DOI:10.11918/202206056

MPE-YOLOv5:面向边缘计算的轻量化YOLOv5手势识别算法

黄凯雯[1,2],房宵杰[1,2],梅 林[1],田涛涛[1],杜兆鹏[1]

(1.哈尔滨工业大学 电子与信息工程学院,哈尔滨 150090;2.通信网信息传输与分发技术重点实验室,石家庄 050081)

摘 要:针对边缘设备计算和存储能力差的问题,本文对传统YOLOv5模型中用于特征提取的主干网络CSPDarkNet53进行轻量化处理,提出了一种轻量化MPE-YOLOv5手势识别算法,以实现模型在低功耗边缘设备上的部署;针对轻量化模型提取特征较少而导致的难以识别大尺度变换目标和微小目标问题,对M-YOLOv5网络设计添加有效通道注意力机制(efficient channel attention,ECA),以缓解因特征通道减少而导致的高层特征信息丢失的问题;同时增加针对微小目标的检测层,提高对微小目标手势的敏感度;并选用EIoU作为预测锚框的损失函数,以提高模型的定位精度。本文在自制数据集和NUS-Ⅱ公共数据集上验证了MPE-YOLOv5算法有效性,并将MPE-YOLOv5算法与轻量化后的M-YOLOv5算法和原始的YOLOv5算法在自制数据集上进行了对比实验。实验结果表明,改进算法的模型参数量、模型大小和计算复杂度分别是原算法的21.16%、25.33%和27.33%,平均精度可达97.2%;与轻量化模型M-YOLOv5相比,MPE-YOLOv5能够在保持原来效率的同时,使平均精度提升8.72%。因此,所提MPE-YOLOv5算法能够较好地平衡模型的检测精度和实时推理速度,可实现在硬件受限的边缘终端上进行部署。

关键词:手势识别;YOLOv5;轻量化模型;通道注意力机制;损失函数

中图分类号:TP391.4 **文献标志码**:A **文章编号**:0367-6234(2023)05-0001-13

MPE-YOLOv5: lightweight YOLOv5 gesture recognition algorithm for edge computing

HUANG Kaiwen[1,2], FANG Xiaojie[1,2], MEI Lin[1], TIAN Taotao[1], DU Zhaopeng[1]

(1. School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150090, China;
2. Science and Technology on Communication Networks Laboratory, Shijiazhuang 050081, China)

Abstract: In view of the weaknesses of poor computing and storage capabilities of edge devices, lightweight processing was carried out on the backbone network CSPDarkNet53 for feature extraction in the traditional YOLOv5 model, and a lightweight gesture recognition algorithm MPE-YOLOv5 was proposed to realize the deployment of the model in low-power edge devices. Considering the problem that it is difficult to identify large-scale transformation targets and tiny targets due to less feature extraction in lightweight model, efficient channel attention (ECA) mechanism was added to alleviate the loss of information after high-level feature mapping due to the reduction of feature channel. A detection layer for tiny targets was added to improve the sensitivity to tiny target gestures. EIoU was selected as the loss function of the detection frame to improve the positioning accuracy. The effectiveness of the MPE-YOLOv5 algorithm was verified on the self-made dataset and NUS-Ⅱ public dataset, and the MPE-YOLOv5 algorithm was compared with lightweight M-YOLOv5 algorithm and original YOLOv5 algorithm on the self-made dataset. Experimental results show that the model parameters, model size, and computational complexity of the improved algorithm were 21.16%, 25.33%, and 27.33% of the original algorithm, and the average accuracy was 97.2%. Compared with the lightweight model M-YOLOv5, MPE-YOLOv5 improved the average accuracy by 8.72% while maintaining the original efficiency. The proposed MPE-YOLOv5 algorithm can better balance between the detection accuracy and real-time reasoning speed of the model, and can be deployed on edge terminals with limited hardware.

Keywords: gesture recognition; YOLOv5; lightweight model; channel attention mechanism; loss function

收稿日期:2022-06-14;**录用日期**:2022-08-02;**网络首发日期**:2022-08-29
网络首发地址:https://kns.cnki.net/kcms/detail/23.1235.T.20220826.1621.016.html
基金项目:黑龙江省优秀青年基金(YQ2021F003);通信网信息传输与分发技术重点实验室基金(FFX22641X005)
作者简介:黄凯雯(1999—),女,硕士研究生;房宵杰(1989—),男,副教授,博士生导师
通信作者:房宵杰,fangxiaojie@hit.edu.cn

