(h, i). \tag{1}$$

Here, $x_c(h, i)$ denotes the component with coordinates (h, i) and channel c in the input feature map, z^h is the output after average pooling along the X-axis, and $z_c^h(h)$ signifies the component in the C_{th} channel with height h. The output expression of the C_{th} channel with width w is illustrated in equation (2):

$$z_c^w(w) = \frac{1}{H} \sum_{0 \leqslant j < H} x_c(j, w). \tag{2}$$

In this equation, $x_c(j, w)$ denotes the component with coordinates (j, w) and channel c in the input feature map, z^w is the output after average pooling along the Y-axis, and $z_c^w(w)$ signifies the component in the C_{th} channel with width w. As demonstrated in equation (3), the feature maps obtained from the two pooling operations are concatenated and subsequently transformed using a shared 1×1 convolution F_1 to reduce the channel dimension. The intermediate result m, obtained by passing the output through a nonlinear activation layer $\delta(\bullet)$ using the h-swish function [37], is then produced

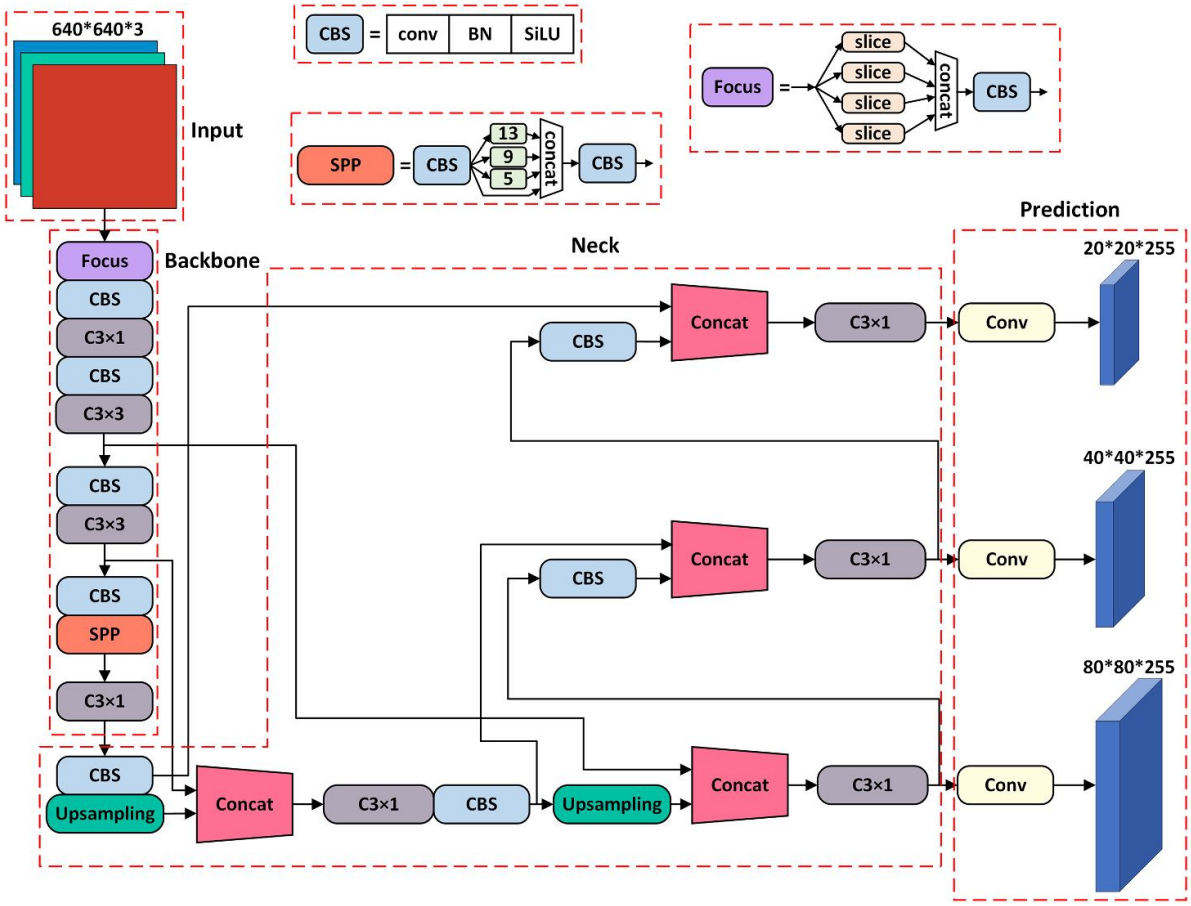


Figure 2. The architecture of the YOLOv5s-v5.0 method.

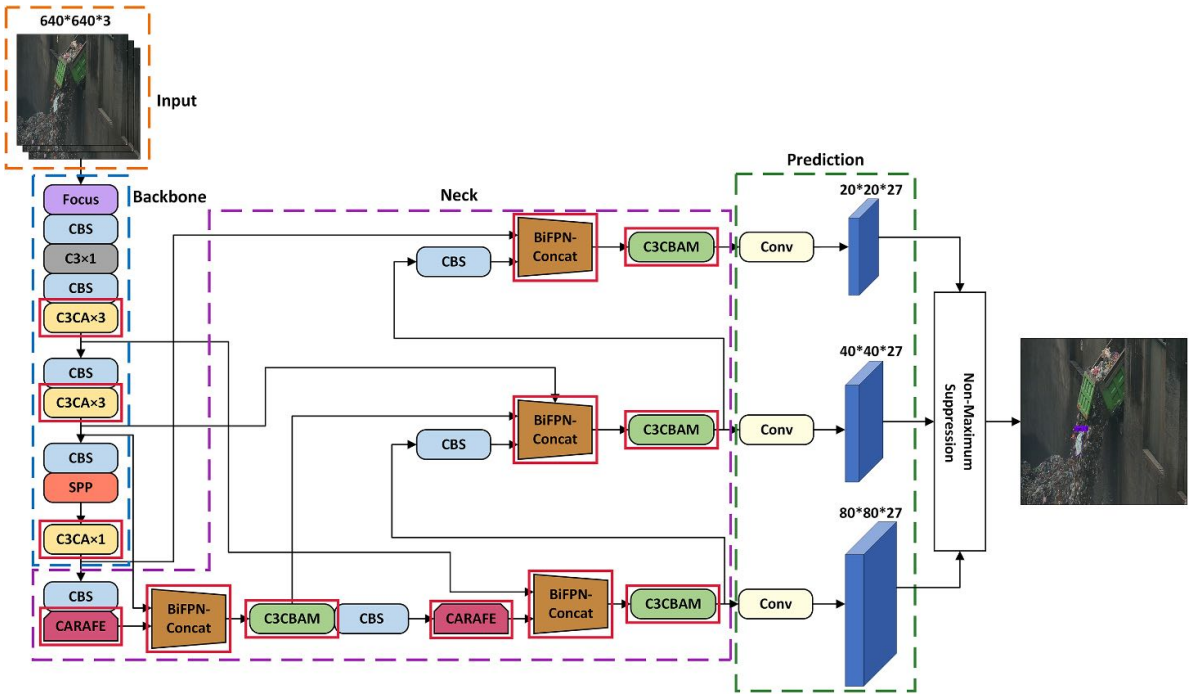


Figure 3. Waste-YOLO network architecture.

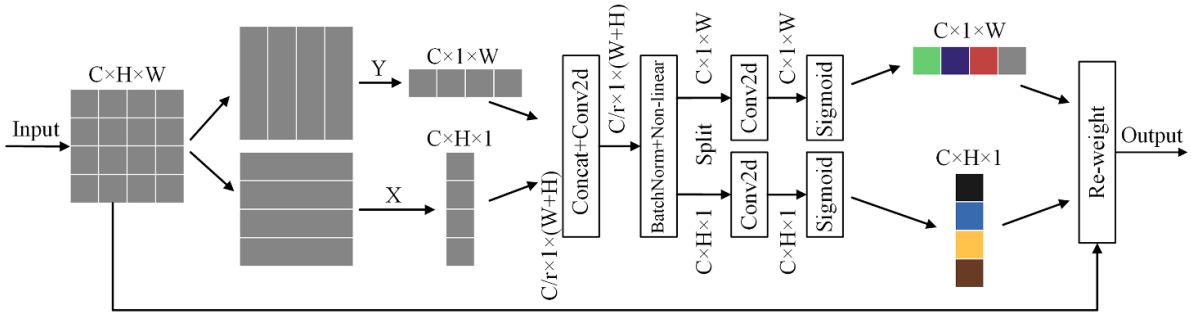


Figure 4. The calculation process of the CA mechanism.

$$m = \delta\left(F_1\left(\left[z^h, z^w\right]\right)\right). \tag{3}$$

Subsequently, the intermediate result m is divided into two independent tensors $\mathbf{m}^h$ and $\mathbf{m}^w$ along the spatial dimension. The feature maps $\mathbf{m}^h$ and $\mathbf{m}^w$ are then transformed to match the number of channels as the input X using two separate 1×1 convolutions F_h and F_w. The sigmoid activation function $\sigma(\cdot)$ is applied to yield the results shown in equations (4) and (5):

$$g^h = \sigma\left(F_h\left(\mathbf{m}^h\right)\right), \tag{4}$$

$$g^w = \sigma\left(F_w\left(\mathbf{m}^w\right)\right). \tag{5}$$

In equations (4) and (5), g^h and g^w denote the attention weight features for the X and Y coordinates, respectively. The final output of the CA module is given by equation (6):

$$y_c(i,j) = x_c(i,j) \times g_c^h(i) \times g_c^w(j). \tag{6}$$

In this equation, $x_c(i,j)$ and $y_c(i,j)$ represent the values of the input and output feature maps at coordinates (i,j) and channel c, respectively.

The CA block in the C3CA module encodes each channel of the input feature map along with its horizontal and vertical coordinates, and then aggregates these features to generate a set of direction-aware attention maps. This allows C3CA to accurately locate objects of interest by establishing long-distance dependencies along one spatial direction while preserving localization information along the other. Furthermore, C3CA can efficiently capture relationships among channels. Given that its input and output feature maps are the same size, C3CA can be seamlessly integrated into any network.

