

# 基于 YOLOv3 的农田鸟类目标检测算法

潘语豪, 危疆树\*, 曾令鹏

四川农业大学信息工程学院, 四川 雅安 625000

**摘要** 针对近年来鸟类啄食对农业生产带来的损失,通过实时检测鸟类优化传统驱鸟器开关策略,提出了一种基于 YOLOv3 检测鸟类的目标检测算法。该方法对 YOLOv3 网络中特征融合进行改进,将 SE 模块嵌入进主干网络的 Darknet53 网络中,关注不同通道特征的重要程度。采用自适应空间特征融合(ASFF)增强网络中特征金字塔网络(FPN)的特征融合,提升各尺度的检测能力。引入 CIOU 边界框回归损失函数,将预测框和目标框在有重叠甚至包含等情况考虑进去,使目标框回归变得更加准确和稳定。改进后的 YOLOv3 模型在自制鸟类数据集上的精度均值(AP)达到 96.65%,单张图像检测耗时仅为 0.058 s,相比于原 YOLOv3 模型在检测速度变化不大的情况下 AP 提高了 2.54 个百分点。该改进方法能达到很好的实时性和更佳检测精度,对农田防治鸟害优化驱鸟器开关策略提供依据。

**关键词** 视觉光学; 目标检测; YOLOv3; SE 模块; 特征融合; 边界框回归损失函数

中图分类号 TP391.4;S126

文献标志码 A

doi: 10.3788/LOP202259.0233001

## Farmland Bird Detection Algorithm Based on YOLOv3

Pan Yuhao, Wei Jiangshu\*, Zeng Lingpeng

College of Information Engineering, Sichuan Agricultural University, Yaan, Sichuan, 625000, China

**Abstract** In view of the loss of agricultural production caused by bird pecking in recent years, a target detection algorithm based on YOLOv3 bird detection was proposed to optimize the switching strategy of traditional bird repellent machine by identifying birds accurately in real time. This method improves the feature fusion in YOLOv3 network, and brings SE module embedded in the Darknet53 network of the trunk network, focusing on the importance of different channel characteristics. Using adaptive spatial feature fusion (ASFF) enhanced network feature pyramid network (FPN) feature fusion to enhance the detection capability of all scales. The loss function of CIOU boundary box regression is used to take into account the overlap or even inclusion of the prediction-box and the target-box, so that the target-box regression becomes more accurate and stable. The improved YOLOv3 model was used on the self-made bird data set. The average precision (AP) reached 96.65%, and the detection time of single image was only 0.058 s, which increased by the 2.54 percentage point compared with the original YOLOv3 model under the condition of little change in detection speed. The improved method can achieve good real-time performance and better detection accuracy, and provide a basis for optimizing the switching strategy of bird repellent for farmland bird prevention and control.

**Key words** visual optics; object detection; YOLOv3; SE module; feature fusion; loss function of bounding-box regression

## 1 引言

鸟类对农业作物的破坏是一个长期存在的严

重问题,鸟类啄食果实会使其发生病变,并且果品质量也会受到严重影响<sup>[1]</sup>,鸟类啄食农作物种子、花瓣花蕾等会造成农作物减产,进而造成较大的经济

收稿日期: 2020-12-29; 修回日期: 2021-01-19; 录用日期: 2021-03-09

基金项目: 赛尔网络下一代互联网技术创新项目(NGII20170605)

通信作者: \*sicauwjs@163.com

损失<sup>[2]</sup>。近年来,随着环保意识的增强,人们加大了对鸟类的保护力度<sup>[3]</sup>,架设防鸟网、撒毒饵诱杀等会伤害鸟类的方法已逐渐被替代。为了减少鸟类对农作物的啄食,人们研究了各种鸟类威慑的方法,如采用无人机方法与一种涉及静态视觉装置的鸟类恐吓方法<sup>[4]</sup>,但是因为无人机的高成本,所以现代驱鸟方法主要还是采用液化气炮驱鸟器、语音驱鸟器和超声波驱鸟器等机械、电子或生物的方式使鸟类产生不适和恐惧感而迅速离开<sup>[5-6]</sup>。但是驱鸟器长期处于工作状态会浪费资源,并且鸟类会产生适应性进而意识到没有实际的危险,从而达不到驱鸟的效果。

近年来,随着深度学习算法的不断发展,出现了各种目标检测算法,其目的是定位并识别图像中单个或多个目标,返回目标位置的同时对其所属类别进行正确分类<sup>[7-9]</sup>,可以大致分为两类:1)两步检测,指的是检测算法分两步完成,需要获得候选区域然后进行分类,如区域卷积神经网络(R-CNN)借鉴了滑动窗口的思想,将候选区域与卷积神经网络(CNN)相结合进行目标定位,但是其训练步骤繁琐并且占用空间<sup>[10]</sup>;Fast-R-CNN在R-CNN基础上提出感兴趣区(ROI)层,以便进行后续的分类和输出回归框操作,加速处理速度<sup>[11]</sup>;Faster-R-CNN则提出了区域生成网络(RPN)候选框提取模块,提高了检测精度和速度,但是仍无法实现实时检测的目的<sup>[12]</sup>。2)单步检测算法,通过主干网络直接给出类别和位置信息,这类基于回归的目标检测算法如基础的YOLO(You Only Look Once)对整张图输入,在输出层回归边界框位置信息和类别<sup>[13]</sup>。YOLOv2在YOLO的基础上采用多尺度训练,根据