为了克服基于传感器等外部设备的手势识别技术[1-4]成本较高的局限性,基于计算机视觉的手势识别技术成为学者们的重要研究方向。该技术先利用分割算法将手势从复杂的背景中分离,再通过图像描述算法获取手势的统计值来提取特征,如 HOG 算法[5]、SIFT 算法[6]、SURF 算法[7]等,之后结合机器学习的分类器,如 SVM[8]、Adaboost[9]等进行分类。但该技术在复杂场景下鲁棒性差、识别精度低,并且算法设计过程复杂,人工和时间成本较高。近年来,由于卷积神经网络(convolutional neural networks,CNNs)强大的提取特征能力以及 CNNs 在目标检测领域的快速发展,基于深度学习的手势识别逐渐成为研究趋势。该技术与传统技术相比准确率较高,在复杂环境下具有良好的鲁棒性和灵活性。为了实现快速识别,单阶段目标检测算法被广泛应用于手势识别领域。文献[10]融合 YOLO 中的定位方法与 CNNs 的分类方法,提出一种 HGDR-Net 算法,在 NUS-II 和 Marcel 这两个数据集上,平均识别精度分别达到 98.65% 和 99.59%,但识别过程复杂,推理速度较慢;文献[11]利用改进的 YOLOv3-tiny-T 算法进行手势识别,推理速度虽快,但是识别精度相对较低,平均识别精度为 92.24%;文献[12]基于 YOLOv3,提出了改进的 YOLOv3-tiny 手势识别方法,平均识别精度可达 97.3%,但该模型复杂度较高,网络模型大小为 27.6 MB。虽然现有的改进模型平均识别精度高,但由于高精度模型是以庞大的参数量和复杂的结构为代价,对于算力和存储能力受限的边缘设备而言,无法直接进行部署,并且识别精度和推理速度是一对相互矛盾的参数,高精度模型一般伴随着识别速度慢的问题。

设计高效且轻量的网络体系结构是一种有效的模型压缩和加速方法,并且能够缓解高精度模型无法部署在边缘设备的现象。近年来,出现了大量的轻量化网络模型,并且逐渐地实现了在边缘设备上的部署。例如,ShuffleNetv2[13]提出了通道分割,并将输入的特征图分成两部分,在加速网络的同时进行特征复用,达到了较好的识别效果;MobileNetv3[14]使用深度可分离卷积和倒残差结构来构建轻量化网络,但在该模型中过度使用逐点卷积使得识别精度下降;GhostNet[15]用更少的参数量来提取更多的特征,并使用输出少的非卷积层输出特征,再对输出使用简单的线性操作来生成更多的特征。但轻量化网络由于本身的参数量少,因此相较于高精度模型,提取的特征信息较少,在实际的识别场景中,会存在识别错误或漏检等情况。

为解决上述问题,本文提出了一种 MPE-YOLOv5 的手势快速识别算法,结合单阶段目标检测模型 YOLOv5[16]和轻量化模型 MobileNetv3[14],并在此基础上进行改进。经验证,该算法能够在平衡网络模型的检测精度和推理速度的同时,解决高精度模型推理框架复杂和轻量化模型因参数量少而带来的模型精度下降、无法有效识别小目标和大尺度变换目标的问题。

1 传统 YOLOv5 算法

YOLOv5[16]是由 UitralyticsLLC 公司在 2020 年提出的基于单阶段的目标检测算法,其基本框架由输入、主干网络、瓶颈网络以及预测网络 4 部分构成。YOLOv5 网络模型的输入通过 Mosaic 方法来丰富数据集。主干网络通过 CSPDarknet53 结构进行特征提取。瓶颈网络采用“FPN + PANet”双塔结构聚合当前的图像特征:FPN 结构自顶向下传递高层的语义特征,对金字塔结构进行语义信息增强;PANet 结构是在 FPN 结构后添加一个自底向上的金字塔结构,对定位信息进行补充,将底层的强定位特征向上传递。最后预测网络将目标进行定位,并利用非极大值抑制 NMS 对目标分类预测并将结果输出。

传统 YOLOv5 中包含 4 个模型:YOLOv5s、YOLOv5m、YOLOv5l 和 YOLOv5x,4 个模型的具体结构一致,不同之处在于模型的深度和宽度。其中,YOLOv5s 是 YOLOv5 系列中最小的模型。在复杂场景下的手势识别任务中,YOLOv5s 的模型参数量为 7.09×10^6,浮点运算次数 F_{FLOPs} 为 1.60×10^{10},所占的内存大小为 13.70 MB。对于计算和存储能力受限的边缘设备来说,其模型大小和计算复杂度仍然较大。此外,内存的读写是十分消耗能量的,加载大框架模型会产生较大的额外开销,影响设备的检测效率。因此,上述的算法适用于在计算能力较强的服务器端上进行检测,对于算力受限的边缘移动设备,本文在原有模型的基础上,提出了一种更加轻量化的快速检测算法,并且以较少的精度损失为代价,较好地平衡了模型的效率和有效性。

