第 59 卷 第 18 期/2022 年 9 月/激光与光电子学进展 研究论文

激光与光电子学进展 先进成像

基于改进YOLOv4算法的室内场景目标检测

李维刚*，杨潮，蒋林，赵云涛

武汉科技大学冶金自动化与检测技术教育部工程研究中心，湖北 武汉 430081

摘要 针对传统方法在室内场景目标检测中存在检测精度低、检测速度慢等问题，提出一种改进的YOLOv4算法模型。构建室内场景目标检测数据集，使用K-means++聚类算法优化先验框参数，提高先验框与目标的匹配度；调整原始YOLOv4的网络结构，将跨阶段局部网络结构融入模型颈部网络中，消除在特征融合阶段梯度反向传播导致的梯度信息冗余现象，提高对室内目标的检测能力；引入深度可分离卷积模块，取代模型中原有的3×3卷积层，减少模型参数，提升检测速度。实验结果表明，改进后的YOLOv4算法在室内场景目标检测数据集上的平均精度达83.0%，检测速度达72.1 frame/s，较原始YOLOv4算法，分别提高了3.2个百分点和6 frame/s，同时模型规模缩小了36.3%，优于目前其他基于深度学习的室内场景目标检测算法。

关键词 目标检测；室内场景；YOLOv4；跨阶段局部网络；深度可分离卷积

中图分类号 TP391.4 **文献标志码** A **DOI**：10.3788/LOP202259.1815003

Indoor Scene Object Detection Based on Improved YOLOv4 Algorithm

Li Weigang*, Yang Chao, Jiang Lin, Zhao Yuntao

Engineering Research Center for Metallurgical Automation and Measurement Technology, Ministry of Education, Wuhan University of Science and Technology, Wuhan 430081, *Hubei, China*

Abstract In this paper, we proposed an improved YOLOv4 algorithm model to solve the problems of low detection accuracy and slow detection speed of traditional indoor scene object detection methods. First, we constructed an indoor scene object detection dataset. Then, we applied the K-means++ clustering algorithm to optimize the parameters of the priori box and improve the matching degree between the priori box and object. Next, we adjusted the network structure of the original YOLOv4 model and integrated the cross stage partial network architecture into the neck network of the model. This eliminates the gradient information redundancy phenomenon caused by the gradient backpropagation in the feature fusion stage and improves the detection ability for indoor targets. Furthermore, we introduced a depthwise separable convolution module to replace the original 3×3 convolution layer in the model to reduce the model parameters and improve the detection speed. The experimental results show that the improved YOLOv4 algorithm achieves an average accuracy of 83.0% and a detection speed of 72.1 frame/s on the indoor scene target detection dataset, which is 3.2 percentage points and 6 frame/s higher than the original YOLOv4 algorithm, respectively, additionally, the model size is reduced by 36.3%. The improved YOLOv4 algorithm outperforms other indoor scene object detection algorithms based on deep learning.

Key words object detection; indoor scene; YOLOv4; cross stage partial network; depthwise separable convolution

1 引 言

室内场景与人们生活息息相关。室内场景目标检测技术是计算机视觉领域的热门研究对象之一，关键是确定目标的位置、大小及类别，在诸如室内移动机器人精准定位与导航、视障人士辅助导航、室内安防监控系统设计等任务中具有重要研究意义[1-2]。不同于其他环境，室内场景结构复杂，目标元素丰富多样，受光线、角度和目标相互遮挡等因素的影响，传统目标检测算法在检测精度、速度和移动设备的部署问题上难以满足实际应用需求。因此，室内场景目标检测算法的设计成为一种极具挑战性的任务。

收稿日期：2021-06-18；**修回日期**：2021-07-08；**录用日期**：2021-07-20

基金项目：国家重点研发计划(2019YFB1310000)、湖北省揭榜制科技项目(2020BED003)、湖北省重点研发计划(2020BAB098)

通信作者：*liweigang. luck@foxmail. com

传统方法大多通过人工构建特征因子来进行室内场景的目标检测任务[3-4]，然而由于人工设计特征表达能力的局限性和计算资源的限制，这类方法在检测精度、速度及鲁棒性方面表现较差。近年来，随着大数据、计算机技术和深度学习的迅速发展[5]，基于卷积神经网络（CNN）的目标检测算法在精度和速度上得到了极大改善，并逐渐成为主流方法[6]。文献[7]基于视觉同步定位与建图（SLAM）系统和神经网络的目标检测算法，提出一种构建复杂几何信息的语义地图方法；文献[8]研究了一种基于深度卷积神经网络的室内目标检测系统，该系统提高了模型的检测性能，但未考虑目标相互遮挡的问题；文献[9]提出一种基于循环卷积神经网络的检测算法，该算法提高了对室内目标的检测精度，但检测速度有所降低。

现阶段，基于深度学习的目标检测算法可分为两类：Two-stage 和 One-stage[10]。Two-stage 类的代表算法有 Faster-RCNN[11]、Mask-RCNN[12]，这类算法基于区域候选网络（RPN）和神经网络分类的理论，检测过程分成两个阶段，先利用 RPN 生成预选框，再通过检测网络实现预选框的分类和回归，这类算法的准确率较高，但检测速度较慢。One-stage 类代表算法有 SSD[13]、YOLO 系列[14-16]，这类算法去掉了 RPN 阶段，可直接获得检测结果，因而检测速度较快，但准确率较低。YOLO 系列中最新的 YOLOv4 算法[17]的检测速度和精度较为平衡，是目前工业应用中最广泛的目标检测算法。

在实际应用中，室内场景目标检测算法必须满足良好的实时性和准确性要求，因此本文选择目前检测速度与精度较为平衡的 YOLOv4 算法作为研究对象。为了进一步提高室内场景中对目标检测的速度和准确率，并使模型更容易部署在移动设备上，提出一种改进 YOLOv4 的室内场景目标检测算法。在先验框聚类过程中，使用 K-means＋＋算法代替原始的 K-means 算法，摆脱对初始化聚类中心的依赖，获取与目标拟合度更高的先验框参数；模型颈部添加跨阶段局部网络（CSPNet）结构[18]，增强特征融合能力，提高检测精度；为了使模型更容易部署在小型嵌入式平台上，同时提高检测速度，使用深度可分离卷积（depthwise separable convolution）代替原始 YOLOv4 中的部分 3×3 标准卷积，降低算法的参数量与计算量。实验结果表明：改进后的 YOLOv4 算法对室内场景目标的检测速度和精度均高于其他算法，可以更好满足实际应用需求。