需求权衡精度和速度<sup>[14]</sup>。SSD300基于Faster-R-CNN和YOLO构建端对端的网络,但缺点是对小目标检测效果不佳<sup>[15]</sup>;YOLOv3采用残差模块构建更深层网络<sup>[16]</sup>,特征图金字塔网络(FPN)架构实现多尺度预测。这类算法的优点是检测速度快、实时性好,但是相比于两步检测算法精度较低。国内外学者对目标检测算法的改进也日益增多,韩永赛等<sup>[17]</sup>优化了RPN对机场区域检测,刘万军等<sup>[18]</sup>优化了YOLOv2多尺度融合特征,Hong等<sup>[19]</sup>对比了多种深度学习算法应用于无人机飞行图像鸟类的检测,Akçay等<sup>[20]</sup>利用Faster-R-CNN对鸟类自动检测计数,陈咏秋等<sup>[21]</sup>基于YOLOv3研究了输电线路鸟类检测。

本文基于SENet(Squeeze-and-Excitation Networks)<sup>[22]</sup>改进YOLOv3的Darknet53网络中的残差模块,使用自适应空间特征融合(ASFF)<sup>[23]</sup>对原YOLOv3中FPN输出的三个特征层再次进行特征融合,并结合三个特征层的特征值来丰富物体信息。本文引入CIOU边界框回归损失函数<sup>[24]</sup>,使目标框回归变得更加稳定,从而提高算法的准确率和目标边界框的定位精度。在训练过程中使用余弦退火衰减法<sup>[25]</sup>,在不增加训练成本的前提下,提升模型效果。

## 2 数据获取方法

### 2.1 数据采集及其标注方法

通过实地农田拍摄和网络采集30个鸟类的视频,每20 frame提取一张图片,考虑到鸟类在不同距离的检测情况并且对其中不清晰的图片进行筛选,最终选取2855张图片作为数据集,部分样本数据如图1所示。



图1 鸟类样本图像

Fig. 1 Images of bird samples

采用了数据增强的方式来扩展数据集,包括随机缩放、剪切和旋转等操作。采用9:1的比例划分

数据集,选择2569张作为训练集,286张作为测试集,再对训练集采用十折交叉验证法划分验证集作

为对算法准确性的估计,采用LabelImg标注工具通过人工对每幅图像进行标注。

## 2.2 YOLOv3网络模型及其改进

### 2.2.1 YOLOv3网络结构

YOLOv3网络是由Redmon等<sup>[16]</sup>提出的一种目

	type	filter	size	output
1×	convolutional	32	3×3	416×416
	convolutional	64	3×3/2	208×208
	convolutional	32	1×1	
	convolutional	64	3×3	
	residual			208×208
2×	convolutional	128	3×3/2	104×104
	convolutional	64	1×1	
	convolutional	128	3×3	
	residual			104×104
	convolutional	256	3×3/2	52×52
8×	convolutional	128	1×1	
	convolutional	256	3×3	
	residual			52×52
	convolutional	512	3×3/2	26×26
	convolutional	256	1×1	
8×	convolutional	512	3×3	
	residual			26×26
	convolutional	1024	3×3/2	13×13
	convolutional	512	1×1	
	4×	convolutional	1024	3×3
residual				13×13

标检测算法。如图2所示, YOLOv3的主干网络Darknet-53借鉴了Resnet<sup>[26]</sup>的思想,在网络中加入了残差模块,有利于解决深层网络的梯度消失和爆炸问题。网络中没有使用池化层和全连接层,特征图的尺寸变换是通过改变卷积核的步长来实现的。

