**Computer Science and Application** 计算机科学与应用, 2022, 12(2), 366-375
Published Online February 2022 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2022.122037

Hans汉斯

# 基于YOLOv5s的注意力改进研究

**彭章龙，余华平***

长江大学，湖北 荆州

收稿日期：2022年1月17日；录用日期：2022年2月14日；发布日期：2022年2月22日

## 摘　要

**随着时间的推进和硬件的不停发展，计算机的计算能力也得到了极大的提升，相应地，以算力为支持的深度学习也得到了飞速发展。作为深度学习的一个分支，目标检测算法的研究愈显突出。针对算法落地以及实时检测的要求，提出了基于YOLOv5s的注意力改进，在相同实验环境下，以不同的改进条件，将同一数据集输入给YOLOv5s训练和测试，通过tensorboard可视化结果得出，所提出的改进对YOLOv5s的准确率、召回率以及mAP有明显提升，对满足实际需求更近一步。**

## 关键词

**深度学习，目标检测，YOLOv5s，注意力**

# Research on Attention Improvement Based on YOLOv5s

**Zhanglong Peng, Huaping Yu***

Yangtze University, Jingzhou Hubei

Received: Jan. 17th, 2022; accepted: Feb. 14th, 2022; published: Feb. 22nd, 2022

## Abstract

**With the advancement of time and the continuous development of hardware, the computing power of computer has also been greatly improved. Accordingly, deep learning supported by computing power has also developed rapidly. As a branch of deep learning, the research of object detection algorithm is becoming more and more prominent. According to the requirements of algorithm landing and real-time detection, an attention improvement based on yoov5s is proposed. Under the same experimental environment and different improvement conditions, the same data set is**

*通讯作者。

文章引用: 彭章龙, 余华平. 基于 YOLOv5s 的注意力改进研究[J]. 计算机科学与应用, 2022, 12(2): 366-375.
DOI: 10.12677/csa.2022.122037

**input to YOLOv5s for training and testing. Through the tensorboard visualization results, it is concluded that the proposed improvement has significantly improved the accuracy, recall and mAP of YOLOv5s, which is a step closer to meeting the actual needs.**

## Keywords

**Deep Learning, Object Detection, YOLOv5s, Attention**



## 1. 引言

近年来，人工智能在算力的支持下得到了飞速发展。在计算机视觉的领域中，目标检测有着重要且广泛的应用，且一直处于火热的研究之中，甚至成为了其他研究领域的基础。面对不同场景的目标检测算法落地于安全监控[1]、交通[2]、医疗[3]甚至军事领域[4]，特别是异常火热的自动驾驶[5]，可见目标检测已经成为了基本。

利用深度学习的卷积神经网络提取到图像的特征信息，将物体分类的同时找出物体所在图像的具体位置，最后将其应用于实际生活中。在应用之前，需要保证模型的推理速度，以及精度，此方面的研究从未停止。Girshick 等[6] 2014 年利用 Region CNN(R-CNN)开启了深度学习在目标检测方面的研究，效果显著，引入了感兴趣区域和 CNN，使 mAP 值在 PASCAL VOC2007 上最好结果的 30%提升到 66.0%，但是精度离实际应用远远不够，检测速度更是达不到落地的要求。Girshick [7] 2015 年的 Fast R-CNN 将 R-CNN 中通过将选择性搜索算法得到的几千个候选框分别放入卷积网络改进为将一张完整的图像放进卷积网络，再得到每张图像的候选框，最后进行分类和回归，大幅减少每张图片耗时，在 PASCAL VOC2007 上 mAP 达到 70.0%。Girshick 等[8] 2016 年的 Faster R-CNN 使用 RPN 网络生成候选框，引入多尺度锚框来检测各种尺度的物体，最后的检测精度和速度明显得到提升，在 PASCAL VOC2007 数据集上 mAP 达到 73.2%。R-CNN 系列等[9] [10] [11] [12]两阶段目标检测算法，在拥有高精度检测效果的同时检测速度缓慢的问题依然突出。Redmon 等 2016 年的 YOLO (You Only Look Once)单阶段检测算法将物体检测问题归于回归问题，给卷积神经网络输入，最后得到边界框的信息以及置信度，在 mAP 为 63.4%的同时 FPS 达到 45，检测速度得到了极大提升，但是检测精度却比不上 Faster R-CNN 检测算法。YOLO 单阶段检测算法检测速度快是该算法能落地的优势，但是发展到现在存在的问题依旧是精度不足。