2　YOLOv4 算法

2.1　算法原理

YOLOv4 的网络结构主要包括四个部分：输入（Input）、主干（Backbone）、颈部（Neck）和头部（Head），网络结构如图 1 所示。

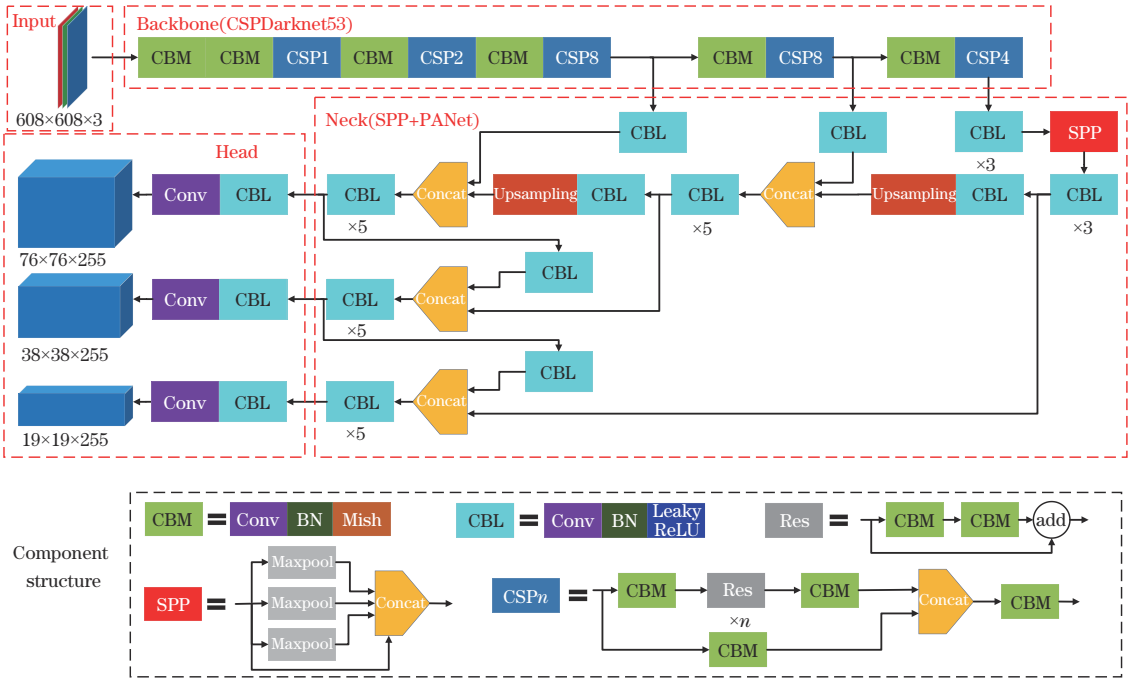

图 1　YOLOv4 网络结构图

Fig. 1　Diagram of YOLOv4 network structure

输入端，YOLOv4 采用自对抗训练（SAT）和 Mosaic 的数据增强方式，增强网络的鲁棒性。通过 SAT 策略，神经网络反向更新图像，在添加扰动后的图像上进行训练，实现数据扩充。通过 Mosaic 方式，4 张图片随机缩放、剪裁、排列合成一张图片，增加样本中的小目标，并缓解 GPU 压力。

主干网络上，YOLOv4采用CSPDarknet53代替YOLOv3中的Darknet53，引入CSPNet结构降低网络计算量，消除网络反向优化时梯度信息冗余现象，增强卷积网络学习能力，在实现网络轻量化的同时保证准确率。采用Mish激活函数，增强深层信息的传播。

颈部网络采用空间金字塔池化（SPP）模块[19]和路径聚合网络（PANet）[20]。SPP网络对特征层进行1×1、5×5、9×9、13×13四种尺度的最大池化（Maxpool）操作，有效提高网络的感受野，并提取出重要的上下文特征。PANet是对特征金字塔网络（FPN）[21]的进一步改进，在FPN的基础上又添加了一个自下而上的路径增强（bottom-up path augmentation）结构，避免了在传递过程中出现浅层信息丢失的问题，提高了网络预测的准确性。

头部网络用于回归和分类，与YOLOv3一样，YOLOv4沿用多尺度预测的方式，输出3个不同大小的feature map，分别检测小、中、大3种目标。通过K-means算法对样本目标进行聚类，得出先验框大小，在此基础上利用相对偏移量计算出预测框的大小和位置。

检测时，先调整输入图像的分辨率（以608×608为例），经过特征提取和特征融合后，图像被划分为$S\times S$个网格，S取值为19、38或76，每个网格单元对应3个预测框，每个预测框附带5种信息（2个中心点坐标偏移量、2个边框大小偏移量和1个目标置信度），同时每个预测框还包含目标属于每种类别的概率。如图2所示，斜线框为目标中心点所在的网格，虚线框为真实框，实线框为预测框，n_{class}指样本的类别数。预测时，通过非极大值抑制（NMS）算法剔除同一目标的多个预测框，只保留最佳的预测框。

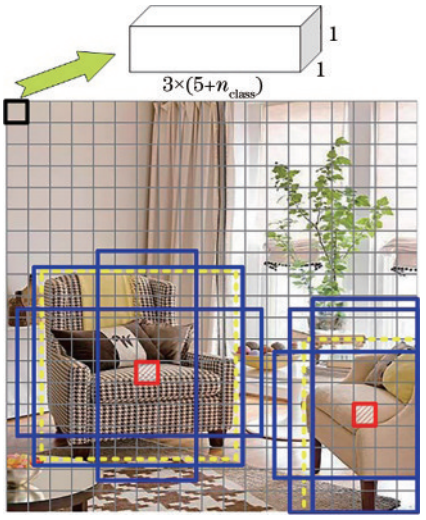

图2　YOLOv4算法在19×19单元格中的预测框

Fig. 2　Prediction box of YOLOv4 algorithm in 19×19 cells

2.2　损失函数

YOLOv4算法的损失函数由预测框回归误差、置信度误差和分类误差3部分组成，其中预测框回归误差采用complete intersection over union（CIoU）损失，置信度误差和分类误差采用交叉熵损失。总损失函数为

$$L=\sum_{i=0}^{S^2}\sum_{j=0}^{B}I_{ij}^{\text{obj}}\left[1-R_{\text{IoU}}+\frac{\rho^2\left(b,b^{\text{gt}}\right)}{m^2}+\alpha\nu\right]+\sum_{i=0}^{S^2}\sum_{j=0}^{B}\left(I_{ij}^{\text{obj}}+\lambda_{\text{noobj}}I_{ij}^{\text{noobj}}\right)\left[\hat{C}_i^j\log C_i^j+\left(1-\hat{C}_i^j\right)\log\left(1-C_i^j\right)\right]+$$
$$\sum_{i=0}^{S^2}I_{ij}^{\text{obj}}\sum_{c\in c_{\text{class}}}\left[\hat{P}_i^j\log P_i^j+\left(1-\hat{P}_i^j\right)\log\left(1-P_i^j\right)\right],\tag{1}$$

$$\alpha=\frac{\nu}{\left(1-R_{\text{IoU}}\right)+\nu},\tag{2}$$

$$\nu=\frac{4}{\pi^2}\left(\arctan\frac{w^{\text{gt}}}{h^{\text{gt}}}-\arctan\frac{w}{h}\right)^2,\tag{3}$$