As depicted in figure 5, the structure of C3CA includes a Resunit block, which represents stackable residual blocks. In the C3CA module, the input feature map is processed through two branches: one with a CBS block and a Resunit block for feature extraction, and the other with a CBS block for residual connection. After channel concatenation, a CBS block and a CA block are applied to produce the output results. Given that the CA block encodes each channel of the input feature map in conjunction with its horizontal and vertical coordinates, and subsequently aggregates these features to produce a set of direction-aware attention maps, C3CA can precisely locate objects of interest by establishing long-distance dependencies along one spatial direction while preserving localization information along the other. Moreover, C3CA can effectively capture relationships among channels. As its input and output feature maps are of identical size, C3CA can be seamlessly integrated into any network. To address limitations in feature extraction using convolution operations, we incorporate C3CA into the backbone part of YOLOv5s to capture global information from abnormal waste images, thereby focusing on targets and significantly minimizing interference from irrelevant background information. Furthermore, the introduction of CA during the downsampling of feature maps amplifies useful information while suppressing noise.

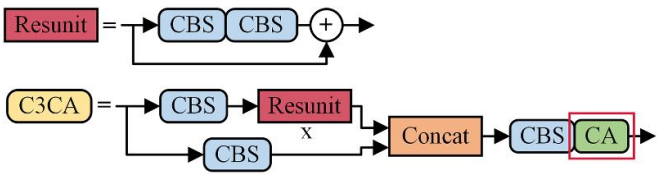


Figure 5. Structure of C3CA module.

4.1.2. C3CBAM. Before we delve into the C3CBAM module, let us briefly review the CBAM [38], a lightweight attention mechanism that can be integrated into CNNs with minimal computational overhead. CBAM comprises a channel attention module and a spatial attention module. The channel attention module assigns different weights to different channels in the feature map, distinguishing those that significantly contribute to the task and making the feature map more effective for subsequent network layers. This process is summarized by equation (7):

$$\begin{aligned} M_c(F) &= \sigma\left(\mathrm{MLP}\left(\mathrm{AvgPool}(F)\right) + \mathrm{MLP}\left(\mathrm{MaxPool}(F)\right)\right) \\ &= \sigma\left(W_1\left(W_0\left(F_{\mathrm{avg}}^c\right)\right) + W_1\left(W_0\left(F_{\mathrm{max}}^c\right)\right)\right), \end{aligned} \tag{7}$$

where F denotes the input feature map, while F_{avg}^c and F_{max}^c represent the feature maps post average pooling and max pooling, respectively. W_0 and W_1 symbolize the two-layer parameters in the multilayer perceptron (MLP) model. The neurons in this two-layer neural network utilize the rectified rinear unit activation function. $\sigma(\cdot)$ is the sigmoid function. During computation, both F_{avg}^c and F_{max}^c share the two-layer parameters W_0 and W_1 in the MLP model.

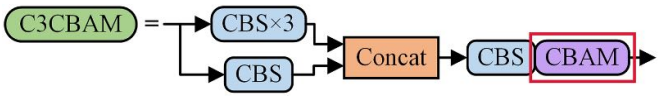


Figure 6. Structure of C3CBAM module.

The spatial attention module focuses on location and pixel information, helping the model identify the target region for learning. The channel attention module first applies global maximum pooling and global average pooling to produce two vectors. These vectors are then processed by an MLP layer and a sigmoid activation function to generate the channel attention map. Subsequently, the spatial attention module uses the channel-refined features as input and applies global maximum pooling, global average pooling, dimension raising and reduction, and a sigmoid function to obtain the spatial feature map. This process is shown in equation (8):

$$\begin{aligned} M_s(F') &= \sigma\left(f^{7\times7}\left(\left[\mathrm{AvgPool}(F');\mathrm{MaxPool}(F')\right]\right)\right) \\ &= \sigma\left(f^{7\times7}\left(\left[F^s_{\mathrm{avg}};F^s_{\mathrm{max}}\right]\right)\right), \end{aligned} \tag{8}$$

where F' denotes the feature map computed by the channel attention module, while F^s_{avg} and F^s_{max} represent the feature maps post average pooling and max pooling, respectively. $f^{7\times7}$ symbolizes a convolution operation with a kernel size of 7×7. $\sigma(\cdot)$ is the sigmoid function. The channel attention map and the spatial feature map are amalgamated to produce a mixed-domain feature map.

In the Waste-YOLO model, the backbone part is responsible for primary feature extraction, and since the neck part does not deepen the network, residual connections in C3 blocks within the neck part are unnecessary. The model performs five downsampling operations in the backbone part and two upsampling and two downsampling operations in the neck part, leading to a constantly changing feature map size. However, small targets in an image can cause spatial feature loss, and as channel numbers increase with network depth, channel information may also be lost. Post BiFPN [39] feature fusion, the data becomes more abstract and if feature loss is not suppressed in time, accuracy may decrease. To mitigate this issue, we introduce CBAM into C3 to create the C3CBAM module (as shown in figure 6), which is placed after BiFPN–Concat layers. This allows CBAM to enhance focus on important channel and spatial features, suppress irrelevant regions, and reduce image feature loss.

4.2. Improvement of upsampling layer

Most object detection algorithms employ nearest-neighbor interpolation for upsampling. Despite its high speed and low computational overhead, this method often leads to significant feature loss and reduced detection accuracy for small targets. Given that our dataset comprises blurry images and small targets, it is crucial to preserve more features during the upsampling process. To tackle this challenge, Waste-YOLO incorporates CARAFE [40] to enhance image definition post-upsampling. As depicted in figure 7, the structure of CARAFE primarily consists of two components: the kernel prediction module and the content-aware reassembly module.

In the kernel prediction module, the channel of an $H\times W\times C$ input feature map is initially compressed from C to C_m using a 1×1 convolution. Assuming that the upsampling size is $K_{\mathrm{up}}\times K_{\mathrm{up}}$, and a unique upsampling kernel is utilized for each position of the output feature map, the size of the predicted upsampling kernel is $\sigma H\times\sigma W\times K_{\mathrm{up}}\times K_{\mathrm{up}}$. For the compressed input feature map from the initial step, an upsampling kernel is predicted via a $K_{\mathrm{encoder}}\times K_{\mathrm{encoder}}$ convolutional layer with an input channel of C_m and an output channel of $\sigma^2K^2_{\mathrm{up}}$. The channel dimension is subsequently expanded in the spatial dimension to yield an upsampling kernel with a shape of $\sigma H\times\sigma W\times K^2_{\mathrm{up}}$. Finally, this kernel is normalized by a softmax function such that the weighted sum of its elements equals 1.

In the content-aware reassembly module, each position in the output feature map is mapped back to its corresponding position in the input feature map. A $K_{\mathrm{up}}\times K_{\mathrm{up}}$ area centered on this position is extracted and dot-multiplied with its corresponding predicted upsampling kernel to finalize upsampling and generate an output feature map with a shape of $\sigma H\times\sigma W\times C$.

Compared to nearest neighbor and bilinear interpolation methods, CARAFE significantly enhances performance on various tasks with minimal computational overhead [40].

4.3. Improved feature fusion network

In CNN models, feature maps in shallow layers undergo fewer downsampling operations compared to those in deep layers. Consequently, shallow layer feature maps possess higher resolution and retain richer texture information. Conversely, deep-layer feature maps have lower resolution due to more frequent downsampling but encapsulate richer semantic information. However, due to variations in camera focal length and shooting angle, the scale of abnormal waste can vary significantly. Single-layer CNN feature representation and the translation invariance of CNN can cause the model to lose position information. As a result, multi-scale feature representation and fusion are indispensable.

Feature fusion networks have witnessed significant development in recent years. To tackle the limitation of one-way feature flow in FPN's [33] top-down path, PANet [34] introduces an additional bottom-up path. Building on this, BiFPN [39] employs bidirectional cross-scale connections while simplifying the network by eliminating nodes with only one input edge.

Therefore, we propose an enhanced BiFPN network as the feature fusion network in Waste-YOLO, replacing the PANet in YOLOv5. Figure 8(b) illustrates this enhanced BiFPN network, and figure 8(a) displays the PANet connections used in YOLOv5s for comparison purposes. Feature fusion is expressed using the equation: $feature=[f_1;f_2;f_3]$, where ';' denotes splicing by channel dimension and f_1,f_2,f_3 symbolize three feature maps within a feature fusion network. The res-

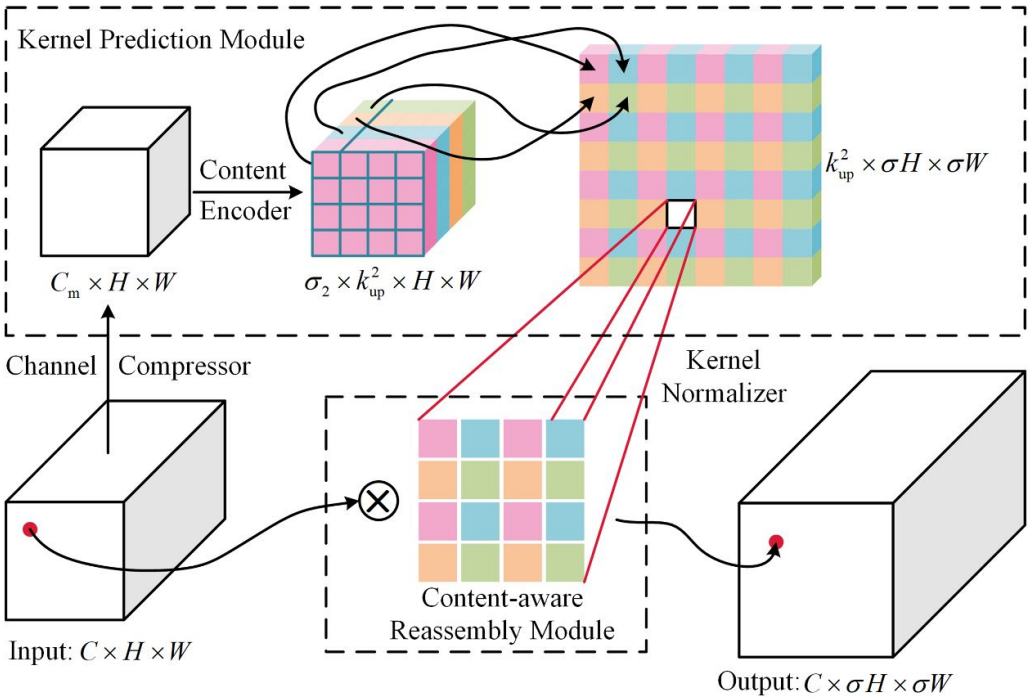


Figure 7. Structure of CARAFE.