目标检测中的精度以及 FPS 一直以来都是研究的对象，在保证精度满足需求的情况下追求实时性是 YOLO [13] [14] [15] [16]算法在实际生活中得到广泛应用的条件，但是目前 YOLO 检测算法的检测精度依然存在着精度以及 mAP 不足的情况，提升目标检测算法的检测精度以及推理速度成为发展的必要。

本文以提高 Ultralytics 公司 2020 年开源的 YOLOv5s 检测算法的精度和 mAP 为目的，采取将注意机制引入 YOLOv5s 网络，相比于其他 3 个版本 YOLOv5m、YOLOv5l、YOLOv5x，YOLOv5s 体积更小，实验更方便，获取网络深层信息更容易，从而探索注意力机制对 YOLOv5s 检测算法的影响，以期提升 YOLOv5s 检测网络的检测精度和 mAP。

## 2. YOLOv5s 网络模型

### 2.1. YOLOv5s 简介

YOLOv5s 网络结构有四个部分，分别是输入端、Backbone、Neck、Output。整体结构如图 1 所示。

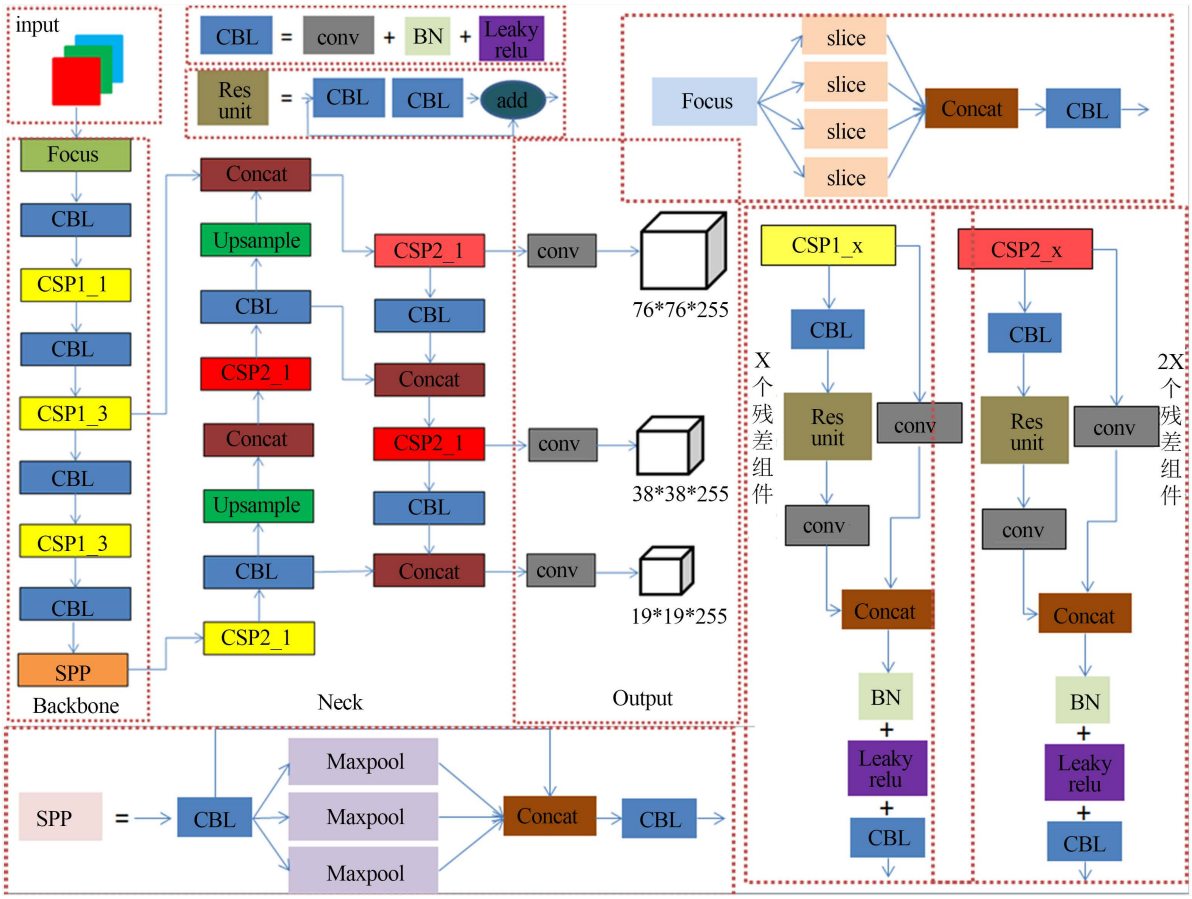


**Figure 1.** Model network structure of YOLOv5s
**图 1.** YOLOv5s 模型网络结构

小目标检测困难一直是目标检测算法中存在的问题，信息少的特点导致检测困难，加上图像模糊，分辨率低等各种问题导致小目标检测成为目标检测的难点。针对这种情况 YOLOv5s 网络结构的输入端进行了 Mosaic 数据增强，将四张图片进行随机裁剪，再拼接到一张图片上作为训练数据，丰富了图片背景，随机缩放增加更多的小目标，利于对小目标检测的同时变相提高了 BatchSize，进行 BN 操作的时候计算了四张图片，达到使网络的鲁棒性更好的目的，所以对自身的 BatchSize 不是很依赖，同时将图片统一尺寸为 460 × 460 × 3。