式中：α为权重函数；ν用于度量长宽比的相似性；S^2为网格数；B为每个网格的先验框个数；I_{ij}^{obj}、I_{ij}^{noobj}分别表示预测的网格是、否包含目标；R_{IoU}和$\rho(b,b^{\text{gt}})$分别表示预测框与真实框的交并比和中心点的欧氏距离；m为预测框和真实框围成的最小封闭框的对角线距离；λ_{noobj}为权重参数，用于平衡正负样本；C_i^j与$\hat{C}_i^j$表示预测框与真实框的置信度；P_i^j与$\hat{P}_i^j$表示预测框与真实框的类别概率；b、w和h分别表示预测框的中心点坐标、宽和高；b^{gt}、w^{gt}和h^{gt}分别表示真实框的中心点坐标、宽和高。

3　改进的YOLOv4算法

3.1　网络结构改进

对原始YOLOv4的颈部和主干网络进行改进，改进后的YOLOv4网络结构如图3所示。对于颈部网络，将CSPNet结构思想运用到SPP和PANet模块中，分别记为CSPSPP和CSPPAN；对于颈部和主干网络，同时修改CSPNet结构中的残差模块，将残差模块中的3×3标准卷积替换为深度可分离（DS）卷积，改进后，主干网络记为DS-CSPDarknet53，颈部网络中的CSPPAN模块记为DS-CSPPAN。

3.1.1　颈部网络中融合跨阶段局部网络结构

在神经网络的推理过程中，往往由于网络优化过程中的梯度信息重复，计算量剧增。CSPNet将基础层的特征映射X_0分成两个部分X_0'和X_0''，X_0'直接连接

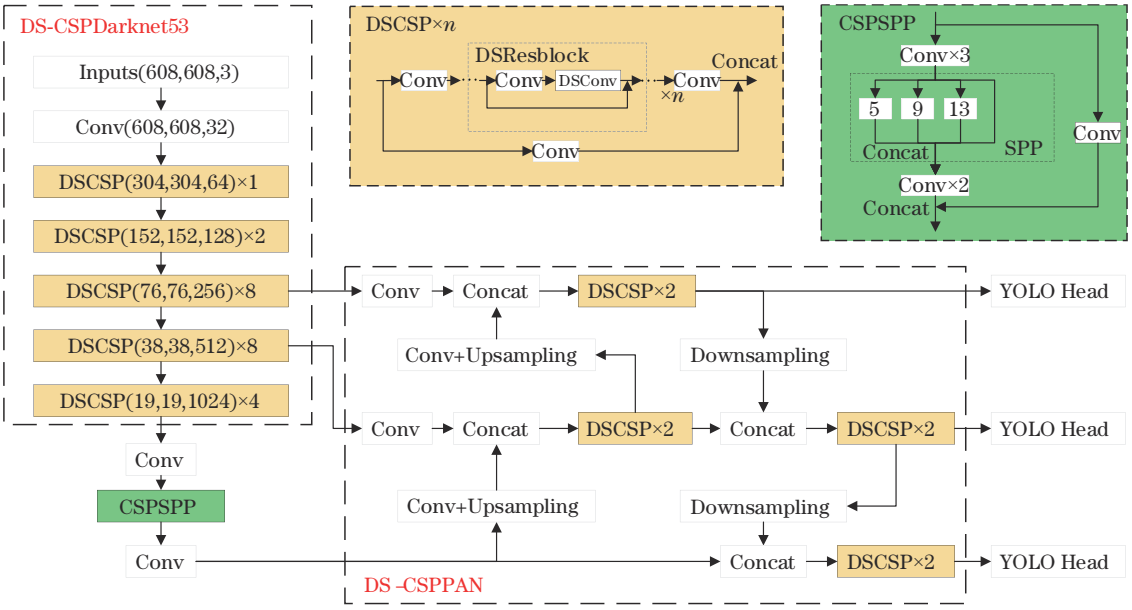

图 3　改进的 YOLOv4 网络结构图

Fig. 3　Diagram of improved YOLOv4 network structure

到末尾的局部过渡层，X_0''则先通过多个局部密集层，再进入局部过渡层与X_0'拼接。在第一个局部过渡层中，输入为$\left[X_0'', X_1, X_2, \cdots, X_{k-1}\right]$，输出为$X_{\mathrm{T}}$，$X_k$为第$k$个密集层的输出，第二个局部过渡层的输入为$X_0'$和$X_{\mathrm{T}}$的拼接值，输出为$X_{\mathrm{U}}$。则整个前馈方程可记为

$$\begin{cases} X_k = w_k * \left[X_0'', X_1, X_2, \cdots, X_{k-1}\right] \\ X_{\mathrm{T}} = w_{\mathrm{T}} * \left[X_0'', X_1, X_2, \cdots, X_k\right] \\ X_{\mathrm{U}} = w_{\mathrm{U}} * \left[X_0', X_{\mathrm{T}}\right] \end{cases}, \tag{4}$$

式中：$*$代表卷积操作；w为第k个密集层的权重。在反向更新权重时，权重方程可写为

$$\begin{cases} w_k' = f(w_k, g_0'', g_1, g_2, \cdots, g_{k-1}) \\ w_{\mathrm{T}}' = f(w_{\mathrm{T}}, g_0'', g_1, g_2, \cdots, g_k) \\ w_{\mathrm{U}}' = f(w_{\mathrm{U}}, g_0', g_{\mathrm{T}}) \end{cases}, \tag{5}$$

式中：f为更新权重的函数；g_k表示第k个密集层的梯度。由权重方程式(5)可知，权重w_{T}'和w_{U}'是通过不同梯度信息分开集成的，这样处理的优势是既保留了原始网络特征重用的特点，又通过截断梯度流的方法防止梯度信息冗余。