2 改进的 YOLOv5 手势识别算法

2.1 主干网络的轻量化处理

YOLOv5 模型的主干网络采用 CSPDarknet53 结构,该结构参数量较多,约占总参数量的 60%,因此为了实现 YOLOv5 的轻量化设计,对主干网络进行裁剪或替换是网络轻量化首选方案。本文在 MobileNet 系列模型[14,17-18]的基础上,构造了轻量化的 MBConv 模块,并将该模块堆叠,作为 YOLOv5 的特征提取网络,从而构建一个结构简单、参数量少、对硬件设备算力要求低的轻量化新架构——M-YOLOv5模型。

图 1 为 MBConv 模块的结构图。该结构先由 1×1卷积将原来的特征图升维，来获取跨通道信息，之后通过 3×3 逐层卷积提取每一层的特征，进而获取通道内的空间信息，再经过 1×1 卷积进行跨通道的信息融合，形成新的特征图。获取新的特征图后，利用 SE 注意力机制[19]获取空间权重来优化模型性能，并且当输入与输出通道数相同时，执行逆残差连接，缓解由于网络深度增加而造成的梯度弥散现象。

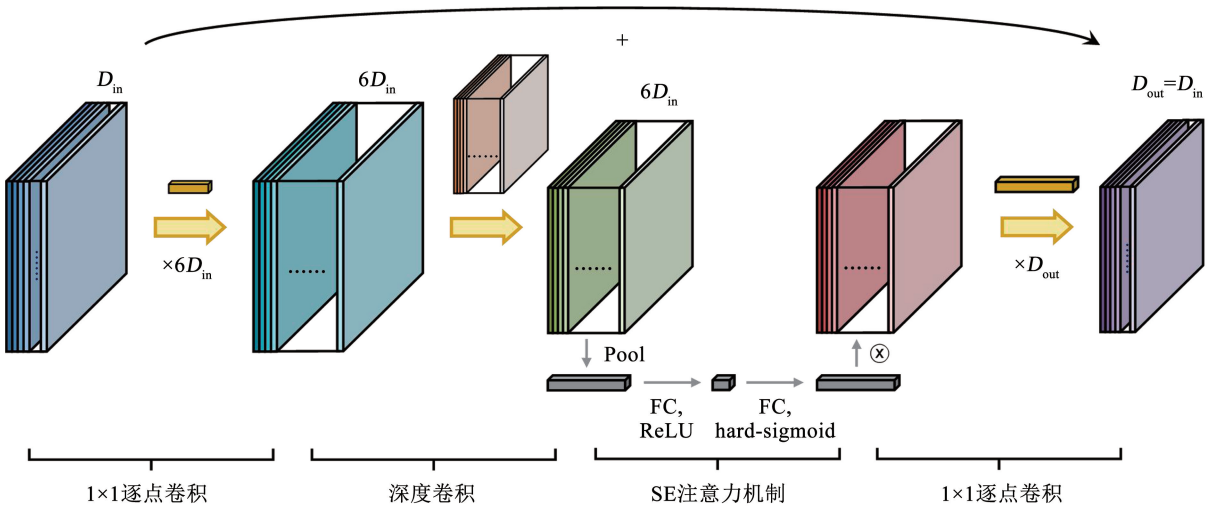


图 1　MBConv 模块结构

Fig. 1　Structure of MBConv module

为了减少深层网络的参数量，本文将第一卷积层 CBS 中的 SiLU 非线性激活函数替换为 h-swish 函数，构造 CBH 模块，对输入的图像进行特征预提取；在 CBH 模块之后，将 11 个 MBConv 模块堆叠，其中，将卷积步长为 1 的 MBConv 模块记为 MBConv_1，卷积步长为 2 的 MBConv 模块记为 MBConv_2。当对特征图进行下采样时，设置 MBConv 模块的步长为 2。为了增加模型对不同尺度目标的灵敏度，在主干网络进行第 2 次、第 3 次和第 4 次下采样时，即在主干网络和瓶颈网络的连接处增加一个 MBConv 模块，并设置 MBConv 模块的步长为 1，使主干网络对不同尺度目标所提取的特征更多而不影响输出特征图的尺寸大小，减少由于主干网络轻量化对模型带来的负面影响；保留原有模型中的 SPPF 模块。把上述结构作为 YOLOv5 主干网络，并保留 YOLOv5 模型中的其他结构，得到轻量化的 M-YOLOv5模型。所构造 M-YOLOv5 模型的主干网络结构如表 1 所示。