YOLOv5s 给网络的 anchor 默认值为[10, 13, 16, 30, 33, 23]，[30, 61, 62, 45, 59, 119]，[116, 90, 156, 198, 373, 326]，网络会根据默认 anchor 训练得到预测框，再根据预测框和真实框的差距来调整模型的网络参数，直到得到最终的预测框。

在 Backbone 网络中，主要的结构有 Focus、CSP、SPP。Focus 结构，最重要的是切片操作，如 YOLOv5s 结构图中的 Focus 部分，460 × 460 × 3 的图片会被切片为 4 个 320 × 320 × 3 的特征图，具体操作是隔位提取像素，再在通道上拼接形成 320 × 320 × 12 的特征图，在计算量微量提升的情况下起到了提取更多特征的效果。CSP 结构借鉴了 CSPNet [17]，在进入 CSP 模块之前都会存在一个 3 × 3 的卷积核，步长为 2，

进行下采样。将特征图作两条路径处理，一条路径上经过 x 个残差组件，另一条路径上经过卷积，将经过两条路径后的特征图进行拼接，在不计算重复梯度信息轻量化的同时保证了准确率。SPP 部分则可以极大地增加感受野，以{1 × 1, 5 × 5, 9 × 9, 13 × 13}的方式进行最大池化，最后将得到的各种尺度特征图进行拼接，提取主干特征的能力得到加强，更加有效地分离最为重要的上下文特征。

在 Neck 模块中，采用了 FPN + PAN 的结构，并引入了 CSP2 结构。FPN 用来传递强语义特征信息，PAN 结构用来传递强定位特征信息，两者结合达到将不同检测层的信息进行融合，提高特征提取的能力。经过一系列卷积操作后进行特征融合，使特征图包含的特征信息更多。