CSPNet易于实现，且具有很好的通用性，可以构造在ResNet、ResNeXt和DenseNet等网络体系中[22]。YOLOv4在Darknet53网络中添加了CSPNet结构，形成的CSPDarknet53网络既减少了网络参数，又提高了检测精度。本文进一步调整YOLOv4模型结构，将CSPNet结构体系融入SPP模块和PANet中的连续卷积模块中，以此提高目标检测效果。

图4为SPP模块改进后的结构图，图5为连续卷积模块改进后的结构图，其中c为输入特征图的通道数，在融合多尺度特征前，将网络分成两个部分，一部

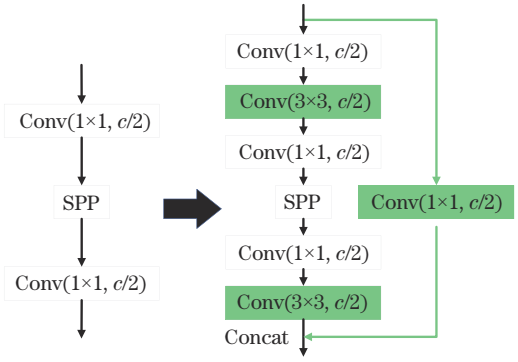

图 4　SPP模块CSPNet结构化

Fig. 4　CSPNet structured in SPP module

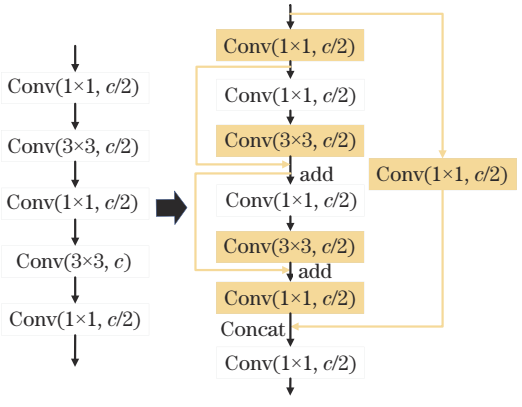

图 5　连续卷积模块CSPNet结构化

Fig. 5　CSPNet structured in continuous convolution module

分特征经过捷径连接，直接与SPP融合后的特征合并。在PANet中，深层特征与浅层特征张量拼接后会经历5个连续卷积层，考虑梯度消失和梯度爆炸的问题，将连续卷积层改为两个连续的残差块，再通过捷径

连接将这两个残差块包围起来，构成CSPNet结构。

3.1.2　引入深度可分离卷积

深度神经网络往往依靠千万级数量的网络参数获得优越的性能，网络结构越复杂，模型层数越深，存储空间和计算损耗的压力也越来越大。在工业实时应用过程中，不能只考虑网络的性能，还要解决大规模神经网络模型在嵌入式设备上的部署问题。目前，对神经网络轻量化已有许多研究，例如网络剪枝[23]、权重量化[24]、知识蒸馏[25]和人工设计轻量化网络[26]等。从人工设计轻量化网络结构的角度出发，引入MobileNet[27]所提出的深度可分离卷积，将其融合在YOLOv4网络中，替换原本大小为3×3、步长为1的标准卷积层，更好地压缩了模型参数量，使模型更容易部署到嵌入式设备上，并且提高了模型的运算速度。

深度可分离卷积由逐深度卷积（depthwise convolution）和逐点卷积（pointwise convolution）组合而成，如图6所示。逐深度卷积将卷积核拆分为单通道形式，在输入特征图像通道数不变的情况下对每个通道进行卷积操作。逐点卷积就是标准的1×1卷积，用于改变特征图像的维度。假设要将通道数为c_1、大小为$w\times h$的特征图像转换到c_2维，在$k_s\times k_s$大小的标准卷积中，参数量为$c_1c_2k_s^2$，计算量为$c_1c_2whk_s^2$；采用深度可分离卷积后的参数量为$c_1k_s^2+c_1c_2$，计算量为$c_1whk_s^2+c_1c_2wh$，降低到标准卷积的$1/c_2+1/k_s^2$。通常对于3×3大小的标准卷积核，改用深度可分离卷积后可使运算量下降到原来的11.1%～12.5%。

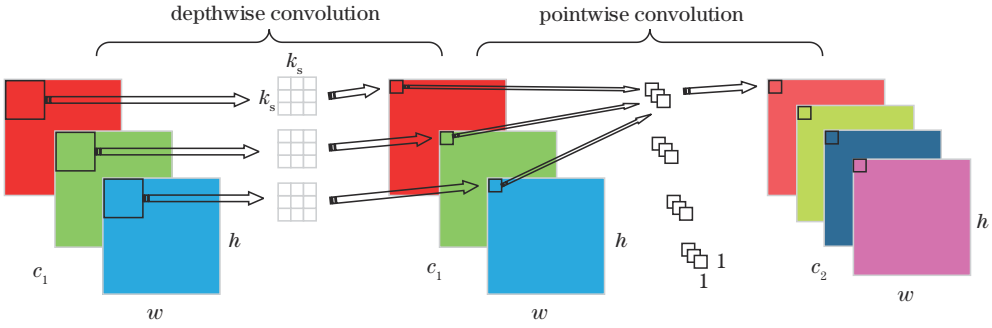

图6　深度可分离卷积

Fig. 6　Depthwise separable convolution

然而，使用深度可分离卷积后，网络的卷积层数有所加深，当网络层数过深时，在反向传播的过程中，一些神经元可能出现梯度消失或梯度爆炸现象，导致网络性能降低，影响检测准确率。为弥补这一问题，在所设计的深度可分离卷积模块中，逐深度卷积和逐点卷积后都增加了批归一化（BN）层和Mish激活函数。BN层可以加快网络收敛，抑制梯度消失或梯度爆炸的情况，Mish激活函数比Leaky ReLU激活函数的曲线更平滑，使深层网络中信息传递的效率更高。

3.2　K-means++聚类算法

YOLOv4属于anchor-based目标检测算法，此类算法中，先验框参数的设置对模型的预测性能至关重要，若先验框大小与目标尺寸不匹配，则会导致训练时正样本个数减少，造成漏检或误检的情况。

YOLOv4通过K-means算法对训练集中的所有检测目标进行聚类，聚类中心设置为9，按照大小顺序将9个聚类结果分配到3个不同尺度的feature map上作为先验框，不同尺度feature map上的先验框必须具有明显的大小差距，以便于检测不同大小的目标。但是，K-means算法依赖初始化聚类中心，不同的初始聚类中心可能导致完全不同的聚类结果，继而影响检测精度。