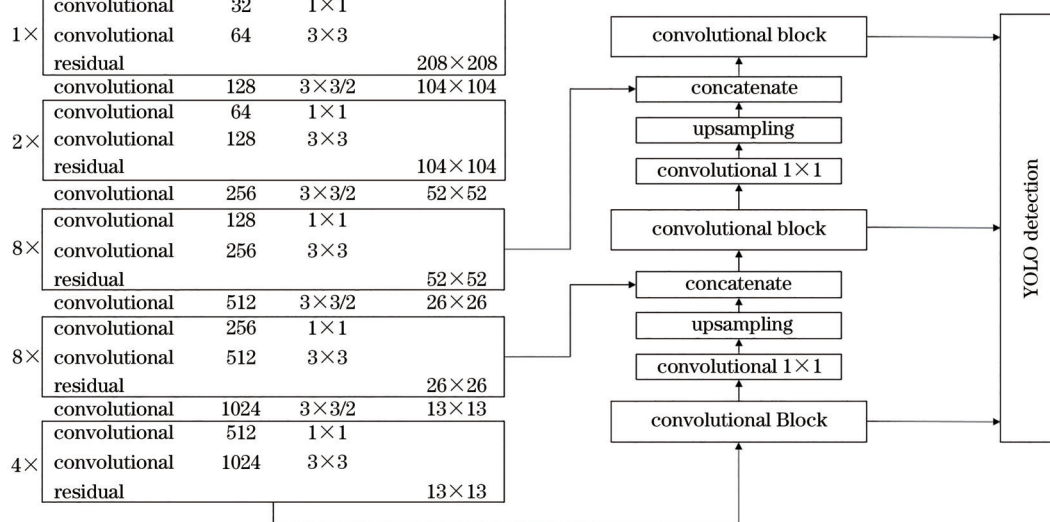


图2 YOLOv3的网络结构图

Fig. 2 Network structure diagram of YOLOv3

YOLOv3在特征融合方式上借鉴FPN<sup>[27]</sup>的思想,采用上采样和合并操作融合三个尺度(13×13、26×26和52×52)的特征层,在多个尺度的融合特征图上独立检测,提升了小目标检测能力。

### 2.2.2 残差模块内通道间的特征融合

SENet是Hu等<sup>[22]</sup>提出的网络结构,主要关注于主干网络中卷积操作里的通道间的特征融合。通过压缩(Squeeze)、激励(Excitation)和重标定(Reweight)操作完成在通道上对原始特征图的重新标定。

如图3所示,本文考虑将SE module嵌入进主干网络中的残差模块(Resblock)中,最关键的两个步骤是Squeeze和Excitation操作,首先是在空间维度层面进行Squeeze,将每个二维的特征通道变为一个实数,代表特征通道上响应的全局分布。然后是一个类似于门控机制的Excitation操作分别生成各个通道的特征权重来表示不同特征通道间的相关性,依照这个权重去提升有用的特征并抑制不太重要的特征。

1) Squeeze。在特征层上执行全局平均池化,对于输出的每一个通道的特征图的所有像素计算

一个平均值,这个数值在某种程度上具有全局的感受野,代表该通道特征向量的重要程度,即 $z_c$ 代表Squeeze操作的结果。

普通卷积操作表示为

$$u_c = \sum_{m=1}^{c'} v_c^m \cdot x^m, \quad (1)$$

式中: $v_c$ 表示第 $c$ 个卷积核; $x^m$ 表示第 $m$ 个输入;最终得到的 $u_c$ 是卷积后的三维矩阵中的第 $c$ 个二维矩阵。严格来说,(1)式是属于原网络的卷积操作,从之后的操作开始才是属于SE module的部分。Squeeze操作的计算公式为

$$z_c = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W u_c(i, j), \quad (2)$$

式中: $z_c$ 为对应特征图的全局信息;将 $H \times W \times C$ 的输入转换成 $1 \times 1 \times C$ 的输出, $H$ 和 $W$ 分别代表图片的高和宽。

2) Excitation。通过上一步Squeeze学习各个通道间的关系,得到不同通道的特征权重。使用两层全连接构成门控机制,门控单元( $s$ )的计算方法为

$$s = \sigma \{ g [ \mathbf{W}_2 \delta ( \mathbf{W}_1 z ) ] \}, \quad (3)$$

式中: $z$ 是Squeeze操作得到数值结果的集合; $\mathbf{W}_1$ 和

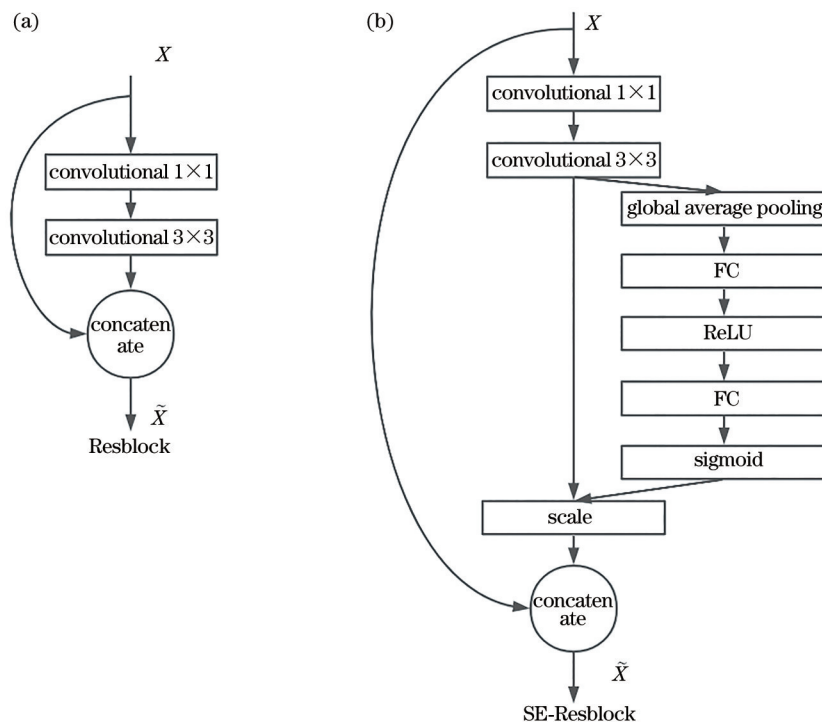


图 3 残差模块。(a)原始的 Resblock; (b) SE-Resblock

Fig. 3 Residual module. (a) Original Resblock; (b) SE-Resblock

$W_2$  分别是两个全连接层的权值矩阵, 维度分别是  $\frac{C}{r} \times C$  和  $C \times \frac{C}{r}$ ,  $r$  是缩放参数,  $C$  参考原文实验取值为 16;  $\delta$  为 ReLU 激活函数;  $\sigma$  为 Sigmoid 激活函数;  $s$  表示通过全连接层和非线性层得到的  $c$  个特征图的权重集合。