在输出端，采用了 GIoU 作为边界框的损失函数和非极大值抑制(NMS)。GIoU 定义如公式(1)所示，IoU 定义如公式(2)所示,GIoU 示意图如图 2 所示。

$$\mathrm{GIoU} = \mathrm{IoU} - \frac{|A_c - U|}{A_c} \tag{1}$$

$$\mathrm{IoU} = \frac{A \cap B}{A \cup B} \tag{2}$$

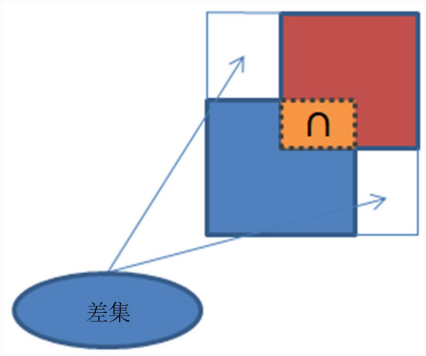


**Figure 2.** Schematic diagram of GIoU
**图 2.** GIoU 示意图

如上图所示，IoU 等于两个边界框相交部分的面积比上两个边界框的并集，当两个边界框无相交部分时就会出现没有梯度的情况，无法实现优化以及重叠方式的区分。GIoU 则先求出包含两个边界框最小的外接矩形面积，再求出不属于两个边界框的面积占外接矩形面积的比例，最后用 IoU 减去这个面积比例。

## 2.2. SE 模块

本文第一组对比实验借鉴了在 2017 年的 ImageNet 分类赛上夺冠的 SENet (Squeeze-and-Excitation Networks)，采用了其中的 SE 模块来学习通道上的相关性，在计算量上稍有提升的同时，筛选出在通道上关联性和重要性较突出的特征信息。第一步会对 SE 模块的输入进行压缩操作，经过全局平均池化之后的特征图会变成 1 × 1 × C 的向量，C 代表通道，第二步将 1 × 1 × C 的向量放进全连接神经网络中，全连接神经网络包含了两个全连接层，第一个全连接层的通道数会和一个缩放因子进行乘积，在经过第一个全连接层的时候会进行缩放，达到通过减少通过数而减少计算量的目的，第二个全连接层则使输出通道数为 C。最后的 Scale 操作将输入的特征图与经过压缩和激励操作的特征图进行通道权重相乘，得到输出，因为使用了全连接层，所以参数量会有所上升。本文将 SE 模块加在主干网络部分，SE 模块结构如图 3 所示。