为了摆脱K-means算法对初始化聚类中心的依赖，消除模型出现局部最优的隐患，采用K-means++算法稳定初始化聚类中心的选择，使模型获得最优的先验框参数，提高模型检测的精度。K-means++算法的具体实现步骤如下：

1）随机从P个样本中选取一个样本作为初始聚类中心o_1；

2）对于$p=1,2,3,\cdots,P$，计算每个样本x_p与当前已有聚类中心的最短距离$d_{p\min}$，并计算x_p作为下一聚类中心的概率：$P(x_p)=d_{p\min}/\sum_{p=1}^{P}d_{p\min}$；

3）选取概率值最大的样本作为下一个聚类中心，重复步骤2），直到K个聚类中心都初始化完成；

4）计算每个样本x_p与当前已有聚类中心的距离d_p'，并将其分到距离与之最近的聚类中心所属的簇中；

5）对于$j=1,2,3,\cdots,K$，在每个簇中更新聚类中心：$o_j=\sum_{x\in o_j}\frac{x}{|o_j|}$；

6）重复步骤4）、5），直到聚类中心位置不再改变。

在上述算法中，样本指训练集中每个真实框的宽、高，聚类中心指先验框的尺寸。一般聚类算法采用欧氏距离来计算样本与聚类中心的距离，但当真实框的尺寸差异较大时，这种方法产生的误差较大。本文沿用YOLOv4中的方式，采用样本与聚类中心的最大交

并比来计算二者的差距，计算表达式为

$$d=1-\frac{S_{box}\cap S_{cen}}{S_{box}\cup S_{cen}},\tag{6}$$

式中：S_{box}代表样本（真实目标框）所表示的区域面积；S_{cen}表示聚类中心（先验框）所表示的区域面积。

4 实验与结果分析

实验使用的硬件配置为 Intel(R) Core i7-9700K CPU，NVIDIA Geforce RTX 2080 Ti 显卡，Windows 操作系统，软件环境为 CUDA11.0，Cudnn8.0，采用 Pytorch1.7 深度学习框架。训练时采用 Adam 优化器，初始学习率为 0.001，动量大小为 0.9，Batchsize 设为 8，迭代次数为 500，采用 Mosaic 数据增强技巧和 Dropblock 正则化方式。

为了使模型能够适应多种室内环境，实验构建了一个复杂背景的室内场景目标检测数据集，使用的图像一部分来自网上，另一部分由实际拍摄获得。为了提高室内场景目标检测的挑战性，选取了不同光照强度与角度的图像，并且图像中目标普遍存在相互遮挡的特点，以此增强模型对室内目标检测的鲁棒性。利用 LabelImg 软件人工标定标签，生成 xml 格式文件记录所有目标的位置、大小和类别信息。数据集共包含 9 类室内物体，分别为柜子（cabinet）、椅（chair）、桌（table）、沙发（sofa）、马桶（closestool）、门（door）、冰箱（refrigerator）、洗衣机（washer）及床（bed），目标分布在卧室、客厅、走廊、厨房、浴室、办公室等不同室内场景中，图片为 JPG 格式，大小均为 640×480，共计 7700 张。将数据按 6∶2∶2 随机分为训练集、验证集、测试集。训练前，将输入图像大小调整为 608×608，仿照 PASCAL VOC 数据集格式，对标注信息的边界框宽、高和中心点坐标进行归一化处理，以减小异常样本对数据的影响。

4.1 评价指标

检测过程中，利用预测结果与实际目标的 IoU 值衡量目标位置是否被成功预测，IoU 阈值设置为 0.5，即 IoU 大于 0.5 记作预测正确，否则记为预测错误。利用平均精度均值（mAP）作为网络的精度指标，表达式为

$$P_{mAP}=\frac{1}{n}\sum_{t=1}^{n}P_{AP}(t),\tag{7}$$

式中：n为类别数；P_{AP}为平均精度，定义为不同召回率下精确率的均值，用于评价样本中某一类的检测精度。P_{AP}的表达式为

$$P_{AP}=\int_0^1 P(R)\mathrm{d}R,\tag{8}$$

式中：P表示精确率，指所有检测出的目标中检测正确的概率；R表示召回率，指所有正样本中正确检测的概率。精确率和召回率的计算公式分别为

$$P=\frac{N_{TP}}{N_{TP}+N_{FP}},\tag{9}$$

$$R=\frac{N_{TP}}{N_{TP}+N_{FN}},\tag{10}$$

式中：N_{TP}指预测正确的正样本数；N_{FP}指预测为正样本但实际为负样本的数量；N_{FN}指预测为负样本但实际为正样本的数量。

在实际应用中，网络往往要部署在移动设备上，所以网络的规模和检测速度是不可忽视的。网络的规模由参数量或权重大小决定，检测速度由 FPS（frames per second）决定，FPS 定义为每秒能够检测的图片数量。

4.2 先验框参数聚类

原始 YOLOv4 的先验框参数是在 COCO 数据集上得到的，与本实验所使用的数据集差别较大，因此有必要重新计算先验框。根据样本的数据分布，使用 K-means 算法和 K-means++算法分别重新生成 9 个聚类中心作为先验框参数，真实框和先验框的大小分布如图 7 所示，图中“·”表示样本中真实框的分布，“+”和“×”分别表示 K-means 算法和 K-means++算法得到的先验框分布。可以看出，K-means 算法生成的聚类中心相对紧凑，例如中间和左上角两对点间的距离非常近，这是初始化中心选取不佳导致的，K-means++算法则可以消除这一现象。图 7 中还显示样本中大多数目标尺寸较小，因此在左下角区域分布密集，K-means++算法生成的聚类中心在密集区域分布较多，这样可以增加小目标的辨识度，提高检测准确率。最终 anchor 参数取值为（79，139），（131，174），（113，288），（202，222），（183，390），（310，269），（462，331），（311，510），（516，532）。

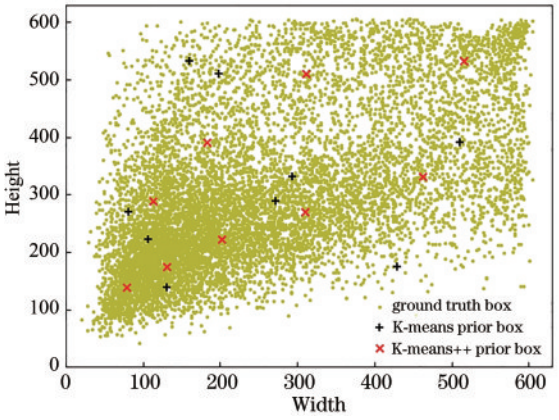

图 7 样本真实框与先验框大小分布散点图

Fig. 7 Scatter plot of the size distribution of sample ground truth box and prior box