ulting feature map used for subsequent transfer post-fusion is denoted by *feature*. In figure 8, C_i, F_i, and P_i signify feature maps generated at different stages of the network. P_4 is produced by concatenating downsampled features of P_3, C_4, and F_4 along the channel dimension. This process is illustrated in equation (9), where we represent the C3CBAM operator as a function F and Downsample refers to double downsampling achieved through convolution operations. Similarly, the generation process P_3 is illustrated in equation (10):

$$P_4 = F([\text{Downsample}(P_3); F_4; C_4]), \tag{9}$$

$$P_3 = F(F_3; C_3). \tag{10}$$

Unlike the BiFPN in EfficientDet [39], Waste-YOLO employs C3CA and CARAFE for feature processing instead of traditional convolutional layers. We have designed a novel Concat layer, christened as BiFPN–Concat, by integrating the characteristics and advantages of the original BiFPN. BiFPN–Concat is applied within Waste-YOLO, replacing the four Concat layers in YOLOv5s. Unlike Concat, which directly concatenates feature channels, BiFPN accomplishes feature fusion by adding weighted features to the feature channels. If the size and channel number of the input feature map and output feature map are identical, BiFPN–Concat can fuse the features of both, sharing a convolution kernel. As depicted in the neck section in figure 3, the second BiFPN–Concat from the top has an additional edge connected to backbone's C3CA, achieving the fusion of shallow and deep features. We introduce learnable weights for each input of BiFPN–Concat to ascertain the significance of different input features and employ the stochastic gradient descent (SGD) optimization method for backpropagation to update weights during network training. This is a capability that the Concat layer lacks. Compared to Concat, BiFPN–Concat not only reduces computational overhead and supports feature fusion of a larger number of feature maps, but also distinguishes the importance of different input features. Therefore, introducing BiFPN–Concat enhances cross-scale feature fusion and enriches both positional and semantic information across different scales.

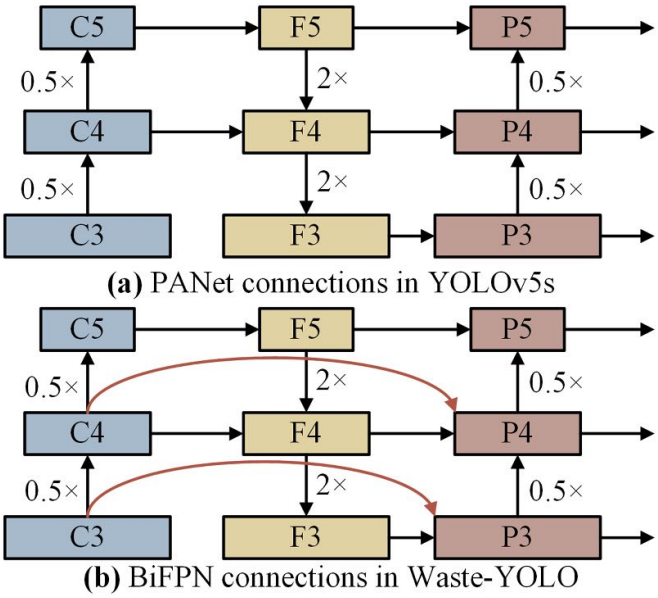


Figure 8. Feature fusion network of YOLOv5s and Waste-YOLO.

4.4. Loss function

The loss function of Waste-YOLO is composed of three components: confidence loss (l_{conf}), category loss (l_{cls}), and bounding box regression loss (l_{box}). The corresponding equations are as follows:

$$Loss = l_{\text{conf}} + l_{\text{cls}} + l_{\text{box}}, \tag{11}$$

$$l_{\text{conf}} = \lambda_{\text{obj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{obj}} \left[-\hat{C}_i \ln C_i - \left(1 - \hat{C}_i\right) \ln\left(1 - C_i\right) \right] + \lambda_{\text{nobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{nobj}} \left[-\hat{C}_i \ln C_i - \left(1 - \hat{C}_i\right) \ln\left(1 - C_i\right) \right], \tag{12}$$

$$l_{\text{cls}} = \sum_{i=0}^{S^2} \sum_{j=0}^{B} \sum_{c \in \text{cls}} I_{ij}^{\text{obj}} \left[-\hat{p}_i(c) \ln\left(p_i(c)\right) - \left(1 - \hat{p}_i(c)\right) \ln\left(1 - p_i(c)\right) \right]. \tag{13}$$

In equations (11)–(13), s^2 signifies the number of grid cells, and B denotes the number of predicted bounding boxes within each grid cell. The binary variables $I_{ij}^{\text{obj}} = 1$ and $I_{ij}^{\text{nobj}} = 1$ indicate whether the jth bounding box in the i th grid cell is responsible for predicting an object or not, respectively. λ_{obj} and λ_{nobj} are the corresponding weight coefficients. C_i and $\widehat{C}_i$ represent the confidence scores of the predicted and actual targets, respectively. c represents the predicted category of the bounding box. $p_i(c)$ and $\widehat{p}_i(c)$ denote the predicted and actual probabilities that a detected object belongs to category c, respectively.

YOLOv5s employs the GIoU loss function as its bounding box regression loss function. This function measures the deviation in position between the target and predicted boxes, as expressed in equation (14):

$$L_{GIoU} = 1 - \text{IoU} + \frac{C - (A \cup B)}{C}. \tag{14}$$

In equation (14), A represents the target box, and B denotes the predicted box. IoU is the ratio of the intersection area to the union area of these two boxes. C signifies the area of the smallest enclosing region that contains both boxes. According to equation (14), when the predicted box and target box do not overlap, the intersection over union (IoU) value is always zero. However, gradient regression can still be performed. When two boxes have overlapping regions, their IoU value may be identical but may not accurately reflect the degree of overlap between them. GIoU considers not only overlapping areas but also non-overlapping areas, providing a more accurate representation of the degree of coincidence between two boxes. This addresses having identical values but different regression effects for the predicted box. However, when a target box completely encloses a predicted box, GIoU cannot distinguish their relative positions and cannot provide an effective optimization direction.

To address these issues, a more advanced loss function called complete-IoU (CIoU) [41] has been proposed. Compared to GIoU, CIoU introduces the aspect ratio of a bounding box and performs regression based on overlapping area, center point distance, and aspect ratio between a predicted box and a target box.

However, CIoU does not consider mismatch angles between a target box and a predicted box. As a result, we introduce SIoU loss [42] as the l_{box} of Waste-YOLO. SIoU loss comprises angle cost (Λ), distance cost (Δ), shape cost (Ω), and IoU cost. The formulas for SIoU loss are defined below. Unlike CIoU, SIoU incorporates angular calculations between a target and predicted boxes, which accelerates network convergence.

$$L_{SIoU} = 1 - \text{IoU} + \frac{\Delta + \Omega}{2}, \tag{15}$$

$$\Lambda = 1 - 2 \cdot \sin^2\left(\arcsin(x) - \frac{\pi}{4}\right), \tag{16}$$

$$\Delta = \sum_{t=x,y} \left(1 - \mathrm{e}^{-(2-\Lambda)\rho_t}\right), \tag{17}$$

$$\Omega = \sum_{t=w,h} \left(1 - \mathrm{e}^{-\omega_t}\right)^{\theta}. \tag{18}$$

5. Experiments and discussions

5.1. Experimental settings

5.1.1. Evaluation indicators. We evaluate the performance of our model using both qualitative and quantitative methods. Qualitative evaluation involves assessing the visible results produced by the model based on images. For quantitative evaluation, we use metrics such as mean average precision (mAP), mAP(0.5:0.95), parameter size, Giga Floating Point Operations (GFLOPs), and graphic processing unit (GPU) inference time. Parameter size and GFLOPs are calculated using the thop and torchsummary Python packages. mAP represents the average precision (AP) when the IoU threshold is 0.5 and reflects the model's recognition ability. mAP(0.5:0.95) represents the average mAP value when the IoU threshold varies from 0.5 to 0.95 in increments of 0.05 and is used to evaluate localization performance and bounding box regression capability. The calculation of mAP involves precision and recall values. The GPU inference time reflects the speed at which our model can process a single frame, encompassing pre-processing time, inference time, and NMS processing time. Parameter size and GFLOPs serve as indicators of model complexity and are defined as follows:

$$\text{Precision} = \frac{TP}{TP + FP}, \tag{19}$$

$$\text{Recall} = \frac{TP}{TP + FN}, \tag{20}$$

$$\mathrm{AP} = \int_0^1 P(R)\,\mathrm{d}R, \tag{21}$$

$$\mathrm{mAP} = \frac{\sum_{i=1}^{N} AP_i}{N}. \tag{22}$$

In equations (19)–(22), *TP* denotes the number of predicted bounding boxes with an IoU value greater than or equal to the preset threshold. *FP* signifies the number of predicted bounding boxes with an IoU value less than the preset threshold. *FN* represents the number of missed targets. *P* and *R* symbolize precision and recall, respectively. *N* indicates the number of categories.

5.1.2. The establishment of abnormal waste image dataset

5.1.2.1. Dataset collection. The abnormal waste image dataset was collected and organized from a WtEPP located in Changsha, China between July 2020 and July 2022. With significant practical implications, the dataset can provide data support and industry references for relevant enterprises to identify abnormal waste and promote the digital construction of intelligent factories. Four Huawei M6721-E-Z31 spherical network cameras are installed inside the garbage yard as part of an abnormal waste detection system to record the dumping waste from an unloading door. Two of them are set on the wall opposite the garbage discharge door to monitor ten discharge doors and the other two are installed on the side of the yard. The camera frame rate is set to 25 frames per second with a shutter speed of 1/50 s, allowing for real-time capture of the entire dynamic process of abnormal waste dumping at a high frame rate. The collected video is in mp4 format with a frame width of 1920, frame height of 1080 and frame rate of 60 frames per second to capture the movement of abnormal waste. Videos containing abnormal waste are manually screened and intercepted before being processed into individual images through post-framing processing. The dataset contains four categories: mattress, gas can, wood and iron sheet. The initial dataset includes 1898 images which are divided into training (1328 images with 1469 labels) and test (570 images with 630 labels) sets using a ratio of 7:3.