表 1　M-YOLOv5 模型的主干网络结构

Tab. 1　Backbone network structure of M-YOLOv5 model

模块	数量	参数量	输出通道数	卷积核大小	步长	输入长宽大小/像素	输出长宽大小/像素
CBH	1	232	8	3×3	2	640×640	320×320
MBConv_2	1	628	8	3×3	2	320×320	160×160
MBConv_2	1	2 696	16	3×3	2	160×160	80×80
MBConv_1	1	3 992	16	3×3	1	80×80	80×80
MBConv_2	1	11 400	24	3×3	2	80×80	40×40
MBConv_1	4	133 578	24	3×3	1	40×40	40×40
MBConv_2	1	71 016	48	3×3	2	40×40	20×20
MBConv_1	2	477 408	48	3×3	1	20×20	20×20
SPPF	1	8 456	48	—	—	20×20	20×20

2.2　小目标检测与多尺度目标检测的改进

YOLO 系列算法[16,20-23]在基于单阶段的目标检测任务中发挥着重要作用，然而将轻量化后的 M-YOLOv5模型直接应用于复杂场景下的手势识别主要存在着几个问题。

首先，由于在 M-YOLOv5 模型中，对主干网络进行了轻量化处理，与原有模型相比，模型的参数量较少，因此其特征网络提取的特征较少，这将会导致小目标的检测精度下降和错误分类，甚至会出现识别不到目标的情况。其次，当输入为小目标图像时，手势只占特征图的较小部分，其余均为背景信息，经过层次较深的网络多次卷积后，背景信息迭代会产生大量的冗余，淹没有效信息，从而导致小目标检测精度下降。因此，本文通过一种有效通道注意力机制（efficient channel attention，ECA）[24]获取跨通道信息，使模型能够更加精确地定位感兴趣的区域并对小目标进行识别，其结构如图 2 所示。

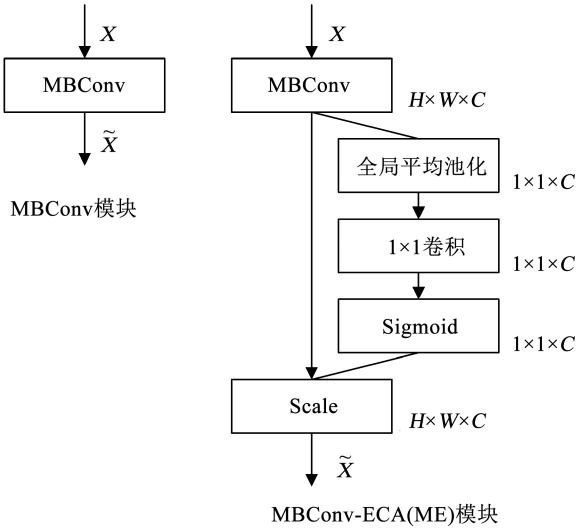


图 2　ECA 结构

Fig. 2　Structure of ECA

如图 2 所示,ECA 主要分为两个步骤。第一步是通道信息的获取。对输入的特征图进行全局平均池化,池化后的特征图维度从 $W \times H \times C$ 被压缩为 $1 \times 1 \times C$,那么每个通道都有对应的权值。第二步是通道信息特征图的生成。将维度为 $1 \times 1 \times C$ 的特征向量进行 1×1 卷积,并经过 Sigmoid 非线性激活函数,得到每个通道的权值。最后将权值作用于特征图上,使用残差连接将其与输入的特征图相乘,使得大权值所对应的特征图输出值更大,这样模型便能够更精确地定位感兴趣的区域。

本文为了精确定位小目标,提升小目标的检测精度,在输入与输出通道数不同的 MBConv 模块之前集成 ECA 注意力机制模块,并与瓶颈网络相连接,不仅减少了由于背景信息迭代而引起的有效信息丢失,也缓解了由于特征通道数增加而导致的高层信息丢失的现象。

此外,在实际手势识别与检测场景下,当手势与用于识别的摄像头距离变化较大时,将使得目标比例变化较为剧烈,即产生大尺度方差。感受野是 CNNs 每一层输出特征图上的像素点在输入特征图上映射的区域大小,当特征图尺度越大时,其感受野越小,即代表在特征图中的映射区域越小,在目标检测任务中可以检测到原图像较小的目标。因此针对大尺度方差问题,在原有的预测网络基础上,增加一个针对微小目标的检测层,使得当输入特征图的尺寸为 640 × 640 时,输出为 4 个不同尺度的特征图,这样能够有效地利用浅层的信息,对小目标的检测更加敏感。