将 K-means 算法与 K-means++算法各自生成的 anchor 参数代入原始 YOLOv4 中进行检测，两种算法的 mAP 值和 9 类物体各自 AP 值如图 8 所示。整体上，对于 K-means++算法得到的聚类中心作为

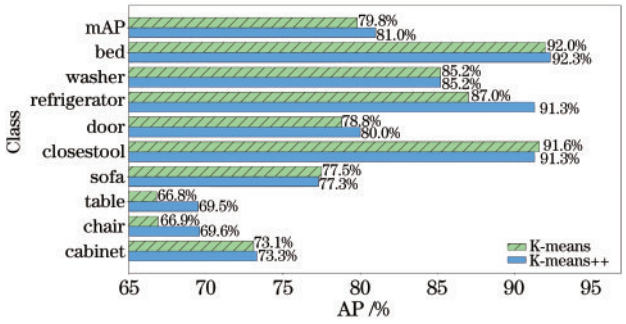

图 8　K-means算法和K-means＋＋算法在原始YOLOv4下的精度对比

Fig. 8　Accuracy comparison between K-means algorithm and K-means＋＋ algorithm under original YOLOv4

anchor参数，网络的检测效果更精确，相对于K-means算法，mAP值增加了1.2个百分点；K-means＋＋算法可以提升对大部分类别的平均预测精度，尤其是冰箱(refrigerator)类从87.0%提升到91.3%，只有小部分类别的AP值略微下降；对于桌(table)、椅(chair)这种类内较大差异的不规则物体，K-means算法效果较差，K-means＋＋算法的AP值均提高了1.7个百分点。综上所述，采用K-means＋＋算法后模型的检测效果更好。

4.3　网络结构的改进

4.3.1　颈部网络CSPNet结构化

改进原始YOLOv4网络的颈部，将CSPNet结构分别构造在SPP模块和PANet的连续卷积模块中，分别记作CSPSPP和CSPPAN，通过消融实验来比较两处改进对YOLOv4网络性能的影响。采用原始CSPDarknet53作为主干网络，利用控制变量法分别比较颈部模块改进前后对网络参数量(Parameters)、mAP和检测速度的影响，结果如表1所示。

表1　颈部网络融入CSPNet结构的性能结果

Table 1　Performance results of the neck network integrated into the CSPNet structure

Method	Neck module	Parameters /10^7	mAP /%	Speed /(frame·s^{-1})
1	SPP＋PAN	5.35	81.0	66.1
2	SPP＋CSPPAN	4.65	81.8	66.5
3	CSPSPP＋PAN	5.95	82.6	63.6
4	CSPSPP＋CSPPAN	5.25	83.6	62.3

表1中，对比方法1、2和3、4可知，CSPNet结构使参数量减少了11.7%～13.0%，这表明对PANet的CSPNet化可以减少模型参数量，并且mAP值也有小幅度提升；对比方法1、3和2、4可知，SPP模块的CSP网络化可以有效提升网络检测精度，分别提高1.6个百分点和1.8个百分点，相较于PANet的CSP化，CSPSPP模块的mAP提升更高，这也证实了在特征融合过程中，CSPNet结构能更好地融合梯度信息，提升检测精度；由于CSPSPP新增了卷积层，整个网络的参数量有所增加，检测速度略微降低。

4.3.2　深度可分离卷积

为了加快网络检测速度，构建易于部署的轻量化网络，满足实际应用中对实时性的要求，引入深度可分离卷积。在YOLOv4网络中，3×3卷积层主要存在于主干网络CSPDarknet53和路径聚合网络PANet中，本实验在这两处引入深度可分离卷积，分别记作DS-CSPDarknet53和DS-PAN。此外，由于CSPNet结构在颈部网络融合后取得了良好的效果，本实验保留CSPSPP模块，分析插入深度可分离卷积的CSPPAN模块(DS-CSPPAN)对网络结果的影响，实验结果如表2所示。

表2　深度可分离卷积对网络性能影响

Table 2　Impact of depthwise separable convolution on network performance

Method	Backbone	Neck	Parameters/10^7	mAP /%	Speed /(frame·s^{-1})
1	CSPDarknet53	PAN	5.95	82.6	63.5
2	CSPDarknet53	CSPPAN	5.25	83.6	62.3
3	DS-CSPDarknet53	PAN	4.41	81.9	67.0
4	DS-CSPDarknet53	CSPPAN	3.89	83.2	68.0
5	CSPDarknet53	DS-PAN	4.72	82.0	68.7
6	CSPDarknet53	DS-CSPPAN	4.60	82.9	65.8
7	DS-CSPDarknet53	DS-PAN	3.28	81.9	72.9
8	DS-CSPDarknet53	DS-CSPPAN	3.24	83.0	72.1

从表2可以看出，对于相同的主干网络，PANet融入CSPNet结构后，取得的mAP值总是提高1个百分点左右，这也进一步表明了CSPNet结构可以很好提升颈部网络的特征融合能力。方法3～8展示了在不同模块中插入深度可分离卷积后的实验结果，对比后可以发现，深度可分离卷积可以明显减少网络的参数

量，提高检测速度，但却减小了模型的容量，导致mAP值略微有所下降。对比方法2和8，在主干网络和CSPNet结构化的PANet中同时使用深度可分离卷积可使模型参数量降到最低，仅为原来的61.7%，这种轻量化的设计使检测速度提升了近16%，代价仅为mAP值降低了0.6个百分点。

4.4　网络性能比较

为体现K-means++算法、颈部网络CSPNet结构化(CSP-Neck)和深度可分离卷积的有效性，进行了消融实验，实验结果如表3所示。

表3中，对于先验框参数的生成，由K-means算法换成K-means++算法后，mAP有了一定的提升，由于网络结构参数未改变，因此检测速度不变。颈部网络CSPNet结构化前后，网络的权重改变不大，检测精度提升了1.4个百分比～2.6个百分比，但检测速度略有减小。深度可分离卷积弥补了颈部网络CSPNet结构化在速度上的损失，模型权重大小减少了近一半，速度有了较大的提升，同时mAP值损失很少，保证了检测的准确率。最终，同时使用K-means++聚类算法和改进的YOLOv4网络结构，在所构建的室内场景目标检测数据集上的mAP达83.0%，检测速度达72.1 frame/s，模型权重大小仅有65 MB，优于原始YOLOv4算法，有效地表明了改进的可行性。