5.1.2.2. Dataset labeling. Different deep neural networks require input data in specific formats. For object detection tasks, the PASCAL visual object classes (VOC) data format is commonly used for manual labeling of datasets because it is widely recognized and easily convertible to other formats. In this study, we use LabelImg software to manually label abnormal waste images in the PASCAL VOC format to meet the requirements of various models for comparative experiments. The label box parameters in the PASCAL VOC format include $x_{\min}$, $y_{\min}$, $x_{\max}$, and $y_{\max}$, representing the x- and y-coordinates of the upper left and lower right vertices of the target. However, YOLOv5 and Waste-YOLO do not support input parameters in this format and require conversion to YOLO format. The label box parameters in YOLO format include x_{center}, y_{center}, w, and h, representing normalized values for the x- and y-coordinates of the target center point location and the width and height of the label box, respectively. The conversion process from PASCAL VOC to YOLO format is defined by equations (23)–(28), where *width* and *height* denote the width and height of the image, respectively

$$x'_{\text{center}} = \frac{x_{\max} - x_{\min}}{2} + x_{\min}, \tag{23}$$

$$y'_{\text{center}} = \frac{y_{\max} - y_{\min}}{2} + y_{\min}, \tag{24}$$

$$x_{\text{center}} = \frac{x'_{\text{center}}}{width}, \tag{25}$$

$$y_{\text{center}} = \frac{y'_{\text{center}}}{height}, \tag{26}$$

$$w = \frac{x_{\max} - x_{\min}}{width}, \tag{27}$$

$$h = \frac{y_{\max} - y_{\min}}{height}. \tag{28}$$

Table 1. The number of abnormal waste image dataset training set labels before and after data augmentation. Original is the original training set. DataAug is the training set after data augmentation.

Training set	Mattress	Gas can	Wood	Iron sheet	Total
Original	547	398	406	118	1469
DataAug	2113	2255	2164	354	6886

Table 2. The number of abnormal waste image dataset test set labels.

	Mattress	Gas can	Wood	Iron sheet	Total
Test set	239	184	166	41	630

5.1.2.3. Dataset augmentation. To improve model generalization ability, we use four data augmentation methods as below to expand the number of training images to 5713 (6886 labels). Table 1 shows the number of labels in the training set before and after data augmentation. Table 2 presents the number of labels in the test set.

- *Random channel swap*: the RGB channels of the image are randomly swapped according to a transformation matrix with a probability of 1.
- *Random horizontal flip*: the image is randomly flipped horizontally with a probability of 1.
- *Random rotation*: The image is rotated randomly by a degree chosen uniformly from the interval (10, 90).
- *Random crop*: the image is randomly cropped according to the size of its bounding boxes with a probability of 1.

Table 3. Experimental environment configuration.

Hardware and software	Name
Operating system	Windows 10 64 bit
Central processing unit	Intel Core i9-10900 K 3.70 GHz
Graphic processing unit	NVIDIA GeForce RTX 3090
Random access memory	32.0 GB
Programming language	Python 3.8.12
Compiling software	PyCharm 2021.1.1
Deep learning dependency	Pytorch 1.9.0 + torchvision 0.10.0 + cudatoolkit 11.1.1

Table 4. The primary hyperparameter values of Waste-YOLO.

Hyperparameter	Value
Initial learning rate (*lr0*)	0.01
Final OneCycleLR learning rate (*lrf*)	0.01
SGD optimizer momentum	0.937
Optimizer weight_decay	0.0005
Warmup epochs	3.0
Warmup initial momentum	0.8
Warmup initial bias *lr*	0.1
IoU threshold during training	0.20
Box loss gain	0.05
Cls loss gain	0.5
Obj loss gain	1.0

5.1.3. Parameter settings

5.1.3.1. Environment configurations and hyperparameter settings. The software and hardware configurations for all experiments are shown in table 3. All experiments use the same environment configuration and dataset. During training, the input image size is 640 × 640, the batch size is 16 and the number of epochs is 300. The SGD optimizer is used to iteratively update the network parameters. YOLOv5s serves as the baseline model. Waste-YOLO follows the hyperparameters and values established by YOLOv5. The primary hyperparameter values are shown in table 4.

5.1.3.2. Learning rate adjustment. We use the warmup method to preheat and periodically adjust the learning rate. At this stage, the learning rate for each iteration is updated until 0.1 using one-dimensional linear interpolation. After that, the learning rate is updated by the cosine annealing algorithm, and finally drops to 0.0001 ($lr0 \times lrf$). The change in learning rate during training is shown in figure 9.

5.2. Experimental results on the abnormal waste image dataset

5.2.1. Data enhancement experiments. To evaluate the effectiveness of the data augmentation operation described in this paper, we train YOLOv5s using both the original and data-augmented abnormal waste image dataset training sets and then validate their performance on the abnormal waste image dataset test set. The corresponding results are presented in figure 10 and table 5.

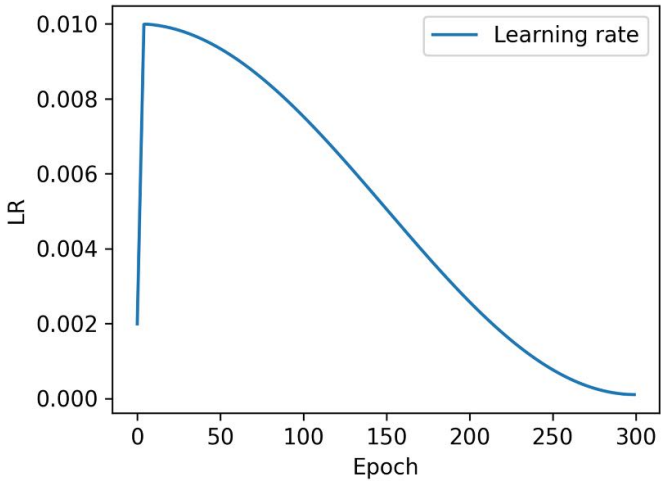


Figure 9. Change in learning rate during training.

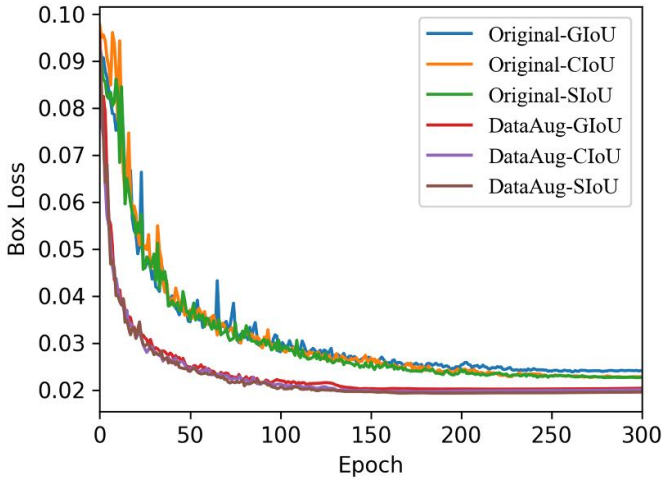


Figure 10. Variation curves for box loss values across different bounding loss functions in data enhancement experiments.

As presented in table 5, we conducted experiments using the baseline model YOLOv5s with three different loss functions: SIoU loss, GIoU loss (the default l_{box} in YOLOv5) and CIoU loss. The average accuracy by three loss functions on the data-augmented abnormal waste image dataset was significantly higher than that on the original abnormal waste image dataset. The mAP and mAP(0.5:0.95) increase by 1.6% and 5.7%, respectively, for both GIoU loss and CIoU loss, and by 1.4% and 5.6%, respectively, for SIoU loss. This fully demonstrates the effectiveness of the data enhancement methods used in this paper. Since SIoU considers the vector angle between the target box and the prediction box, it achieved the highest average accuracy in our experiment. On the data-augmented abnormal waste image dataset, SIoU loss surpasses GIoU loss and CIoU loss by 0.5% and 0.2% in mAP, respectively, and by 0.7% and 0.3% in mAP(0.5:0.95), respectively. In terms of inference speed, SIoU loss was slightly faster than both GIoU loss and CIoU loss. These results indicate that SIoU loss

Table 5. Data enhancement experiments result on the abnormal waste image dataset test set using YOLOv5s. Original and DataAug, represent the original training set and the training set after data augmentation, respectively.

Model	Dataset	Bounding box regression loss	mAP(0.5)	mAP(0.5:0.95)	Speed-GPU (ms)	Params (10^6)	FLOPs (G)
YOLOv5s	Original	GIoU	92.1%	68.9%	14.6	7.072	16.4
		CIoU	92.4%	69.3%	14.4		
		SIoU	92.8%	69.7%	14.3		
	DataAug	GIoU	93.7%	74.6%	14.5		
		CIoU	94.0%	75.0%	14.4		
		SIoU	94.2%	75.3%	14.3		

Table 6. Ablation experiments result on the abnormal waste image dataset test set. '+' represents adding modules based on YOLOv5s. Data for the proposed Waste-YOLO model is bolded.

Model	mAP(0.5)	mAP(0.5:0.95)	Params (10^6)	FLOPs (G)	Speed-GPU (ms)
YOLOv5s	94.2%	75.3%	7.072	16.4	14.3
+C3CBAM	95.2%	75.7%	7.090	16.4	17.4
+C3CA	94.7%	75.9%	7.107	16.5	15.5
+CARAFE	95.6%	77.4%	7.205	16.9	15.1
+BiFPN	94.2%	74.3%	8.139	17.8	14.7
+C3CA + BiFPN	95.0%	76.1%	8.175	17.8	16.1
+C3CBAM + CARAFE	95.7%	77.4%	7.224	16.9	16.6
+C3CA + C3CBAM + BiFPN	95.3%	76.6%	8.194	17.9	17.2
Waste-YOLO	**96.2%**	**77.4%**	**8.328**	**18.4**	**18.1**

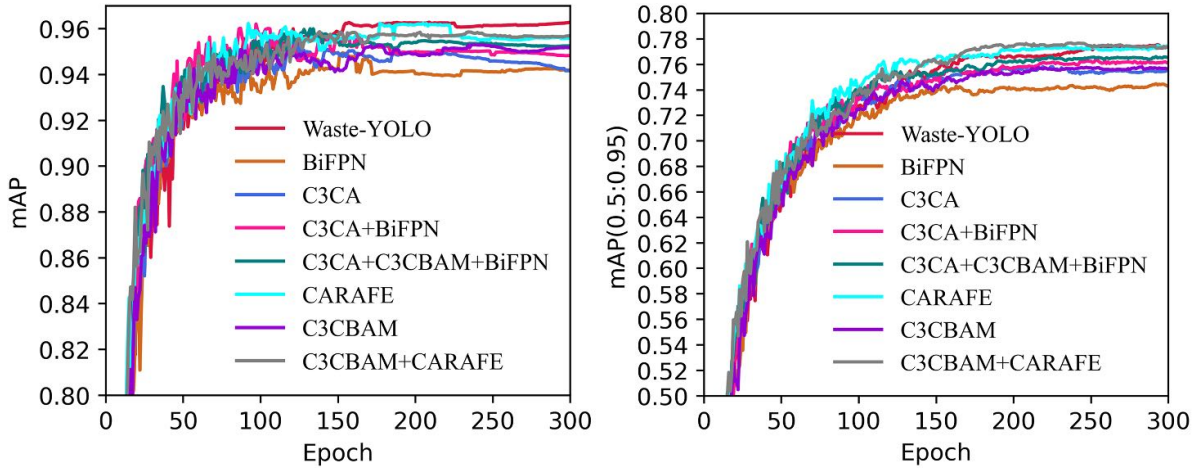
has better bounding box regression capability than both GIoU loss and CIoU loss while having minimal impact on inference speed when used with the YOLOv5s model.