3) Reweight。如图 3 通过乘法逐通道将 Excitation 输出的权重加权到原来的特征上。

$$\tilde{x}_c = s_c \cdot u_c, \quad (4)$$

式中:  $\tilde{x}_c$  是  $\tilde{X}$  的某个特征通道的一个特征图;  $s_c$  是  $s$  中的一个权重。

### 2.2.3 特征金字塔自适应特征融合

ASFF 的核心内容是在不同尺度进行特征融合时能自适应地调整权重。YOLOv3 采用在类似 FPN 的特征金字塔结构进行上采样和融合操作, 融合了 3 个不同尺度, 在多个尺度的特征图上分别独立做检测。但是 FPN 存在的问题是不同的特征尺度之间通过直接相加的方式会有差异性, 而 ASFF 的思想就是与 FPN 相结合, 对于每一层尺度都学习一个融合权重, 在预测特征图时能更好地利用不同尺度的特征。图 4 表示改进后的网络结构。

如图 5 所示, 以 ASFF-2 结构为例,  $X_1, X_2, X_3$  分

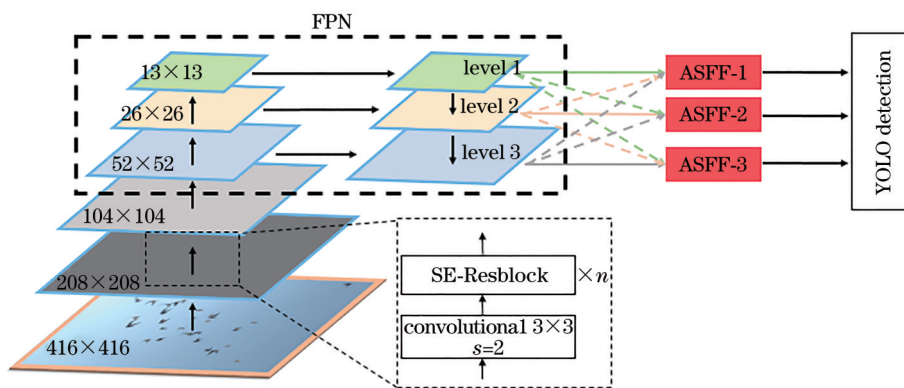


图 4 改进后的网络结构

Fig. 4 Improved network structure

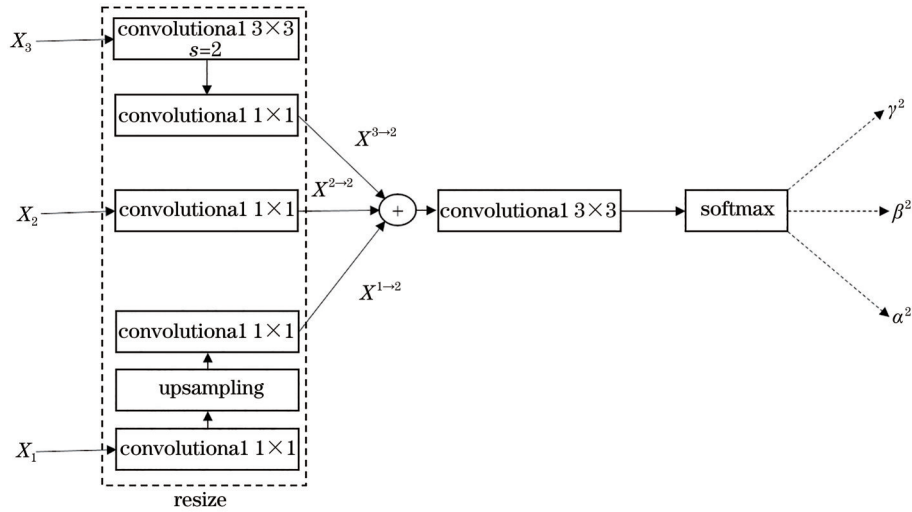


图 5 ASFF-2 的结构

Fig. 5 Structure of ASFF-2

别是来自 level 1, level 2, level 3 的特征, 不同层的特征乘上权重参数( $\alpha^2$ 、 $\beta^2$  和  $\gamma^2$ ) 并求和得到。

$$y_{ij}^l = \alpha_{ij}^l \cdot x_{ij}^{1 \rightarrow l} + \beta_{ij}^l \cdot x_{ij}^{2 \rightarrow l} + \gamma_{ij}^l \cdot x_{ij}^{3 \rightarrow l} \quad (5)$$