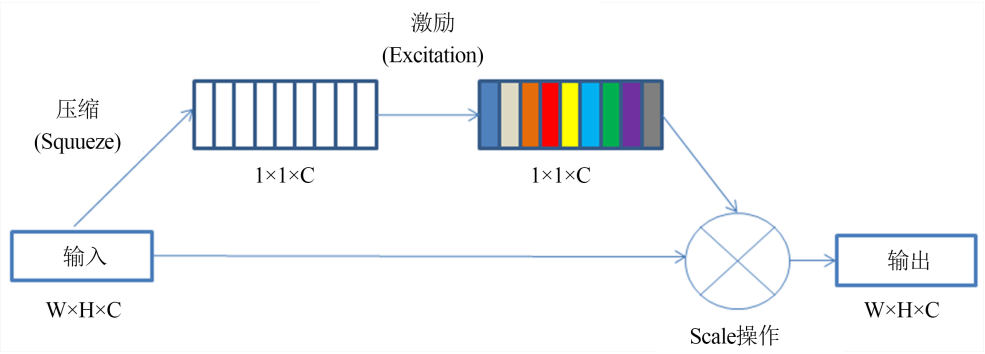


**Figure 3.** SE module
**图 3.** SE 模块

### 2.3. CBAM 模块

本文第二组对比实验采用了 CBAM (Convolutional Block Attention Module)。CBAM 模块分为两个部分，通道注意力部分和空间注意力部分，通道注意力的核心部分是使用 1 × 1 的卷积核来提取信息，偏置部分在实验时设置为 false，空间注意力部分则是通过分别在通道维度上采取求平均和，最后进行合并，合并结果为通道数是 2 的卷积层，再通过一个卷积得到。实验证明顺序使用通道注意力机制和空间注意力机制优于先使用空间注意力机制再使用通道注意力机制的组合和并行使用两个注意力机制的方式。将通道注意力部分和空间注意力部分进行结合，结合之后使用广播机制对原始特征图进行特征提取，得到 CBAM 的输出特征图。CBAM 模块增强了特征在通道和空间上的表现，使网络学习需要关注和摒弃不需要关注特征信息的能力得到提升，像人类视觉一样捕捉表现突出的信息，由于 CBAM 模块的效果是对特征信息采取精细化分配处理，所以本文会将 CBAM 模块加在主干网络部分。CBAM 结构如图 4 所示。

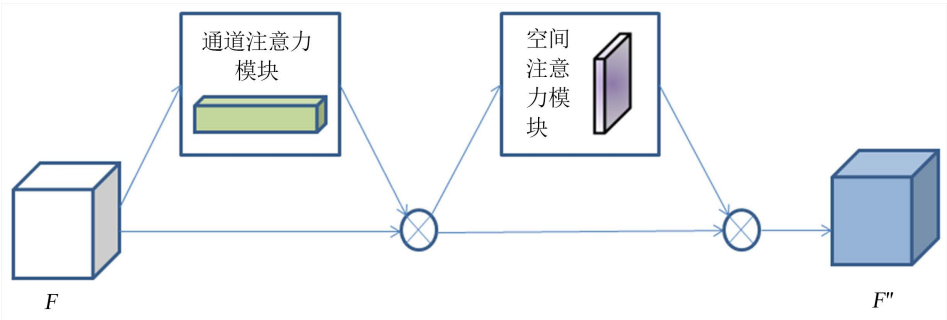


**Figure 4.** CBAM module
**图 4.** CBAM 模块

如上图所示，网络中间的特征图 $F \in R^{C\times H\times W}$ 作为 CBAM 层的输入，$F$ 经过一个共享的两层神经网络(通道注意力模块)之后与 $F$ 进行 element-wise 乘法得到新的特征图 $F' \in R^{C\times H\times W}$，$F'$ 会继续进入空间注意力模块(并行进入一个通道的最大池化和平均池化后再在通道上进行拼接)后与 $F'$ 进行 element-wise 乘法得到 $F''$，$F'' \in R^{C\times H\times W}$。

## 3. 实验及结果

### 3.1. 实验数据集及实验环境

实验数据集采用 PASCAL VOC2007，类别数为 20，并混合在学校采景的数据集进行扩充，为了保

持 PASCAL VOC 数据集的格式，对扩充的数据集用 LabelImg 进行目标标注，将标注好的文件以 PASCAL VOC 的命名规则命名，以 xml 作为标注好的文件后缀名，并使标注文件和图片名字一致。划分后的混合数据集如表 1 所示。

**Table 1.** Division of data sets
**表 1.** 数据集划分

| 数据集划分 | 数量(张) |
|---|---|
| 总数据集 | 8000 |
| 训练集 | 5600 |
| 测试集 | 2400 |

实验环境采用 Linux 操作系统，编程语言使用 Python 3.8，深度学习框架为 PyTorch，进行实验以及数据获取，详细实验环境如表 2 所示。

**Table 2.** Detailed configuration of experimental environment
**表 2.** 实验环境详细配置

| 实验环境 | 具体配置 |
|---|---|
| 操作系统 | Ubuntu 16.04 |
| 编程语言 | Python 3.8.5 |
| 处理器 | Intel(R) Core(TM) i5-9300H CPU @ 2.40GHz |
| GPU | NVIDIA GeForce GTX 1650 |
| GPU 加速环境 | CUDA11.3 |

### 3.2. YOLOv5s 模型的训练参数设置

训练使用了在 Coco 和 PASCAL VOC 上的 yolov5s 预训练模型，优化算法选用 Adam，实验所有参数保持一致，迭代数为 300 轮，BatchSize 为 16，Adam 的动量因子为 0.999，初始学习率为 0.001。学习率调整方式为：学习率的初始值为 0.001，先用线性插值对学习率进行预热，预热动量因子为 0.95，轮数为 5，再采用余弦退火算法调整。