Figure 10 illustrates the training progress of six models listed in table 5 by displaying their bounding box loss curves. As can be seen from figure 10, the models trained on both datasets eventually converge. For the models trained on the original abnormal waste image dataset, the final loss value was around 0.026. For the models trained on the data-augmented abnormal waste image dataset, the final loss value was reduced to 0.02. Thus, a larger amount of data can help YOLOv5s fit better during training. Moreover, with lower bounding box regression loss and less oscillation during training, SIoU loss outperforms GIoU loss and CIoU loss.

Overall, these results demonstrate the effectiveness of data augmentation and SIoU loss's superior localization accuracy. Thus, we choose SIoU as the l_{box} of Waste-YOLO and data-augmented abnormal waste image dataset in subsequent experiments.

5.2.2. Ablation experiments. To test the effectiveness of each proposed module, we conduct the ablation experiments presented in table 6, figures 11–13.

In table 6, YOLOv5s + C3CBAM and YOLOv5s + C3CA increase the mAP of YOLOv5s by 1.0% and 0.5%, respectively, and the mAP(0.5:0.95) by 0.4% and 0.6%, respectively. This demonstrates that the attention mechanism can help the model focus on practical feature information and enhance CNN's feature extraction ability. YOLOv5s + CARAFE surpasses YOLOv5s by 1.4% in mAP and 2.1% in mAP(0.5:0.95), proving that CARAFE can enhance the definition of upsampling feature maps compared to nearest neighbor interpolation, thereby significantly improving model accuracy. The YOLOv5s + BiFPN decreases the mAP(0.5:0.95) of YOLOv5s by 1.0%. We believe this is because YOLOv5s' backbone network is not processed, resulting in some irrelevant features being redundant and key features being further lost after neck feature fusion, affecting BiFPN weight training effectiveness. However, when BiFPN is used alone with CA and CBAM, both mAP@0.5 and mAP(0.5:0.95) are improved. This shows that CA enhances the extraction of critical information during feature extraction and CBAM further filters out redundant and irrelevant information in channels and spatialities during feature fusion. Ultimately, the embedding of CA and CBAM enables BiFPN to optimize its neurons' weights and biases to better values using the SGD optimizer during the back-propagation phase of training. The proposed Waste-YOLO network performs the best by exceeding YOLOv5s by 2.0% in mAP and 2.1% in mAP(0.5:0.95). In terms of the number of parameters, the introduction of BiFPN increases far more than the introduction of CBAM, CA, and CARAFE. Overall, Waste-YOLO has 17.76% more parameters and 12.19% more GFLOPs than YOLOv5s with an inference speed only 3.8 ms slower. In summary, although the proposed Waste-YOLO slightly increases complexity and inference time, it significantly improves mAP. Figure 11 shows the change curve of mAP values with epoch during the training process of each model in table 6. The results demonstrate that Waste-YOLO outperforms YOLOv5s in detection accuracy, exhibiting the highest mAP and mAP(0.5:0.95). Waste-YOLO improves mAP and mAP(0.5:0.95) via the positive aggregation and reorganization of channels and



(a) The change curve of mAP **(b)** The change curve of mAP(0.5:0.95)

Figure 11. Results of ablation experiments for each model on the testing set.

spatial features facilitated by C3CA and C3CBAM, the enhancement of feature map resolution and receptive field brought about by CARAFE, and the balance of semantic and spatial information of features maintained by BiFPN. Additionally, the SIoU loss function contributes to a lower prediction box localization error. In summary, Waste-YOLO focuses more on abnormal waste features than the competing methods.

Figure 12 depicts a visual comparison of detection effects between Waste-YOLO and YOLOv5s. We randomly choose one image from each of the four categories in the abnormal waste test set for detection. Waste-YOLO can accurately detect all targets with higher accuracy than YOLOv5s, indicating that its ability to detect abnormal waste is stronger. As shown in figure 12(d), Waste-YOLO's predicted box is better suited to the target, further demonstrating the superior bounding box regression ability of SIoU compared to GIoU loss.

In figure 13, we employ Grad-CAM [43] to compare the heatmaps of Waste-YOLO and YOLOv5s. The redder regions in the image indicate a higher network response and contribution, signifying the importance of each location for detection. Compared to Waste-YOLO, YOLOv5s disperses more attention on unrelated regions. This demonstrates that CA and CBAM can help Waste-YOLO focus more on critical feature information of actual objects than YOLOv5s, thus improving the detection accuracy.

5.2.3. Algorithm comparisons. In this experiment, we compare Waste-YOLO with 11 popular and different types of object detection algorithms in terms of quantitative and qualitative evaluation. All algorithms are trained using the scratch training strategy (without pre-trained weights) to ensure fairness. The results are presented in table 7 and figure 14.

In table 7, Waste-YOLO performs the best in mAP and second in mAP(0.5:0.95) following YOLOv8n. The single-frame inference speed of Waste-YOLO is fast enough to meet the real-time requirements of the WtEPP where abnormal waste data was collected for this study. Furthermore, its accuracy surpasses other mainstream object detection algorithms from the same period and exceeds that of subsequently released algorithms such as YOLOv7-Tiny, YOLOv6n, and DAMO-YOLO-T.

For visual evaluation, we implement Faster-R-CNN, EfficientDet-D1, YOLOv8n, and our proposed Waste-YOLO (the top four models ranked by mAP in table 7) on two abnormal waste images randomly selected from the test set featured occlusion and severe deformation during falling. As shown in figure 14, Waste-YOLO has the best detection results. It detects an iron sheet and a mattress with the highest probabilities of 0.85 and 0.93, respectively. This superior performance can be attributed to several factors: The introduction of CA and CARAFE maximizes information about target areas and CBAM further filters background information; deep feature fusion by BiFPN allows Waste-YOLO to effectively complete detection on abnormal waste. Although YOLOv8n has the highest mAP(0.5:0.95) in table 7, its detection performance is not as good as Waste-YOLO in terms of visual evaluation. Specifically, YOLOv8n has a lower detection probability of iron sheets and matrices compared to Waste-YOLO. Additionally, during the detection of iron sheet images, YOLOv8n produces overlapping prediction boxes with inaccurate positioning.

5.2.4. Experiments with imbalanced data. Focal loss (FL) [44] was developed to address the issue of imbalanced positive and negative samples and complex samples. Based on FL, quality focal loss (QFL) [45] addresses FL's limitation of only supporting discrete labels and varifocal loss (VFL)

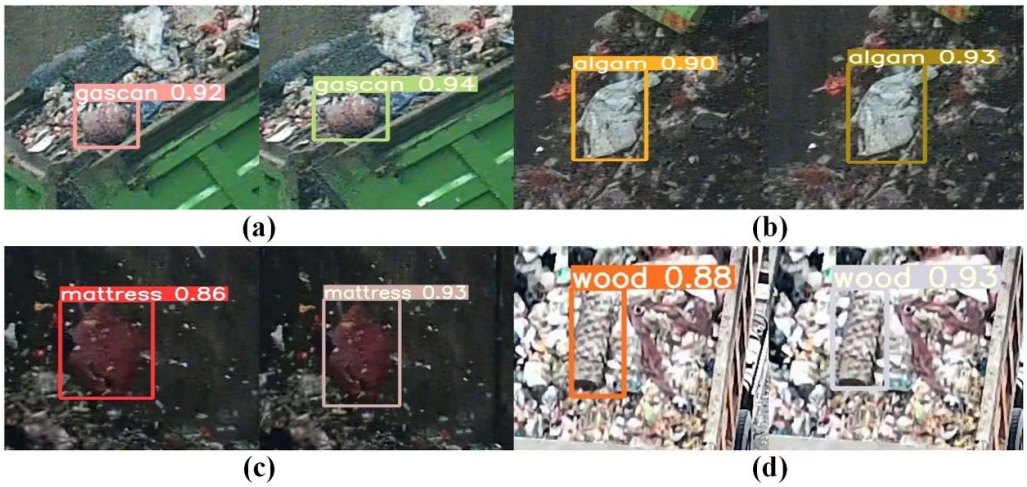


Figure 12. Comparison of detection effects between YOLOv5s (left) and Waste-YOLO (right) for each group of images.