Level 1, Level 2, Level 3 的特征图通过 Resize 操作得到  $X^{1 \rightarrow 2}$ ,  $X^{2 \rightarrow 2}$ ,  $X^{3 \rightarrow 2}$ , 保证在特征融合时输入的尺寸一致。

$$\alpha_{ij}^l + \beta_{ij}^l + \gamma_{ij}^l = 1, \quad (6)$$

权重参数  $\alpha_{ij}^l$ ,  $\beta_{ij}^l$ ,  $\gamma_{ij}^l$  经过 concat 之后通过 Softmax 使它们的范围都在  $[0, 1]$ , 并且和为 1。

#### 2.2.4 边界框回归损失函数优化

YOLOv3 根据预测框和目标框的中心点坐标以及宽高信息作为独立变量设定均方误差(MSE)。但是框的中心点和宽、高存在一定的关系, 因此可以利用交并比(IoU)<sup>[28]</sup>, 即预测框和目标框的交并比, 代替 MSE。但是一个好的边界框回归损失函数应该考虑更多的几何因素, Zheng 等<sup>[24]</sup>提出的 CIOU 损失函数将目标框与预测框的距离、重叠率以及尺度都考虑进去, 使得目标框回归变得更加稳定, 不会像 IoU 等在训练过程中出现发散的问题, 其收敛精度更高, 表达式为

$$L_{CIOU} = 1 - I_{IoU} + \frac{\rho^2(b, b_{gt})}{d_m^2} + \alpha \cdot v, \quad (7)$$

式中:  $b$  和  $b_{gt}$  表示预测框与目标框的中心点;  $\rho^2(b, b_{gt})$  表示欧氏距离;  $d_m$  表示预测框与目标框的最小外接矩形的对角线距离;  $\alpha$  为权重函数;  $v$  为测量长宽比一致性的参数。如图 6 所示, 左上框表示目标框, 右下框表示预测框, 虚线框为最小外接矩形,  $d_m$  和  $d$  分别表示最小外接矩形的对角线距离和



图 6 预测框和目标框之间的归一化距离

Fig. 6 Normalized distance between prediction frame and target frame

两个框中心点的欧氏距离。

$\alpha$  和  $v$  的表达式为

$$\alpha = \frac{v}{1 - I_{IoU} + v}, \quad (8)$$

$$v = \frac{4}{\pi^2} \left( \arctan \frac{w_{gt}}{h_{gt}} - \arctan \frac{w_p}{h_p} \right), \quad (9)$$

式中:  $w_{gt}$  和  $h_{gt}$  分别表示目标框的宽和高;  $w_p$  和  $h_p$  分别表示预测框的宽和高。

#### 2.3 实验环境

模型的训练在 Windows 10 操作系统, Pytorch 框架下完成, 实验硬件环境为微星 B450M MORTAR MAX 主板, CPU 型号为 AMD Ryzen 5 3600, GPU 型号为 NVIDIA GeForce RTX 2060 SUPER 8 G, 软件环境为 CUDA 10.1, CUDNN 7.6, Python3.7。

## 2.4 模型训练策略

### 2.4.1 训练方法

利用 YOLOv3 在 COCO 数据集上的预训练模型初始化参数,可以加快模型的训练速度。共设置 500 个 epoch,使用冻结训练,在前 250 个 epoch 冻结主干网络,初始学习率设为 0.001 可以收敛得更快。在第 251 个 epoch 解冻训练,同时学习率减小为 0.0001,更新整个网络参数权重。

### 2.4.2 余弦退火衰减法

余弦退火衰减法不同于传统方法,学习率随 epoch 的增加会快速下降,模型会找到局部最优点并保存当前的模型。之后,学习率会陡然提升到一个较大的值,从当前的局部最优点逃离并重新寻找新的局部最优点,然后不断重复这个过程调整学习率使其按照周期变化直到训练完成。

$$l_{\text{new}} = l_{\text{min}} + (l_{\text{init}} - l_{\text{min}}) \times [1 + \cos\left(\frac{E_{\text{epoch}}}{T_{\text{max}}} \pi\right)], \quad (10)$$

式中: $l_{\text{min}}$  表示最小学习率; $l_{\text{init}}$  表示初始学习率; $T_{\text{max}}$  表示学习率变化周期的四分之一; $l_{\text{new}}$  表示得到的新的学习率。本文训练设置  $T_{\text{max}}$  为 5,  $l_{\text{min}}$  为 0.00001,前 100 个 epoch 学习率变化曲线如图 7 所示。

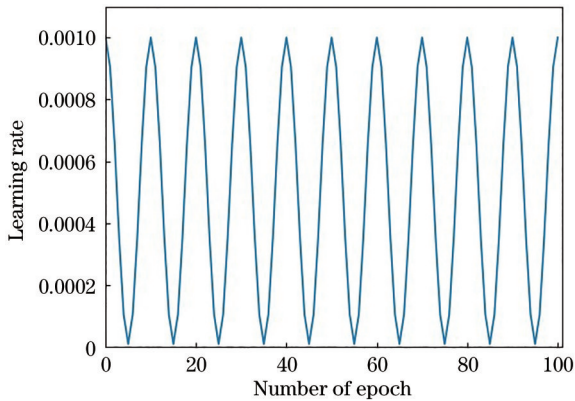


图 7 采用余弦退火衰减法的学习率随 epoch 次数变化曲线  
Fig. 7 Curve of learning rate changing with epoch times using annealing attenuation method