### 3.3. 评价指标

本实验对比由原 YOLOv5s、YOLOv5s + SELayer、YOLOv5s + CBAM 形成，评价指标使用精度(Precision)，召回率(Recall)以及平均精度均值(Mean-Average-Precision, mAP)，相关计算方式如公式 3、公式 4 所示。

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

TP 代表实际为正样本的同时模型预测为正样本的数量，FP 代表实际为负样本的同时模型预测为正样本的数量，FN 代表实际为正样本的同时模型预测为负样本的数量，TN 代表实际为负样本同时预测为负样本的数量。

Precession 计算的是预测为正样本且预测正确的样本数量占所有预测为正样本数量的比例，Recall 计

算的是预测为正样本且预测正确的样本数量占所有实际为正确的样本数量。因为每张图像的类别可能不同，所以采用 mAP 值作为评价指标，根据每个类别的 Precision 值和 Recall 值可以分别绘制出各个类别的 P-R 曲线，AP 计算的是 PRC (Precision-Recall Curve)曲线下面积，mAP 计算的是所有类别 AP 的平均值，mAP 指标比较如图 5 所示。

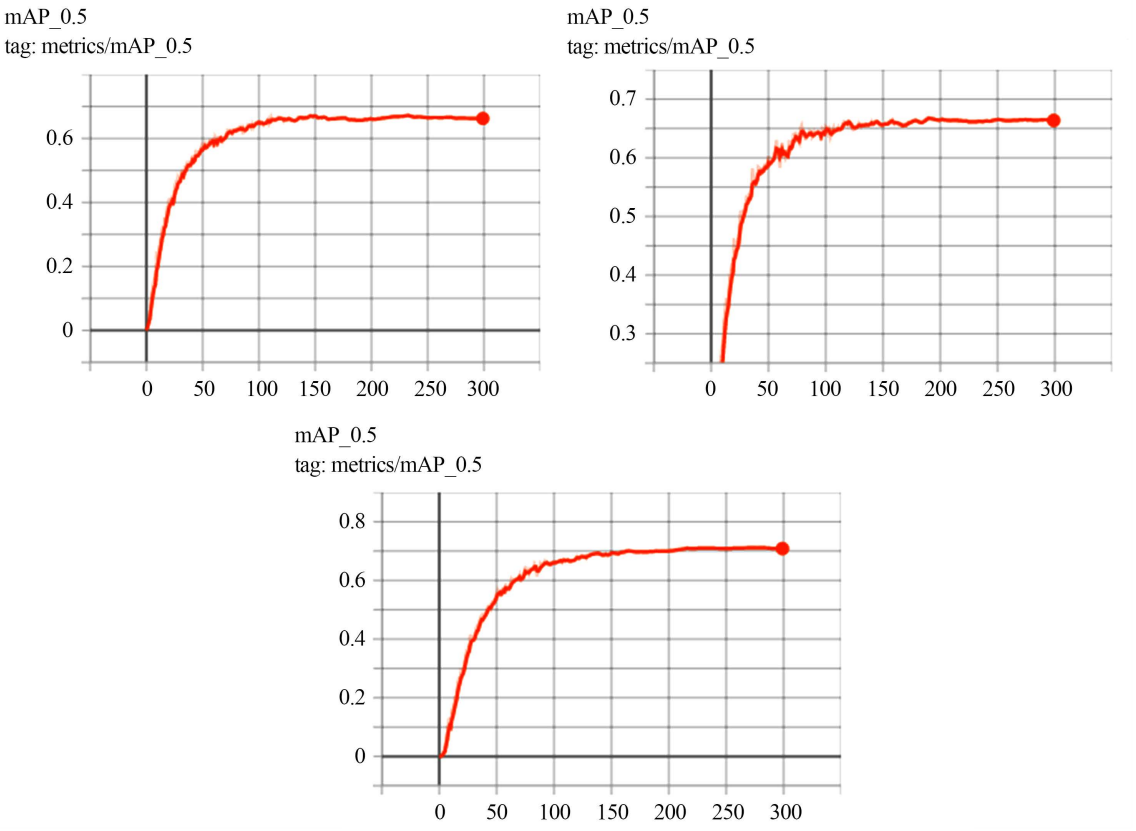


**Figure 5.** MAP values of the original YOLOv5s, YOLOv5s + SE module and YOLOv5s + CBAM module
**图 5.** 原 YOLOv5s、YOLOv5s + SE 模块和 YOLOv5s + CBAM 模块三个模型的 mAP 值

从三个模型训练所得到的 mAP 值可以得出，注意力机制使 YOLOv5s 模型计算微量增加的同时性能会得到一定提升。

## 3.4. 实验结果及分析

原 YOLOv5s、YOLOv5s + SE 模块和 YOLOv5s + CBAM 模块的性能对比结果如表 3 所示。

**Table 3.** Comparison of models performance
**表 3.** 模型性能对比

| 网络模型 | P/% | R/% | mAP/% |
|---|---|---|---|
| YOLOv5s | 70.15 | 62.63 | 66.39 |
| YOLOv5s + SE | 71.39 | 63.95 | 67.67 |
| YOLOv5s + CBAM | 75.23 | 66.93 | 71.04 |

表 3 的数据显示，与原 YOLOv5s 模型对比，加了注意力机制后，精度、召回率和 mAP 值均得到提