表3　不同改进的性能比较

Table 3　Performance comparison of different improvements

Improvement strategy			Weight size /MB	mAP / %	Speed / (frame·s^{-1})
K-means++	CSP-Neck	DS			
			102	79.8	66.1
		√	52	79.3	78.6
	√		101	81.2	62.4
	√	√	65	80.6	71.9
√			102	81.0	66.1
√		√	52	80.8	78.1
√	√		101	83.6	62.3
√	√	√	65	83.0	72.1

图9、图10展示了原始YOLOv4算法与改进后的YOLOv4算法在室内场景中的检测结果。主观上，改进YOLOv4算法对目标物体的检测置信度更高，检测框的位置偏差更小，并且误检和漏检的情况更少。由图10可知，改进后算法在每种目标上的平均检测精度均高于原始算法，说明了先验框参数改进和网络结构改进对提高检测效果的有效性，对比图8可知，对于提升检测准确率，网络结构改进比先验框参数改进更有优势。

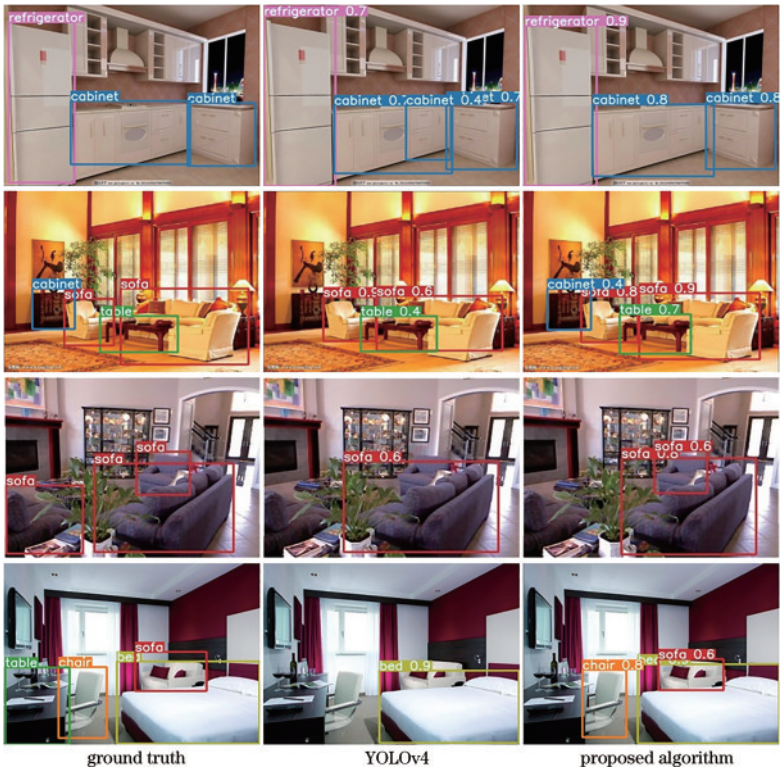

图9　算法改进前后检测效果对比

Fig. 9　Comparison of detection results before and after algorithm improvement

从检测误差的角度来看，改进后的YOLOv4算法缓解了漏检和误检的情况，但在桌(table)、椅(chair)这两类目标上的检测精度仍相对较差，产生这一现象的原因有很多。例如，受拍摄视角和类内差异的影响，目标物体的表现特征变化较大，一般网络难以准确检测，本文通过改进网络结构来提升网络的特征融合能力，提高了目标的整体识别效果，在桌、椅两类别上的AP值分别提升了3.2个百分点和4.9个百分点。此外，若训练集

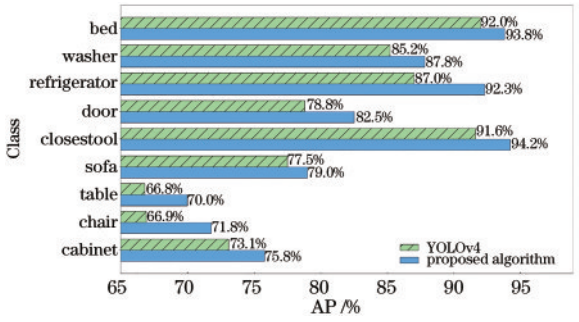

图 10　算法改进前后在各类别上检测结果对比

Fig. 10　Comparison of detection results on each class before and after algorithm improvement

中的样本分布不平衡，很容易导致模型对样本少的类别出现漏检现象，解决方法之一就是扩增弱势样本数量，平衡正负样本分布，这将是未来的研究方向之一。

表 4 记录改进 YOLOv4 算法与 Faster-RCNN、SSD、YOLOv3、YOLOv4、YOLOv5 算法的检测性能。结果表明，Faster-RCNN 作为 Two-stage 目标检测算法的代表，其 mAP 值大于 SSD 和 YOLOv3 这些 One-stage 算法，但是检测速度和精度并不平衡，其速度仅有 21.3 frame/s，远低于 One-stage 类算法，这也证实了 One-stage 和 Two-stage 目标检测算法各自的优势与缺陷。相比于其他 One-stage 算法，YOLOv4 已经取得较好的检测效果，检测精度和速度较为平衡。所提改进的 YOLOv4 算法的 mAP 值达 83.0%，速度达 72.1 frame/s，在原始 YOLOv4 的基础上分别提高了 3.2 个百分点和 6 frame/s。相较于目前 Github 上公认的 YOLOv5 算法，所提算法在检测速度相同的情况下具有更好的检测精度。在模型体积上，由于改进的 YOLOv4 算法在网络结构上进行了轻量化设计，模型大小仅有 65 MB，相比原始 YOLOv4 算法，降低了 36.3%。

表 4　不同检测算法的性能对比

Table 4　Performance comparison of different detection algorithms

Model	Weight size /MB	mAP /%	Speed /(frame·s^{-1})
Faster-RCNN-resnet50	315	79.0	21.3
SSD-resnet50	108	78.2	40.5
YOLOv3	118	75.2	60.8
YOLOv4	102	79.8	66.1
YOLOv5	94	81.5	72.5
Improved YOLOv4	65	83.0	72.1

5　结　　论

室内场景目标检测具有重要研究意义，针对如何优化模型的检测准确率与速度问题，提出一种改进的 YOLOv4 算法。首先，使用 K-means++算法代替传统 K-means 算法获取先验框参数；然后，使用 CSPNet 结构优化颈部的 SPP 模块和 PANet，提高模型的准确率；最后，采用深度可分离卷积代替原始网络中的部分 3×3 标准卷积，降低模型参数量，使模型更容易部署在小型移动设备上，同时提升检测速度。