### 2.4.3 评价指标

对测试样本计算精度 ( $P_{\text{recision}}$ ) 和召回率 ( $R_{\text{ecall}}$ ),引入精度均值 (AP) 和检测单张图片耗时作为评估指标。以  $R_{\text{ecall}}$  为横轴,  $P_{\text{recision}}$  为纵轴绘制  $P$ - $R$  曲线并对其积分求出曲线下的面积,表达式为

$$P_{\text{recision}} = T_{\text{P}} / (T_{\text{P}} + F_{\text{P}}), \quad (11)$$

$$R_{\text{ecall}} = T_{\text{P}} / (T_{\text{P}} + F_{\text{N}}), \quad (12)$$

$$A_{\text{P}} = \int_0^1 p(r) dr, \quad (13)$$

式中: $T_{\text{P}}$  为分类正确的正样本; $F_{\text{P}}$  为分类错误的负样本; $F_{\text{N}}$  为分类错误的正样本; $p(r)$  为  $P$ - $R$  曲线函数表达。

## 3 分析与讨论

### 3.1 不同特征融合模块优化性能

表 1 中 Version-1~Version-3 分别表示在网络层采用不同特征融合模块对网络性能的影响。相比于原模型,其中 Version-1 添加 ASFF 模块,AP 提升了 0.87 百分点, $t$  延长了 0.004 s。Version-2 添加 SE 模块,AP 提升了 0.23 百分点, $t$  缩短了 0.005 s。Version-3 同时添加 ASFF 与 SE 模块,AP 提升 1.81 百分点, $t$  仅延长了 0.004 s。由表 1 可知,分别添加不同特征融合模块都会给模型带来性能的提升,并且这些增益是互补的,将其结合在一起能进一步增强网络性能。

表 1 不同特征融合模块在测试集上的性能对比

Model	ASFF	SE	AP / %	$t$ / s
YOLOv3			94.11	0.047
Version-1	✓		94.98	0.051
Version-2		✓	94.34	0.042
Version-3	✓	✓	95.92	0.052

### 3.2 边界框回归损失函数优化性能

实验研究了采用不同边界框回归损失函数对已改进的网络结构的影响。由表 2 可知, YOLOv3-ours 为本文改进的最终版本,相比于没有采用 CIOU 损失函数的 Version-3, AP 提升了 0.73 百分点, $t$  仅延长了 0.006 s。相比于原版, AP 提升了 2.54 百分点, $t$  仅延长了 0.011 s,检测速度仍能满足实时检测的标准。图 8 对比了 YOLOv3、Version-3 和 YOLOv3-ours 三种算法的损失值变化曲线。由图 8 可知,本文提出的 YOLOv3-ours 相比于

表 2 边界框回归损失函数优化在测试集上的性能对比

Model	CIOU	AP / %	$t$ / s
YOLOv3		94.11	0.047
Version-3		95.92	0.052
YOLOv3-ours	✓	96.65	0.058

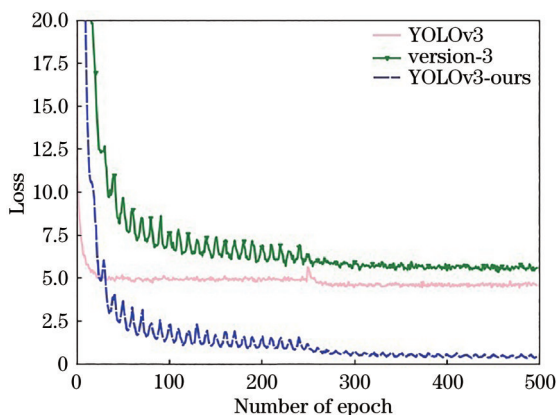


图 8 不同检测算法损失值随 epoch 次数变化曲线

Fig. 8 Loss curves of different detection algorithms varying with number of epoch