由于增加了检测层,需要对检测的锚框进行重新配置,锚框的配置如表 2 所示。

整合以上改进方案,整体的网络结构如图 3 所示。

表 2　锚框配置表

Tab. 2　Detection frame allocation table

特征图尺度/像素	感受野	小锚框长宽/像素	中锚框长宽/像素	大锚框长宽/像素
20 × 20	大	(116,90)	(156,198)	(373,326)
40 × 40	中	(30,61)	(62,45)	(59,119)
80 × 80	小	(10,13)	(16,30)	(33,23)
160 × 160	微小	(5,6)	(8,14)	(15,11)

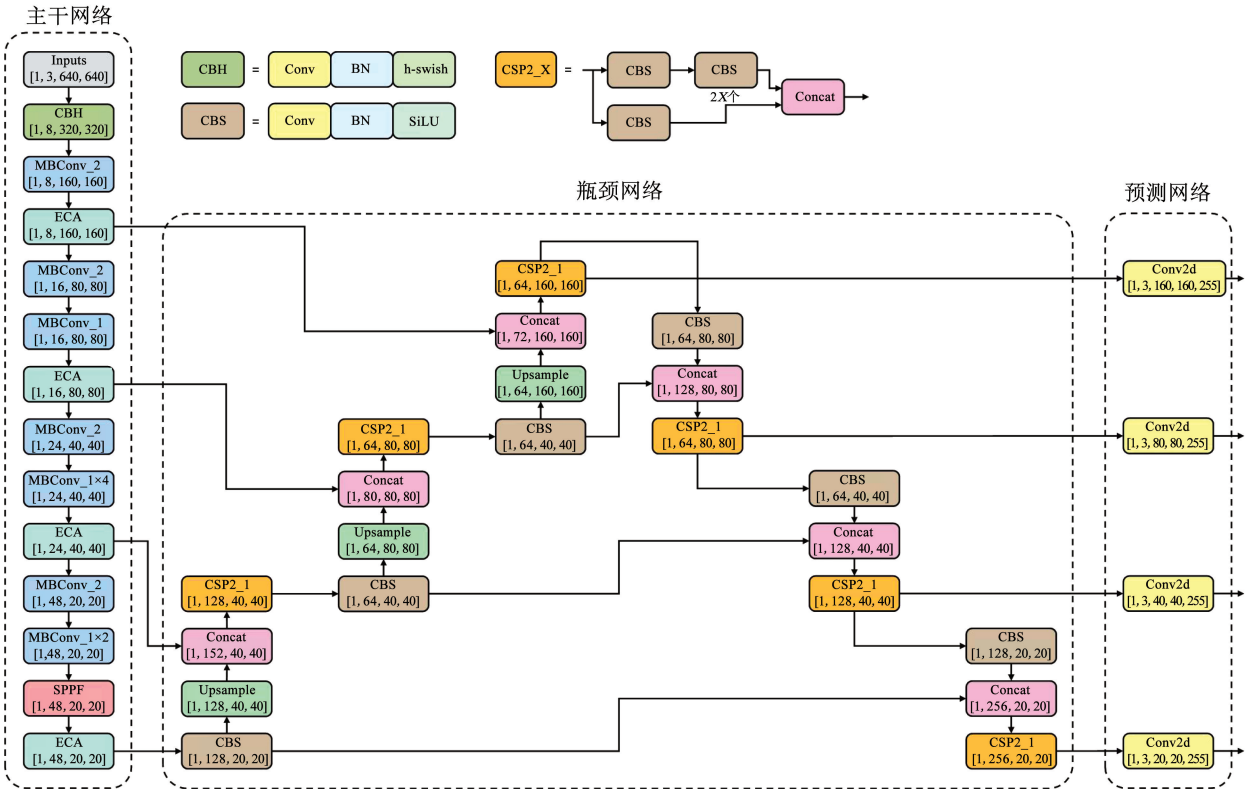


图 3　MPE-YOLOv5 整体网络结构图

Fig. 3　Overall network structure of MPE-YOLOv5

2.3　边界框损失函数的改进

在对 CNNs 进行训练时，当模型对目标预测得越准确，损失函数值越小，模型的鲁棒性也越好。YOLO 系列模型的损失函数包含置信度损失函数 L_{object}、边界框损失函数 L_{box} 和分类损失函数 L_{class} 三类损失，因此 YOLO 系列模型的损失函数为 3 个损失加权求和所得，如式(1)所示：

$$L = a \cdot L_{\text{object}} + b \cdot L_{\text{box}} + c \cdot L_{\text{class}} \tag{1}$$