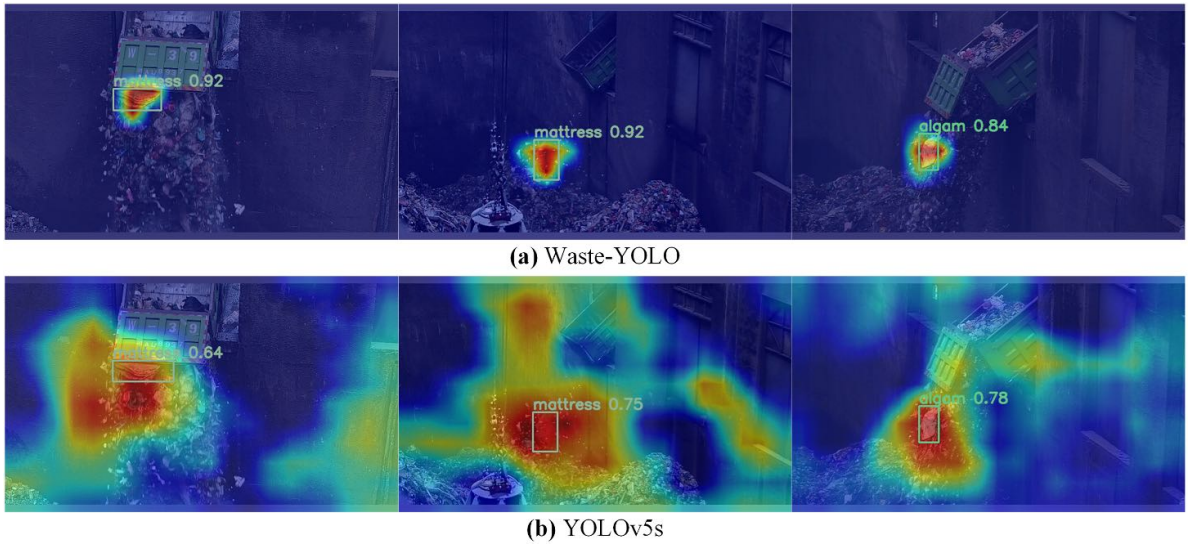
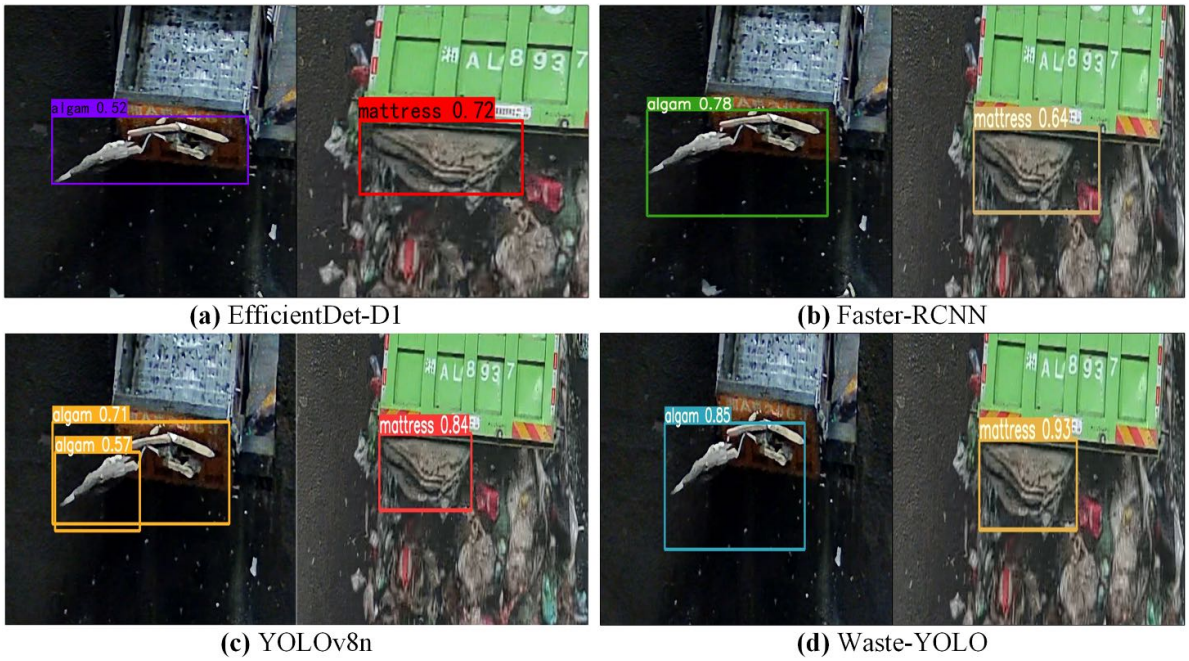


Figure 13. Comparison of heat maps of Waste-YOLO and YOLOv5s.

Table 7. Comparison of different networks' performance on the abnormal waste image dataset test set. Data for the proposed Waste-YOLO model is bolded.

Method	mAP(0.5)	mAP(0.5:0.95)	Params (10^6)	FLOPs (G)	Speed-GPU (ms)
YOLOv8n	95.9%	81%	3.012	8.2	16.5
YOLOv7-Tiny	94.20%	75.2%	6.023	13.2	13.65
YOLOv6n	94.00%	74.3%	4.3	11.1	17.29
DAMO-YOLO-T	92.40%	73.5%	8.57	18.25	20.32
YOLOX-s	94.05%	72.7%	8.939	26.642	29.29
SSD	60.87%	30.2%	3.941	6.042	16.74
YOLOv4-Tiny	84.42%	50.7%	5.881	16.157	18.47
CenterNet	89.42%	61.0%	32.665	109.338	31.72
Faster-R-CNN	95.65%	64.0%	28.306	946.508	61.49
EfficientDet-D1	95.81%	72.5%	6.557	11.205	57.61
YOLOv3-Tiny	88.10%	62.7%	8.677	12.9	9.72
Waste-YOLO	**96.2%**	**77.4%**	**8.328**	**18.4**	**18.1**



(a) EfficientDet-D1 **(b)** Faster-RCNN

(c) YOLOv8n **(d)** Waste-YOLO

Figure 14. Comparison of detection effects of four algorithms ranking the top four in mAP.

Table 8. Results of focal loss family experiment on the abnormal waste image dataset test set.

Model	Classification and confidence loss	mAP(0.5)	mAP(0.5:0.95)	Params (10^6)	FLOPs (G)	Speed-GPU (ms)
Waste-YOLO	FL	93.1%	75.9%	8.328	18.4	18.4
	QFL	93.4%	76.3%			18.5
	VFL	94.3%	76.8%			18.7
	BCEWithLogitsLoss	96.2%	77.4%			18.1

[46] integrates object confidence and localization accuracy into detection scores.

In our dataset, there is an imbalance in the number of labels between iron sheets and the other three types of targets. As a result, we replace the classification loss and confidence loss BCEWithLogitsLoss of Waste-YOLO with one from the FL family to explore whether this could improve model performance. The results of these experiments are presented in table 8 and figure 15.

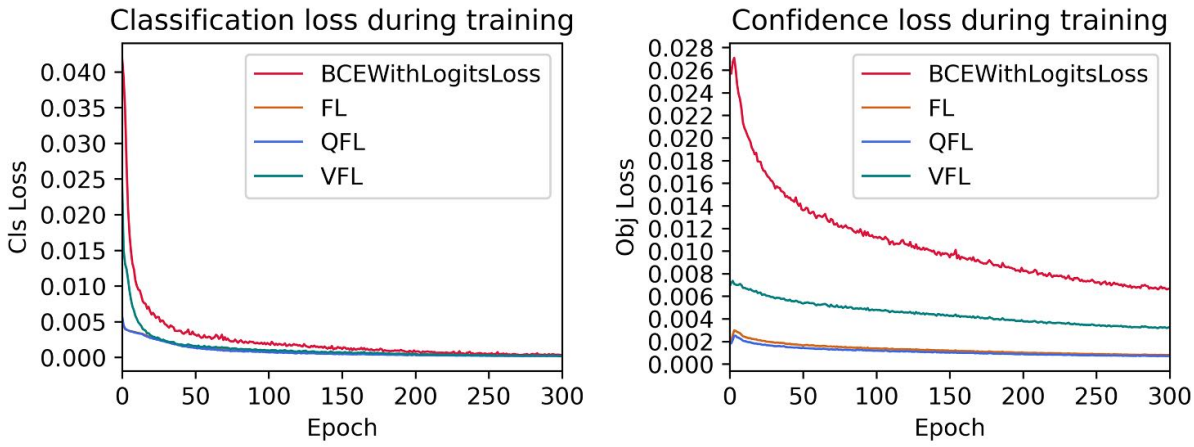
In table 8, with the help of BCEWithLogitsLoss, Waste-YOLO achieves the best results on all the metrics. Figure 15 presents the classification loss and confidence loss for these loss functions during training and validation. It can be seen that the BCEWithLogitsLoss value goes higher than VFL, FL and QFL. Furthermore, FL, QFL and VFL converge at earlier epochs and their curves were smoother than BCEWithLogits. These results demonstrate that while members of the FL family converged faster than BCEWithLogitsLoss their accuracy and speed were not as good. This is because [44–46] were tested on the COCO dataset, which is much larger than our abnormal waste image dataset. It is considered that the small size of our dataset prevented us from fully utilizing the advantages of members of the FL family, which were designed to balance various APs rather than improve the final mAP.

In conclusion, BCEWithLogitsLoss is more suitable for Waste-YOLO to implement detection on the abnormal waste image dataset.

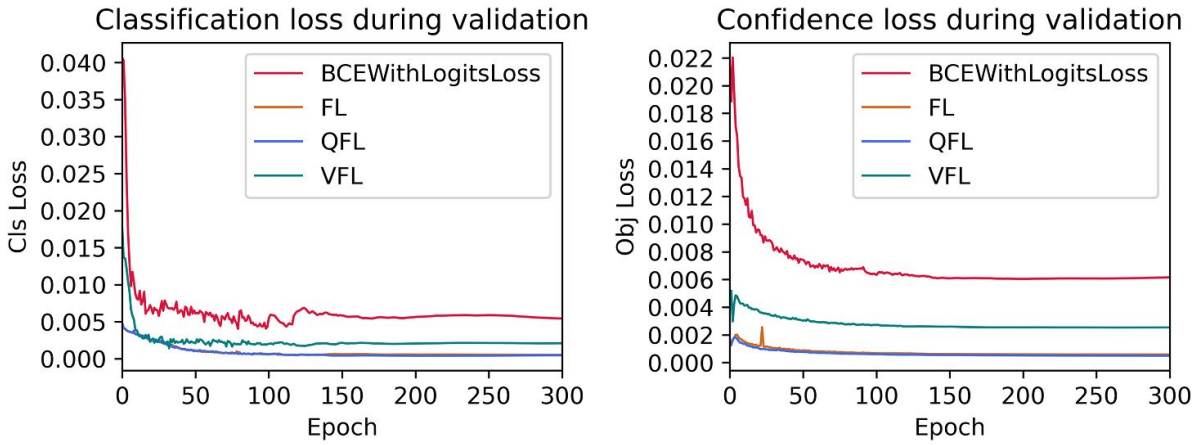
5.3. Experimental results on public datasets

To further verify the universality of Waste-YOLO, we test it on two public datasets: PASCAL VOC [47] and VisDrone2021 [48]. The PASCAL VOC dataset consists of VOC2007 and VOC2012 and contains 16 551 training images and 4952 test images. The VisDrone2021 dataset is used for object detection and tracking in visual data obtained from aerial drones. It includes ten categories with 6471 training images, 548 validation images, and 1610 test images. These images feature many dense and small targets. For our tests, we combined the validation and test sets into one.