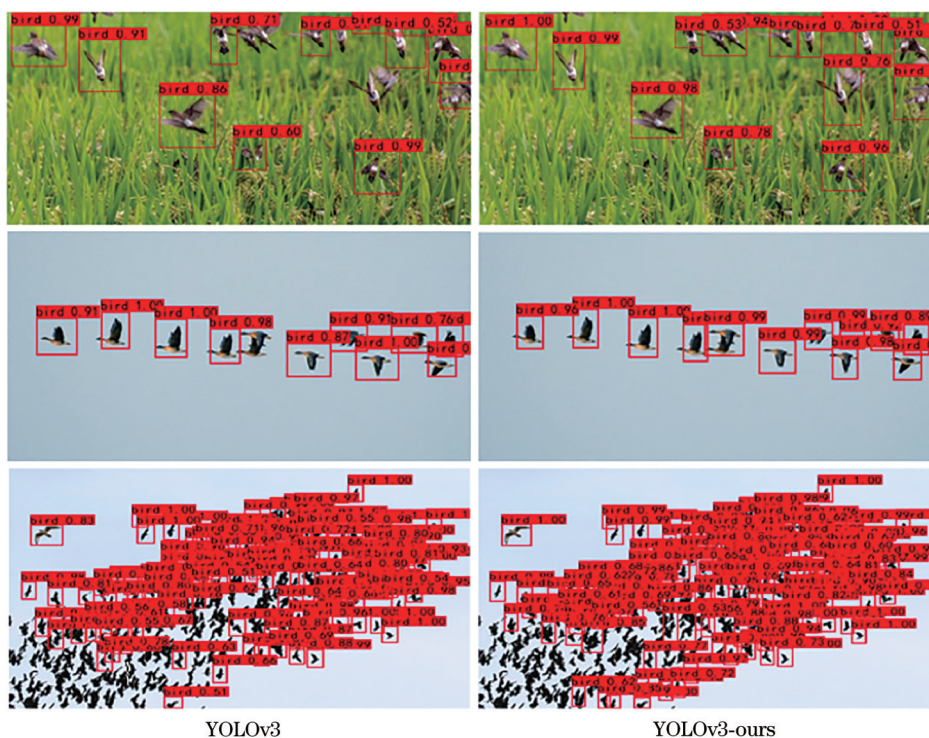


图 9 改进前后检测算法效果对比

Fig. 9 Comparison of detection algorithm before and after improvement

由表 3 可知,相比于 YOLOv3 模型 AP 提升了 2.54 百分点,  $t$  仅延长了 0.011 s, 检测速度仍能满足实时检测的标准。对比目前主流检测算法, 本文所

表 3 不同检测算法比较

Table 3 Comparison of different detection algorithms

Model	AP / %	$t$ / s
YOLOv3	94.11	0.047
SSD300	93.72	0.859
Faster-R-CNN	97.38	1.244
YOLOv3-ours	96.65	0.058

YOLOv3, 使用余弦退火衰减法后, 损失值的变化曲线呈波浪状下降趋势, 在第 250 个 epoch 解冻训练后收敛速度变化不大的情况下得到的损失值更低。相比于 Version-3, 在使用 CIUO 后, 能得到更快的收敛速度和更低的损失值。

### 3.3 与其他目标检测算法对比实验

实验对比了改进的 YOLOv3 模型和目前主流检测算法在应用中的性能。本文模型在主干网络和特征金字塔部分分别嵌入 ASFF 和 SE Module 两个特征融合模块, 使用 CIUO 优化边界框回归损失函数。图 9 表示改进前后模型检测结果对比, 可以看出在检测精度和置信度上改进后的模型都有提升。

提出的模型检测精度上仅比 Faster-R-CNN 低了 0.73 百分点, 单张图检测速度约为 SSD300 的 15 倍, Faster-R-CNN 的 20 倍, 综合性能更优, 更适合于农田鸟类的实时目标检测。

## 4 结 论

为了实现实时检测农田鸟类优化传统驱鸟器开关策略的目的, 本文提出了一种改进 YOLOv3 的方法, 在主干网络和特征金字塔部分嵌入特征融合模块 ASFF 和 SE Module 提升模型提取特征能力,

提高鸟类小目标的检测准确性。引入CIOU边界框回归损失函数提升模型预测框准确定位能力,增强模型收敛效果。本文模型与原YOLOv3模型相比,AP提升了2.54个百分点, $t$ 仅延长了0.011 s。与主流目标检测方法SSD300相比,AP高出2.93个百分点,单张图检测速度快出15倍。与Faster-R-CNN相比,AP仅低0.73个百分点,单张图检测速度快出20倍。该模型在检测精度和检测速度的综合考虑下,能保证农田鸟类实时检测的目的。尽管本文方法有较好的性能,但仍存在的不足之处是提升检测精度的同时不可避免地降低了检测速度,后续研究工作可以通过对网络进行目前流行的剪枝技术来达到减少模型参数、提高检测速度的目的。