式中，YOLOv5 的默认值为 $a = 0.7$、$b = 0.05$ 和 $c = 0.3$，边界框损失函数为 CIoU[25]，即 $L_{\text{box}} = L_{\text{CIoU}}$，当边界框预测越准确时，$L_{\text{box}} = L_{\text{CIoU}}$ 将越小，模型的性能也越好。

CIoU 损失函数在 DIoU 损失函数[25]的基础上，增加了影响因子 αv，将预测锚框与真实锚框的交并比损失 $L_{\text{IoU}} = 1 - I_{\text{U}}$、预测锚框与真实锚框的中心点距离损失 $L_{\text{dis}} = \rho^2(b, b_{\text{gt}})/c^2$ 和横纵比损失 $L_{\text{asp}} = \alpha v$ 考虑在内，CIoU 函数的数学表达式如式(2)[25]所示：

$$L_{\text{CIoU}} = L_{\text{IoU}} + L_{\text{dis}} + L_{\text{asp}} = 1 - I_{\text{U}} + \frac{\rho^2(b, b_{\text{gt}})}{c^2} + \alpha v \tag{2}$$

其中，式(2)的各变量如式(3)～(5)所示：

$$I_{\text{U}} = \left| \frac{B \cap B_{\text{gt}}}{B \cup B_{\text{gt}}} \right| \tag{3}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w_{\text{gt}}}{h_{\text{gt}}} - \arctan\frac{w}{h}\right)^2 \tag{4}$$

$$\alpha = \frac{v}{(1 - I_{\text{U}}) + v} \tag{5}$$

式中：B 和 B_{gt} 分别为预测锚框和真实锚框，c 为外界矩形框的对角线长度，ρ 为预测锚框和真实锚框的中心点距离，b 和 b_{gt} 分别为预测锚框和真实锚框的中心点，h 和 w 分别为预测锚框的高度和宽度，h_{gt} 和 w_{gt} 分别为真实锚框的高度和宽度。

式(3)中，I_{U} 为预测锚框和真实锚框的交并比，即两锚框的最大重叠区域，因此 I_{U} 值越大，说明 2 个锚框的重叠度越高，即 $I_{\text{U}} \to 1$，则 $L_{\text{IoU}} = 1 - I_{\text{U}} \to 0$；式(4)中，$v$ 为预测锚框和真实锚框的宽高比相似度，其用类似均方差的形式来表示，v 越小，2 个锚框的宽高比的相似度越高；式(5)中，α 为 v 的权重函数，其根据 I_{U} 和 v 的值来调节，若预测锚框 B 与真实锚框 B_{gt} 重叠区域小，则 v 的影响较小，重点调整 B 和 B_{gt} 之间距离，反之，若 B 和 B_{gt} 的重叠区域大，则 v 的影响大，那么重点调整 B 和 B_{gt} 的纵横比。

但 v 映射的是宽高比相似度的差异，是一个相对值，而不是宽高比与其置信度的差异。针对这一问题，文献[26]提出了 EIoU 边界框损失函数，该函数在 CIoU 函数的基础上，将影响因子 αv 拆分为宽度和高度的损失值，分别计算真实锚框与预测锚框的长宽，如式(6)[26]所示：

$$L_{\text{asp}} = \frac{\rho^2(w, w_{\text{gt}})}{c_w^2} + \frac{\rho^2(h, h_{\text{gt}})}{c_h^2} \tag{6}$$

式中：c_w 为外界矩形框的宽度，c_h 为外界矩形框的高度。

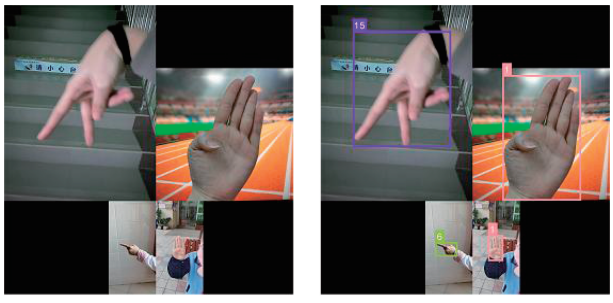
将 YOLOv5 中使用的 CIoU 边界框损失函数替换为 EIoU 损失函数，即使 $L_{\text{box}} = L_{\text{EIoU}}$，延续 CIoU 函数中的损失约束，并且将影响因子 αv 拆分为宽度和高度的损失值，直接使得预测锚框和真实锚框的宽高差值最小，模型在训练时损失函数的收敛速度更快，更有助于模型在学习的过程中降低损失值，提高定位精度，从而提高模型的识别精度。