实验结果表明：K-means++聚类算法可以稳定先验框参数的选取，消除模型出现局部最优的隐患，相较于原 YOLOv4 算法，mAP 值提高 1.2 个百分点；使用 CSPNet 结构优化网络颈部，消除梯度信息冗余，增强网络的学习能力，优化后模型的检测精度明显提升，mAP 值达 83.6%，但检测速度略有下降；使用深度可分离卷积轻量化网络模型后，相较于原始 YOLOv4 算法，权重减小了 36.3%，速度提高了近 16 frame/s；改进后，算法的准确率和速度较为平衡，在室内场景数据集上的平均精度达 83.0%，同时检测速度为 72.1 frame/s，均高于现阶段其他目标检测算法，由于模型权重更小，更适合配置于移动设备中。

本文研究内容仍然存在进一步优化的空间，例如所指出的算法都运用 anchor-based 思想。目前已存在基于 anchor-free 的目标检测算法，这类算法除去了 NMS 的过程，缩短了检测时间，因此在后续工作中，可以借鉴该思想来优化算法，摆脱对先验框的依赖，进一步提升检测速度。

参　考　文　献

[1] Zhang H J, Zhang C N, Yang W, et al. Localization and navigation using QR code for mobile robot in indoor environment[C]//2015 IEEE International Conference on Robotics and Biomimetics, December 6-9, 2015, Zhuhai, China. New York: IEEE Press, 2015: 2501-2506.

[2] Rafique A A, Jalal A, Kim K. Statistical multi-objects segmentation for indoor/outdoor scene detection and classification via depth images[C]//2020 17th International Bhurban Conference on Applied Sciences and Technology (IBCAST), January 14-18, 2020, Islamabad, Pakistan. New York: IEEE Press, 2020: 271-276.

[3] Espinace P, Kollar T, Soto A, et al. Indoor scene recognition through object detection[C]//2010 IEEE International Conference on Robotics and Automation, May 3-7, 2010, Anchorage, AK, USA. New York: IEEE Press, 2010: 1406-1413.

[4] Kim J, Lee C H, Young-Chul, et al. Optical sensor-

based object detection for autonomous robots[C]//2011 8th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), November 23-26, 2011, Incheon, Korea (South). New York: IEEE Press, 2011: 746-752.

[5] Zhu S G, Du J P, Ren N. A novel simple visual tracking algorithm based on hashing and deep learning[J]. Chinese Journal of Electronics, 2017, 26(5): 1073-1078.

[6] 罗会兰, 陈鸿坤. 基于深度学习的目标检测研究综述[J]. 电子学报, 2020, 48(6): 1230-1239.
Luo H L, Chen H K. Survey of object detection based on deep learning[J]. Acta Electronica Sinica, 2020, 48(6): 1230-1239.

[7] 邹斌, 林思阳, 尹智帅. 基于YOLOv3和视觉SLAM的语义地图构建[J]. 激光与光电子学进展, 2020, 57(20): 201012.
Zou B, Lin S Y, Yin Z S. Semantic mapping based on YOLOv3 and visual SLAM[J]. Laser & Optoelectronics Progress, 2020, 57(20): 201012.

[8] Afif M, Ayachi R, Said Y, et al. An evaluation of RetinaNet on indoor object detection for blind and visually impaired persons assistance navigation[J]. Neural Processing Letters, 2020, 51(3): 2265-2279.

[9] 姚晓宇. 基于深度学习的室内目标检测的方法研究[D]. 成都: 电子科技大学, 2018.
Yao X Y. Research on indoor object detection based on deep learning[D]. Chengdu: University of Electronic Science and Technology of China, 2018.

[10] 李维刚, 叶欣, 赵云涛, 等. 基于改进YOLOv3算法的带钢表面缺陷检测[J]. 电子学报, 2020, 48(7): 1284-1292.
Li W G, Ye X, Zhao Y T, et al. Strip steel surface defect detection based on improved YOLOv3 algorithm [J]. Acta Electronica Sinica, 2020, 48(7): 1284-1292.

[11] 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.

[12] He K M, Gkioxari G, Dollár P, et al. Mask R-CNN[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(2): 386-397.

[13] 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, 2015, 9905: 21-37.

[14] 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, June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 779-788.

[15] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition, July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 6517-6525.

[16] Redmon J, Farhadi A. YOLOv3: an incremental improvement[EB/OL]. (2018-04-08)[2021-02-03]. https://arxiv.org/abs/1804.02767.

[17] Bochkovskiy A, Wang C Y, Liao H Y M. YOLOv4: optimal speed and accuracy of object detection[EB/OL]. (2020-04-23)[2021-05-09]. https://arxiv.org/abs/2004.10934.

[18] Wang C Y, Liao H Y M, Wu Y H, et al. CSPNet: a new backbone that can enhance learning capability of CNN[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), June 14-19, 2020, Seattle, WA, USA. New York: IEEE Press, 2020: 1571-1580.

[19] He K M, Zhang X Y, Ren S Q, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1904-1916.

[20] Liu S, Qi L, Qin H F, et al. Path aggregation network for instance segmentation[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE Press, 2018: 8759-8768.

[21] 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, July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 936-944.

[22] Wang C Y, Bochkovskiy A, Liao H Y M. Scaled-YOLOv4: scaling cross stage partial network[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 20-25, 2021, Nashville, TN, USA. New York: IEEE Press, 2021: 13024-13033.

[23] 张有波, 郭威, 周悦, 等. 基于多粒度剪枝的水下遗迹实时目标检测[J]. 激光与光电子学进展, 2021, 58(14): 1410019.
Zhang Y B, Guo W, Zhou Y, et al. Real-time target detection of underwater relics based on multigranularity pruning[J]. Laser & Optoelectronics Progress, 2021, 58 (14): 1410019.

[24] Jacob B, Kligys S, Chen B, et al. Quantization and training of neural networks for efficient integer-arithmetic-only inference[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE Press, 2018: 2704-2713.

[25] Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network[EB/OL]. (2015-03-09)[2021-05-07]. https://arxiv.org/abs/1503.02531.

[26] 于长东, 毕晓君, 韩阳, 等. 基于轻量化深度学习模型的粒子图像测速研究[J]. 光学学报, 2020, 40(7): 0720001.
Yu C D, Bi X J, Han Y, et al. Particle image velocimetry based on a lightweight deep learning model [J]. Acta Optica Sinica, 2020, 40(7): 0720001.

[27] Howard A G, Zhu M L, Chen B, et al. MobileNets: efficient convolutional neural networks for mobile vision applications[EB/OL]. (2017-04-17)[2021-06-05]. https://arxiv.org/abs/1704.04861.