Table 9 presents the results of Waste-YOLO compared to 12 other models on the PASCAL VOC dataset and the results compared to YOLOv5s on the VisDrone2021 dataset. In terms of mAP and mAP(0.5:0.95), Waste-YOLO improves upon



(a) Loss change curves during training



(b) Loss change curves during validation

Figure 15. Curves of loss values for different classification and confidence loss functions during training and validation.

YOLOv5s by 2.0% and 3.4%, respectively, on the PASCAL VOC dataset, and by 0.9% and 0.4%, respectively, on the VisDrone2021 dataset, with only a 1.4 ms slower single-frame detection speed. Tables 10 and 11 provide detailed single-class AP comparison results of our method and YOLOv5s for all categories on the PASCAL VOC2007 test set and VisDrone2021 test set, respectively. As can be seen, the AP results of Waste-YOLO are higher than those of YOLOv5s on almost all categories. Therefore, Waste-YOLO outperforms YOLOv5s in terms of quantitative evaluation on the two public datasets.

Figures 16 and 17 compare the visual performance of Waste-YOLO with YOLOv5s on the PASCAL VOC and VisDrone2021 test sets, respectively. For simple scenes with large objects in figures 16(a) and (b), Waste-YOLO obtains higher detection accuracy than YOLOv5s. This indicates that Waste-YOLO can effectively detect ordinary objects. For scenarios with dense and similar targets in figure 16(c), YOLOv5s detects only a few sheep in the front while missing others while Waste-YOLO accurately identifies not only all sheep in the front but also the small sheep in the distance.

This demonstrates that Waste-YOLO has a higher recognition recall rate and better detection ability for small targets. As shown in figure 16(d), Waste-YOLO can yield more accurate frames to detect more boats than YOLOv5s. This further proves that SIoU loss has better bounding box positioning ability than GIoU loss. In figures 17(a) and (b), Waste-YOLO successfully detects an occluded white truck on the left side and two highly-overlapped trucks on the upper right and several densely-overlapped motorbikes in the central region, while YOLOv5s fails to do so. In addition, Waste-YOLO identifies more pedestrians and vehicles in the distance than YOLOv5s. The results indicate that our proposed Waste-YOLO has better detection in both occlusion and overlapping conditions and excels at detecting small dense objects.

Table 9. Performance of different models on PASCAL VOC07 + 12 and VisDrone2021. Data for the proposed Waste-YOLO model is bolded.

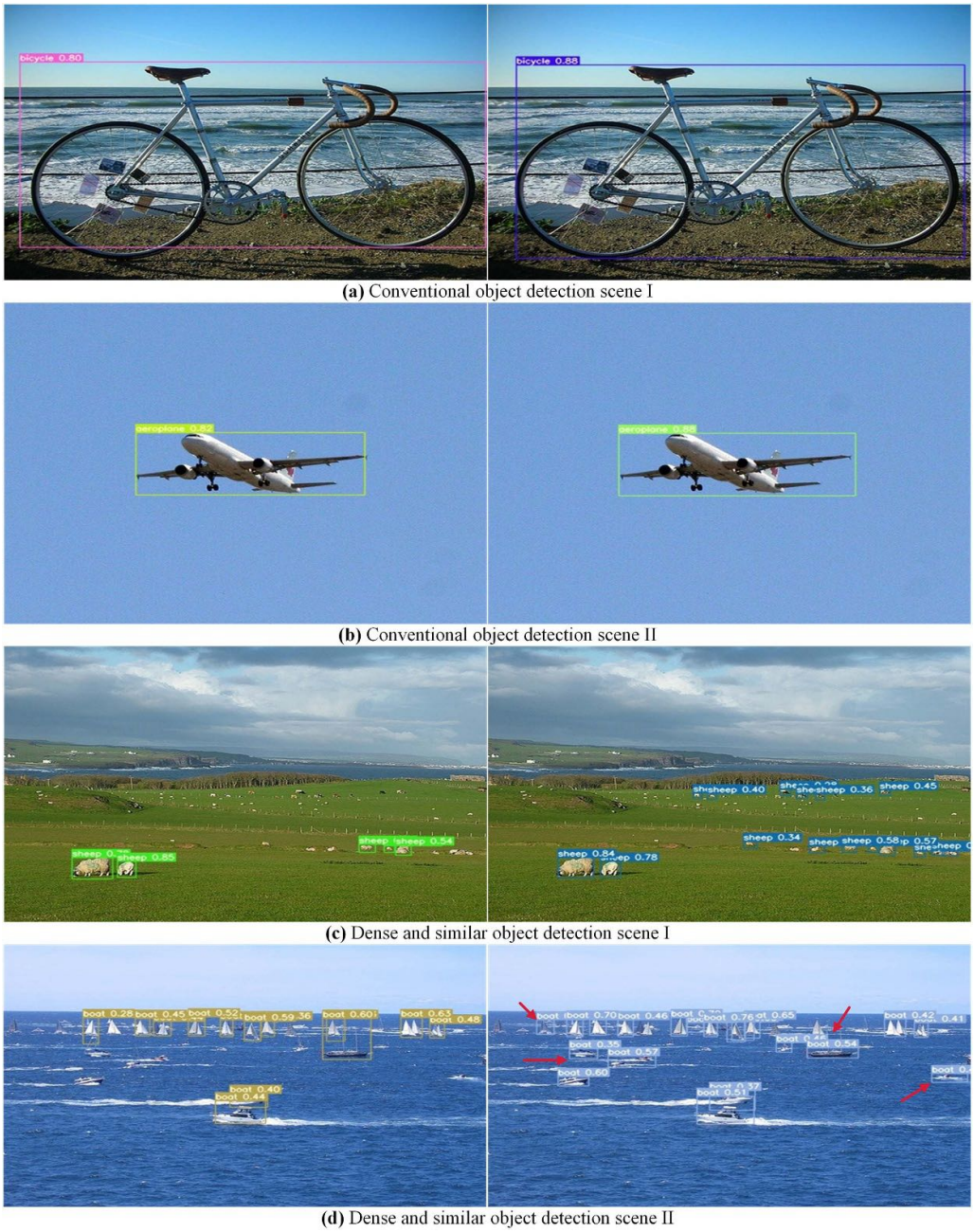
Dataset	Method	Epoch	mAP(0.5)	mAP(0.5:0.95)	Params (10^6)	FLOPs(G)	Speed-GPU (ms)
VOC07 + 12	YOLOv5s	150	71.5%	44.0%	7.115	16.5	9.1
	YOLOv8n	150	73.2%	51.5%	3.015	8.2	11.7
	YOLOv7-Tiny	150	70.7%	43.9%	6.066	13.3	7.65
	YOLOv6n	150	70.0%	46.8%	4.3	11.1	9.92
	DAMO-YOLO-T	150	66.2%	41.9%	8.64	18.46	21.27
	YOLOX-s	150	72.3%	44.7%	8.945	26.676	15.91
	SSD	150	49.63%	20.4%	6.071	7.297	14.24
	YOLOv4-Tiny	150	50.6%	21.9%	5.918	16.216	6.54
	CenterNet	150	46.2%	26.4%	32.688	109.563	13.82
	Faster-RCNN	150	65.9%	36.5%	28.47	946.705	40.73
	EfficientDet-D1	150	55.1%	31.5%	6.57	11.423	51.90
	YOLOv3-Tiny	150	49.7%	22.2%	8.714	13.0	6.5
	Waste-YOLO	**150**	**73.5%**	**47.4%**	**8.371**	**18.5**	**13.9**
VisDrone2021	YOLOv5s	300	28.3%	14.5%	7.088	16.5	17.1
	Waste-YOLO	**300**	**29.2%**	**14.9%**	**8.344**	**18.4**	**18.5**

Table 10. Single-class AP results on the VOC07 + 12 test set.

Category	YOLOv5s	Waste-YOLO
Airplane	81.1	85.1
Bicycle	78.7	81.2
Bird	68.4	69.3
Boat	57.4	62.6
Bottle	56.9	57.5
Bus	76.4	79.2
Car	83.5	84.0
Cat	79.3	82.7
Chair	57.0	55.9
Cow	81.5	83.0
Dining table	65.5	65.6
Dog	72.8	76.3
Horse	82.3	83.7
Motorbike	79.7	80.3
Person	81.4	82.1
Potted plant	46.6	47.4
Sheep	71.8	72.7
Sofa	62.0	65.3
Train	80.0	81.2
TV monitor	75.1	75.3

Table 11. Single-class AP results on the VisDrone2021 validation + test set.

Category	YOLOv5s	Waste-YOLO
Pedestrian	27.4	28.0
People	20.8	21.8
Bicycle	7.83	9.09
Car	68.6	69.2
Van	30.1	31.2
Truck	25.9	26.8
Tricycle	14.7	15.3
Awning-tricycle	10.7	11.3
Bus	48.9	49.9
Motor	28.9	29.8



(a) Conventional object detection scene I

(b) Conventional object detection scene II

(c) Dense and similar object detection scene I

(d) Dense and similar object detection scene II

Figure 16. Comparison of detection effects between YOLOv5s (left column) and Waste-YOLO (right column) on the VOC test set. Red arrows indicate areas of comparison.

According to the blurred image in figure 17(c), YOLOv5s not only fails to detect several pedestrians on the left and three side-by-side pedestrians below but also incorrectly identifies a bin at the bottom as a pedestrian, while Waste-YOLO detects more actual pedestrians.

In summary, Waste-YOLO has stronger feature extraction ability for blurry images than YOLOv5s with higher accuracy and recall. By introducing SIoU loss, our proposed model considers factors such as overlapping area, center point distance and aspect ratio similarity between target and prediction boxes to improve network regression accuracy and enhance sensitivity to small objects. As such, our model's bounding box detection effect is more aligned with actual targets than that of YOLOv5s.

Figure 17. Comparison of detection effects between YOLOv5s (left column) and Waste-YOLO (right column) on the VisDrone2021 test set. Red arrows indicate areas of comparison.