3　实验结果和分析

3.1　实验环境与数据集

本文所有实验均在操作系统 Windows11 下执行，处理器为 Intel® Core™ i7-11800HQ CPU@2.30 GHz，32 GB运行内存，显卡为 NVIDA GeForce RTX 3070。本文基于深度学习框架 PyTorch1.10，搭建的软件环境为 Anaconda3.0、PyCharm2021.3、Python3.8，GPU 加速软件为 CUDA11.4 和 CUDNN8.2。

由于现有的关于小目标手势数据集有限，本文提出了由个人采集并制作的 26 类复杂背景小目标手势数据集[27]，包括如下步骤：1）采集手势图像，通过移动设备拍摄手势图像并进行整理分类，对其进行标签标注，所述手势类别包括 A～Z 的 26 种手势；2）拼接图像构建大尺度方差的小目标数据集。所述小目标手势样本标注示例如图 4 所示。



(a) 小目标手势姿势样本　　(b) 小目标手势姿势样本标注

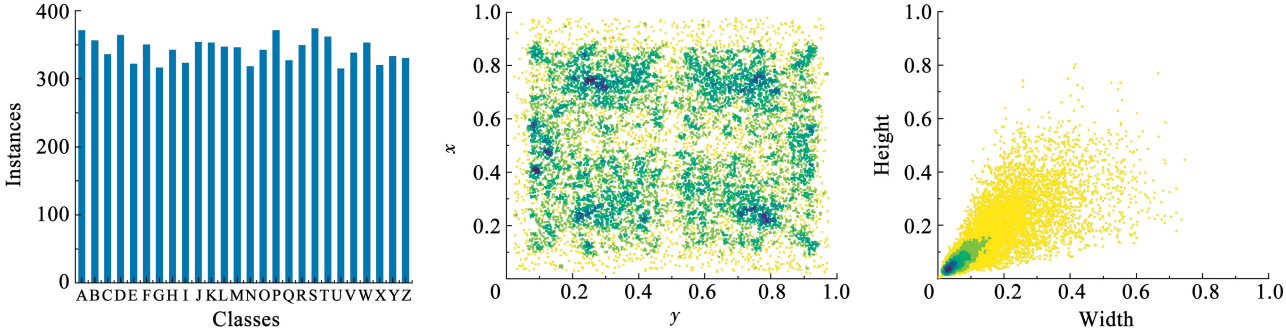
图 4　小目标手势样本标注示例

Fig. 4　Annotation examples of small target gesture samples

如图 4 所示，构建大尺度方差的小目标数据集的具体步骤为：1）随机选定 4 张样本进行拼接，拼接方式选用对角拼接；2）提取出 4 张手势样本高度和宽度的像素值，根据提取出的相对像素进行缩放，使得高像素样本所占图像比例较大，低像素样本所占图像比例较小，并确定出合并后大图的像素；

3)确定拼接中心点,4 张样本各选一个顶点拼接,即左上样本的右下角、右上样本的左下角、左下样本的右上角和右下样本的左上角;4)新建一个空的大图,将各个小样本进行填充,构建大尺度方差的小目标样本。

自制大尺度方差的小目标数据集较为合理地模拟了变化剧烈的目标比例,拼接后一共有 11 140 个小目标手势样本,包含 26 种手势,由 54 个人员组成。为了验证该数据集的均衡性和有效性,本文对该自制数据集进行可视化,结果见图 5。



(a)手势类别与其对应的样本数量　(b)各样本中心点坐标归一化分布　(c)各样本长宽归一化分布

图 5　大尺度方差的小目标手势数据集可视化结果

Fig. 5　Visualization results of small target gesture dataset with large-scale variance

由图 5(a)可知,自制小目标手势数据集基本符合均衡性要求,更有助于提高目标检测算法的效率。由图 5(b)、(c)可知,各个手势样本的中心点分布较为均匀,并且以小目标手势样本占大多数。验证了该数据集的有效性。

在本文的实验中,手势数据集按照 6∶2∶2 比例划分为训练集、验证集和测试集。输入模型的手势图像尺寸为 640×640,模型训练迭代次数为 200,批处理大小为 16,初始和最终学习率分别为 0.01 和 0.1,权重衰减率为 0.000 5,动量因子为 0.937,优化器为 SGD。本实验将对网络模型训练结果、可视化结果、算法对比结果和消融实验结果进行分析,以验证所提出 MPE-YOLOv5 算法的有效性和可行性。