升。在 YOLOv5s 基础上加了 SE 模块后，精度提升 1.24%，召回率提升 1.32%，mAP 提升 1.28%；在 YOLOv5s 基础上加了 CBAM 模块后提升较大，精度提升 5.08%，召回率提升 4.3%，mAP 提升 4.65%。从模型的感受野出发，卷积神经网络都是采取级联的方式来逐步地增加感受野，但是当算法模型层数不足时，感受野的获取就会存在缺陷，采取注意力机制，这里使用了 SE 模块和 CBAM 模块，相当于级联了一定数据量的卷积操作让模型获取更大的感受野，从而使模型的性能得到提升。SE 模块的压缩操作得到的特征图相当于全局感受野，从而使 YOLOv5s 模型的性能得到略微提升，较于在分类任务上显著的效果，应用于目标检测任务上时，效果不太显著，可能因为感受野的原因，得到提升的效果，小目标的特点又导致(小目标在检测任务上难于分类任务)提升效果不太显著。CBAM 模块的通道注意力部分比 SE 模块多了一个最大池化并行操作的同时利用特征图的空间关系得到空间注意力特征图，对比于通道注意力，空间注意力会使模型重视特征信息的空间位置，利于检测任务的定位，两个部分串联后利于模型性能的提升，较于 SE 模块，在目标检测任务上 CBAM 模块优于 SE 模块。

算法测试对比效果如图 6，图 7，图 8 所示，

**Figure 6.** Detection effect of original YOLOv5s
**图 6.** 原 YOLOv5s 检测效果

**Figure 7.** Detection effect of YOLOv5s + SE module
**图 7.** YOLOv5s + SE 模块检测效果

**Figure 8.** Detection effect of YOLOv5s + CBAM
**图 8.** YOLOv5s + CBAM 检测效果

从图 6 可知，原 YOLOv5s 检测出现了检测错误的情况(墙体上“大”字下面)，从图 7 可知，在 YOLOv5s 加上 SE 模块改进后，检测错误的情况消失，从图 8 可知在 YOLOv5s 加上 CBAM 模块改进后，检测错误消失的同时置信度得到提升。

## 4. 结论

本文对目前检测效果较好且轻量的 YOLOv5s 进行了注意力机制的改进，通过将改进前与改进后的评价指标进行对比，得出注意力机制在增加微量计算的同时可以提升模型的检测性能的结论，实验检测效果的对比也证明了注意力机制可以改善模型对物体的检测。综上所述，本文所提出的改进方法可以提升模型的性能，但是存在实验环境和条件的限制，召回率低的问题没有解决，最终也未能将本实验所测试的算法部署、落地测试，后续会从数据集、nms 以及注意力机制优化等方向继续改进。

## 参考文献


[1] 高明, 左红群, 柏帆, 田清阳, 葛志峰, 董兴宁, 甘甜. 融合视觉关系检测的电力场景自动危险预警[J]. 中国图象图形学报, 2021, 26(7): 1583-1593.