6. Conclusion

This paper presents Waste-YOLO, a novel YOLOv5s-based detection framework for real-time abnormal waste detection in complex backgrounds with high accuracy. The framework incorporates CA and CBAM blocks at the end of the C3 module to enhance the attention mechanism and filter out irrelevant features such as background and noise. It also employs CARAFE to increase the resolution of feature maps after upsampling and an improved BiFPN network to strengthen feature fusion between consecutive layers. Furthermore, it adopts SIoU as the bounding box loss function to improve the localization and regression of predicted boxes. To assess the performance of Waste-YOLO, we have developed an abnormal waste image dataset comprising four common types: mattresses, gas cans, wood, and iron sheets. We have conducted extensive experiments on this dataset and several public datasets. The quantitative results show that Waste-YOLO achieves significant improvement in detection accuracy over YOLOv5s with a slight efficiency trade-off. The qualitative results also confirm the superior detection effect of Waste-YOLO compared to several state-of-the-art methods including YOLOv8n. Therefore, the proposed abnormal waste image dataset can facilitate further research on abnormal waste detection, and the excellent performance of Waste-YOLO demonstrates its potential for practical engineering applications.

Future research may focus on increasing the data volume and abnormal waste categories, applying lightweight optimization techniques such as model pruning and knowledge distillation to further enhance detection speed, and deploying the framework on embedded devices with limited computing resources such as the NVIDIA Jetson Nano. Moreover, we will explore the use of more advanced object detection algorithms and enhanced techniques in abnormal waste detection.

Data availability statement

The data that support the findings of this study are available upon reasonable request from the authors.

Acknowledgments

This work was partly supported by the National Key Research and Development Program of China (Grant 2019YFE0105300), the National Natural Science Foundation of China (Grants 62377010, 62273139, 62171184 and 62106072), the Key Research Foundation of Education Bureau of Hunan Province (Grant 22A0021), and the Science and Technology Innovation Program of Hunan Province (Grant 2020GK2020).

ORCID iDs

He Wang https://orcid.org/0000-0002-9362-1062
Lianhong Wang https://orcid.org/0000-0001-8016-3042
Hua Chen https://orcid.org/0000-0002-1563-5274
Xiaoyao Li https://orcid.org/0000-0002-4600-1215
Xiaogang Zhang https://orcid.org/0000-0002-2576-2576
Yicong Zhou https://orcid.org/0000-0002-4487-6384

References

[1] Xu M and Lin B 2023 Accessing people's attitudes towards garbage incineration power plants: evidence from models correcting sample selection bias *Environ. Impact Assess. Rev.* **99** 107034
[2] Devi K S, Sujana O and Singh T C 2018 Hazardous waste management in India—a review *Int. J. Creat. Res. Thoughts* **6** 1547–55
[3] Lowe D G 2004 Distinctive image features from scale-invariant keypoints *Int. J. Comput. Vis.* **60** 91–110
[4] Dalal N and Triggs B 2005 Histograms of oriented gradients for human detection *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)* (*San Diego*) pp 886–93
[5] Jégou H, Douze M, Schmid C and Pérez P 2010 Aggregating local descriptors into a compact image representation *2010 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)* (*San Francisco*) pp 3304–11
[6] Peng X, Zou C, Qiao Y and Peng Q 2014 Action recognition with stacked Fisher vectors *European Conf. on Computer Vision (ECCV)* (*Zurich*) pp 581–95
[7] Salimi I, Dewantara B S B and Wibowo I K 2018 Visual-based trash detection and classification system for smart trash bin robot *2018 Int. Electronics Symp. on Knowledge Creation and Intelligent Computing (IES-KCIC)* (*Bali*) pp 378–83
[8] Chen L, Wu X, Sun C, Zou T, Meng K and Lou P 2023 An intelligent vision recognition method based on deep learning for pointer meters *Meas. Sci. Technol.* **34** 055410
[9] Huang Z, Hu H, Shen Z, Zhang Y and Zhang X 2022 Lightweight edge-attention network for surface-defect detection of rubber seal rings *Meas. Sci. Technol.* **33** 085401
[10] Ren S, He K, Girshick R and Sun J 2017 Faster R-CNN: towards real-time object detection with region proposal networks *IEEE Trans. Pattern Anal. Mach. Intell.* **39** 1137–49
[11] Girshick R 2015 Fast R-CNN *Proc. IEEE Int. Conf. on Computer Vision (ICCV)* (*Santiago*) pp 1440–8
[12] Nowakowski P and Pamuła T 2020 Application of deep learning object classifier to improve e-waste collection planning *Waste Manage.* **109** 1–9
[13] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C Y, Berg A C 2016 SSD: single shot MultiBox detector *European Conf. on Computer Vision (ECCV)* (*Amsterdam*) pp 21–37
[14] Redmon J, Divvala S, Girshick R and Farhadi A 2016 You only look once: unified, real-time object detection *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (*Las Vegas*) pp 779–88
[15] Redmon J and Farhadi A 2017 YOLO9000: better, faster, stronger *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (*Honolulu*) pp 7263–71
[16] Redmon J and Farhadi A 2018 YOLOv3: an incremental improvement (arXiv:1804.02767)
[17] Bochkovskiy A, Wang C Y and Liao H Y M 2020 YOLOv4: optimal speed and accuracy of object detection (arXiv:2004.10934)
[18] Ultralytics 2020 YOLOv5 repository (available at: https://github.com/ultralytics/yolov5) (Accessed 28 March 2023)
[19] Ge Z, Liu S, Wang F, Wang F, Li Z and Sun J 2021 YOLOX: exceeding YOLO series in 2021 (arXiv:2107.08430)
[20] Li C *et al* 2022 YOLOv6: a single-stage object detection framework for industrial applications (arXiv:2209.02976)
[21] Wang C Y, Bochkovskiy A and Liao H Y M 2022 YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors (arXiv:2207.02696)
[22] Xu X, Jiang Y, Chen W, Huang Y, Zhang Y and Sun X 2022 DAMO-YOLO: a report on real-time object detection design (arXiv:2211.15444)
[23] Ultralytics 2023 YOLOv8 repository (available at: https://github.com/ultralytics/ultralytics) (Accessed 28 March 2023)
[24] Lin T Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P and Zitnick C L 2014 Microsoft COCO: common objects in context *European Conf. on Computer Vision (ECCV)* (*Zurich*) pp 740–55
[25] Patel D, Patel F, Patel S, Patel N, Shah D and Patel V 2021 Garbage detection using advanced object detection techniques *2021 Int. Conf. on Artificial Intelligence and Smart Systems (ICAIS)* (*Coimbatore*) pp 526–31
[26] Mao W-L, Chen W-C, Fathurrahman H I K and Lin Y-H 2022 Deep learning networks for real-time regional domestic waste detection *J. Clean. Prod.* **344** 131096
[27] Li J, Chen J, Sheng B, Li P, Yang P, Feng D D and Qi J 2022 Automatic detection and classification system of domestic waste via multimodel cascaded convolutional neural network *IEEE Trans. Industr. Inform.* **18** 163–73
[28] Yang M and Thung G 2016 Classification of trash for recyclability status *CS229 Project Report, 2016* p 3
[29] Mittal G, Yagnik K B, Garg M and Krishnan N C 2016 Spotgarbage: smartphone app to detect garbage using deep learning *Proc. 2016 ACM Int. Joint Conf. on Pervasive and Ubiquitous Computing* (*New York*) pp 940–5
[30] Proença P F and Simoes P 2020 TACO: trash annotations in context for litter detection (arXiv:2003.06975)

[31] Panwar H, Gupta P K, Siddiqui M K, Morales-Menendez R, Bhardwaj P, Sharma S and Sarker I H 2020 AquaVision: automating the detection of waste in water bodies using deep transfer learning *Case Stud. Chem. Environ. Eng.* **2** 100026
[32] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (*Las Vegas*) pp 770–8
[33] Lin T Y, Dollar P, Girshick R, He K, Hariharan B and Belongie S 2017 Feature pyramid networks for object detection *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (*Honolulu*) pp 2117–25
[34] Liu S, Qi L, Qin H, Shi J and Jia J 2018 Path aggregation network for instance segmentation *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (*Salt Lake City*) pp 8759–68
[35] Rezatofighi H, Tsoi N, Gwak J, Sadeghian A, Reid I and Savarese S 2019 Generalized intersection over union: a metric and a loss for bounding box regression *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* (*Long Beach*) pp 658–66
[36] Hou Q, Zhou D and Feng J 2021 Coordinate attention for efficient mobile network design *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* (*Virtually*) pp 13713–22
[37] Howard A *et al* 2019 Searching for MobileNetV3 *Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV)* (*Long Beach*) pp 1314–24
[38] Woo S, Park J, Lee J Y and Kweon I S 2018 CBAM: convolutional block attention module *Proc. European Conf. on Computer Vision (ECCV)* (*Munich*) pp 3–19
[39] Tan M, Pang R and Le Q V 2020 EfficientDet: scalable and efficient object detection *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* (*Seattle*) pp 10781–90
[40] Wang J, Chen K, Xu R, Liu Z, Loy C C and Lin D 2019 CARAFE: Aware ReAssembly of FEatures *Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV)* (*Seoul*) pp 3007–16
[41] Zheng Z, Wang P, Liu W, Li J, Ye R and Ren D 2020 Distance-IoU loss: faster and better learning for bounding box regression *Proc. AAAI Conf. on Artificial Intelligence* (*New York*) pp 12993–3000
[42] Gevorgyan Z 2022 SIoU loss: more powerful learning for bounding box regression (arXiv:2205.12740)
[43] Selvaraju R R, Cogswell M, Das A, Vedantam R, Parikh D and Batra D 2017 Grad-CAM: visual explanations from deep networks via gradient-based localization *Proc. IEEE Int. Conf. on Computer Vision (ICCV)* (*Venice*) pp 618–26
[44] Lin T Y, Goyal P, Girshick R, He K and Dollár P 2017 Focal loss for dense object detection *Proc. IEEE Int. Conf. on Computer Vision (ICCV)* (*Venice*) pp 2980–8
[45] Li X, Wang W, Wu L, Chen S, Hu X, Li J, Tang J and Yang J 2020 Generalized focal loss: learning qualified and distributed bounding boxes for dense object detection *Advances in Neural Information Processing Systems* vol 33 pp 21002–12
[46] Zhang H, Wang Y, Dayoub F and Sunderhauf N 2021 VarifocalNet: an IoU-aware dense object detector *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* (*Nashville*) pp 8514–23
[47] Everingham M, Van Gool L, Williams C K I, Winn J and Zisserman A 2010 The PASCAL visual object classes (VOC) challenge *Int. J. Comput. Vis.* **88** 303–38
[48] Cao Y *et al* 2021 VisDrone-DET2021: the vision meets drone object detection challenge results *Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV) Workshops* (*Montreal*) pp 2847–54