### 参 考 文 献

- [1] Anderson A, Lindell C A, Moxcey K M, et al. Bird damage to select fruit crops: the cost of damage and the benefits of control in five states[J]. *Crop Protection*, 2013, 52: 103-109.
- [2] Liu P P, Zhao X R, Zhang H J, et al. Research progress on common agricultural harmful birds and their control in China[J]. *Jiangsu Agricultural Sciences*, 2010, 38(2): 139-141.  
刘培培, 赵欣如, 张红娟, 等. 中国常见农业害鸟及其防治研究进展[J]. *江苏农业科学*, 2010, 38(2): 139-141.
- [3] Li C H, Wang Z Y, Ouyang F Z, et al. Bird damage of rape and control methods[J]. *Crop Research*, 2013, 27(4): 373-375.  
李财厚, 王宗尧, 欧阳凤仔, 等. 油菜田鸟害及其防治措施[J]. *作物研究*, 2013, 27(4): 373-375.
- [4] Wang Z H, Fahey D, Lucas A, et al. Bird damage management in vineyards: comparing efficacy of a bird psychology-incorporated unmanned aerial vehicle system with netting and visual scaring[J]. *Crop Protection*, 2020, 137: 105260.
- [5] Xiong W J, Wu X B, Pang G L, et al. Modern technology of driving birds in farmland[J]. *Modernizing Agriculture*, 2009(6): 51.  
熊文江, 吴显斌, 庞国利, 等. 现代农田驱鸟技术[J]. *现代化农业*, 2009(6): 51.
- [6] Zhang C, Li G L, Jin C, et al. On design and test of intelligent equipment of driving farmland birds[J]. *Journal of Southwest China Normal University (Natural Science Edition)*, 2016, 41(5): 81-87.  
张驰, 李光林, 金城, 等. 农田智能驱鸟器设计与试验[J]. *西南师范大学学报(自然科学版)*, 2016, 41(5): 81-87.
- [7] Jiao L C, Zhang F, Liu F, et al. A survey of deep learning-based object detection[J]. *IEEE Access*, 2019, 7: 128837-128868.
- [8] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks [J]. *Communications of the ACM*, 2017, 60(6): 84-90.
- [9] Duan Z J, Li S B, Hu J J, et al. Review of deep learning based object detection methods and their mainstream frameworks[J]. *Laser & Optoelectronics Progress*, 2020, 57(12): 120005.  
段仲静, 李少波, 胡建军, 等. 深度学习目标检测方法及其主流框架综述[J]. *激光与光电子学进展*, 2020, 57(12): 120005.
- [10] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//2014 IEEE Conference on Computer Vision and Pattern Recognition, June 23-28, 2014, Columbus, OH, USA. New York: IEEE Press, 2014: 580-587.
- [11] Girshick R. Fast R-CNN[C]//2015 IEEE International Conference on Computer Vision (ICCV), December 7-13, 2015, Santiago, Chile. New York: IEEE Press, 2015: 1440-1448.
- [12] Ren S Q, He K M, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, 39(6): 1137-1149.
- [13] Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 779-788.
- [14] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 6517-6525.
- [15] Liu W, Anguelov D, Erhan D, et al. SSD: single shot MultiBox detector[M]//Leibe B, Matas J, Sebe N, et al. *Computer vision-ECCV 2016. Lecture notes in computer science*. Cham: Springer, 2016, 9905: 21-37.
- [16] Redmon J, Farhadi A. YOLOv3: an incremental improvement[EB/OL]. (2018-04-08) [2020-12-25]. <https://arxiv.org/abs/1804.02767>.



- [17] Han Y S, Ma S P, Li S, et al. Airport area detection based on optimized regional convolutional neural network[J]. *Laser & Optoelectronics Progress*, 2020, 57(10): 101021.  
韩永赛, 马时平, 李帅, 等. 基于优化区域卷积神经网络的机场区域检测[J]. *激光与光电子学进展*, 2020, 57(10): 101021.
- [18] Liu W J, Wang F, Qu H C. Object detection model based on multi-scale feature integration[J]. *Laser & Optoelectronics Progress*, 2019, 56(23): 231007.  
刘万军, 王凤, 曲海成. 融合多尺度特征的目标检测模型[J]. *激光与光电子学进展*, 2019, 56(23): 231007.
- [19] Hong S J, Han Y, Kim S Y, et al. Application of deep-learning methods to bird detection using unmanned aerial vehicle imagery[J]. *Sensors*, 2019, 19(7): 1651.
- [20] Akçay H G, Kabasakal B, Aksu D, et al. Automated bird counting with deep learning for regional bird distribution mapping[J]. *Animals*, 2020, 10(7): 1207.
- [21] Chen Y Q, Sun L Q, Zhang Y Z, et al. Research on bird detection technology for electric transmission line based on YOLO v3[J]. *Computer Engineering*, 2020, 46(4): 294-300.  
陈咏秋, 孙凌卿, 张永泽, 等. 基于YOLO v3的输电线路鸟类检测技术研究[J]. *计算机工程*, 2020, 46(4): 294-300.
- [22] Hu J, Shen L, Albanie S, et al. Squeeze-and-excitation networks[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020, 42(8): 2011-2023.
- [23] Liu S T, Huang D, Wang Y H. Learning spatial fusion for single-shot object detection[EB/OL]. (2019-11-25)[2020-12-25]. <https://arxiv.org/abs/1911.09516>.
- [24] Zheng Z H, Wang P, Liu W, et al. Distance-IoU loss: faster and better learning for bounding box regression[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, 34(7): 12993-13000.
- [25] Loshchilov I, Hutter F. SGDR: stochastic gradient descent with warm restarts[DB/OL]. (2017-05-03)[2020-12-25]. <https://arxiv.org/abs/1608.03983>.
- [26] He K M, Zhang X Y, Ren S Q, et al. Deep residual learning for image recognition[C] //2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 770-778.
- [27] Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C] //2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 936-944.
- [28] Yu J H, Jiang Y N, Wang Z Y, et al. UnitBox: an advanced object detection network[C] //Proceedings of the 24th ACM international conference on Multimedia, October 15-19, 2016, Amsterdam, The Netherlands. New York: ACM, 2016: 516-520.