[2] 李厚杰, 王法胜, 贺建军, 周瑜, 李威, 窦宇轩. 基于伪样本正则化 Faster R-CNN 的交通标志检测[J]. 吉林大学学报(工学版), 2021, 51(4): 1251-1260.

[3] 管子玉. 人工智能赋能智慧医疗[J]. 西北大学学报(自然科学版), 2021, 51(1): 1-32.

[4] 王瑶, 胥辉旗, 姜义, 张鑫. 基于深度学习的舰船目标检测技术发展综述[J]. 飞航导弹, 2021(2): 76-81.

[5] 张新钰, 邹镇洪, 李志伟, 刘华平, 李骏. 面向自动驾驶目标检测的深度多模态融合技术[J]. 智能系统学报, 2020, 15(4): 758-771.

[6] Girshick, R., Donahue, J., Darrell, T. and Malik, J. (2014) Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. 2014 *IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, 23-28 June 2014, 580-587. https://doi.org/10.1109/CVPR.2014.81 https://arxiv.org/abs/1311.2524.

[7] Girshick, R. (2015) Fast R-CNN. 2015 *IEEE International Conference on Computer Vision*, Santiago, 7-13 December 2015, 1440-1448. https://doi.org/10.1109/ICCV.2015.169 https://arxiv.org/abs/1504.08083

[8] Ren, S.Q., He, K.M., Girshick, R. and Sun, J. (2016) Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. https://arxiv.org/abs/1506.01497

[9] 吴雪, 宋晓茹, 高嵩, 陈超波. 基于深度学习的目标检测算法综述[J]. 传感器与微系统, 2021, 40(2): 4-7+18.

[10] 张泽苗，霍欢，赵逢禹. 深层卷积神经网络的目标检测算法综述[J]. 小型微型计算机系统，2019，40(9): 1825-1831.

[11] 方路平，何杭江，周国民. 目标检测算法研究综述[J]. 计算机工程与应用, 2018, 54(13): 11-18+33.

[12] Rajeshwari, P., Abhishek, P., Srikanth, P. and Vinod, T. (2019) Object Detection: An Overview. *International Journal of Trend in Scientific Research and Development*, **3**, 1663-1665. https://doi.org/10.31142/ijtsrd23422

[13] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2016) You Only Look Once: Unified, real-Time Object Detection. 2016 *IEEE Conference on Computer Vision and Pattern Recongnition*, Las Vegas, 27-30 June 2016, 779-788. https://doi.org/10.1109/CVPR.2016.91

[14] Redmon, J. and Farhadi, A. (2017) YOLO9000: Better, Faster, Stronger. *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, 21-26 July 2017, 6517-6525. https://doi.org/10.1109/CVPR.2017.690

[15] Redmon, J. and Farhadi, A. (2018) YOLOv3: An Incremental Improvement. https://arxiv.org/abs/1804.02767

[16] Bochkovskiy, A., Wang, C.Y. and Liao, H.M. (2020) YOLOv4: Optimal Speed and Accuracy of Object Detection. https://arxiv.org/abs/2004.10934

[17] Wang, C.Y., Liao, H.M., Yeh, I., Wu, Y.-H., Chen, P.-Y. and Hsieh, J.-W. (2019) CSPNet: A New Backbone That Can Enhance Learning Capability of CNN. 2020 *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, Seattle, 14-19 June 2020, 1571-1580. https://doi.org/10.1109/CVPRW50498.2020.00203 https://arxiv.org/abs/1911.11929