3.2　评价指标

本文的目的是在保证模型检测精度和速度的同时,减少模型的训练时间和模型大小。因此,通过以下的评价指标验证模型的性能,其中包括效率(即计算复杂度 F_{FLOPs}、模型参数量 P_{param}、模型大小 W_{size} 和每秒检测帧数 F_{frame})和有效性(即精确度 P、回召率 R、平均精度 P_{MA} 和精度密度 D_{mAP}),以下具体说明各个评价指标的定义,具体检测指标定义如表 3 所示。

表 3　检测指标定义

Tab. 3　Definition of detection indexes

预测	正样本	负样本
预测为正	True Positive (T_{P})	False Positive (F_{P})
预测为负	False Negative (F_{N})	True Negative (T_{N})

准确率 P 为预测为正的正样本与所有预测为正样本的比值,如式(7)所示:

$$P=\frac{T_{\mathrm{P}}}{T_{\mathrm{P}}+F_{\mathrm{P}}} \tag{7}$$

召回率 R 为预测为正的正样本与所有正样本的比值,如式(8)所示:

$$R=\frac{T_{\mathrm{P}}}{T_{\mathrm{P}}+F_{\mathrm{N}}} \tag{8}$$

P_{A} 为 $P-R$ 曲线与坐标轴包围的面积,模型性能的好坏影响着面积的大小,当 $P-R$ 曲线所包围的面积越大时,即当曲线越靠近(1,1)坐标时,模型的性能越好。P_{MA} 为所有分类 P_{A} 的平均值,当 P_{MA} 越大,则算法的检测效果越好,P_{A} 和 P_{MA} 分别如式(9)、式(10)所示:

$$P_{\mathrm{A}}=\int_0^1 P(R)\,\mathrm{d}R \tag{9}$$

$$P_{\mathrm{MA}}=\frac{\sum_{q=1}^{Q}P_{\mathrm{A}}(q)}{Q} \tag{10}$$

式中 Q 为目标类别数量。

精度密度 D_{mAP}[28] 为平均精度与模型参数的比值,该值越高,模型的检测效率越高,如式(11)所示:

$$D_{\mathrm{mAP}}=\frac{P_{\mathrm{MA}}}{P_{\mathrm{param}}} \tag{11}$$

3.3　实验结果与分析

3.3.1　训练结果与分析

MPE-YOLOv5 算法在测试集中各类别的有效性检测结果如表 4 所示。由表 4 可知,MPE-YOLOv5 算法的精准率 P、召回率 R 和平均精度 P_{MA} 分别达到 94.5%、91.1%、97.2%,能够有效地实现在复杂场景下的手势识别和分类。

表 4　MPE-YOLOv5 算法各类别的准确率、回召率和平均精度检测结果

Tab. 4　Detection results of P, R, and P_{MA} of MPE-YOLOv5 algorithm for each category

类别	P	R	P_{MA}	类别	P	R	P_{MA}	类别	P	R	P_{MA}
A	0.914	0.941	0.955	J	1.000	0.917	0.958	S	0.947	0.898	0.949
B	0.980	1.000	0.995	K	0.995	0.933	0.988	T	0.895	0.810	0.966
C	0.934	0.955	0.988	L	0.937	1.000	0.995	U	0.950	0.902	0.978
D	0.971	1.000	0.995	M	0.897	0.846	0.937	V	0.870	0.916	0.978
E	1.000	0.992	0.995	N	0.935	0.653	0.965	W	0.980	1.000	0.995
F	0.944	0.957	0.985	O	0.966	0.857	0.967	X	0.931	0.722	0.892
G	0.914	1.000	0.984	P	0.977	1.000	0.995	Y	0.914	1.000	0.995
H	0.928	0.941	0.990	Q	0.966	0.936	0.993	Z	0.953	0.893	0.963
I	0.979	0.929	0.972	R	0.879	0.700	0.889	All	0.945	0.911	0.972

如图 6 所示为训练 MPE-YOLOv5 模型后在测试集上所得到 $P-R$ 曲线。从图 6 可得，该网络模型的性能良好。

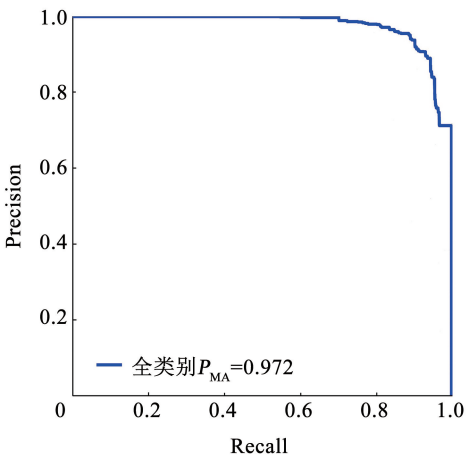


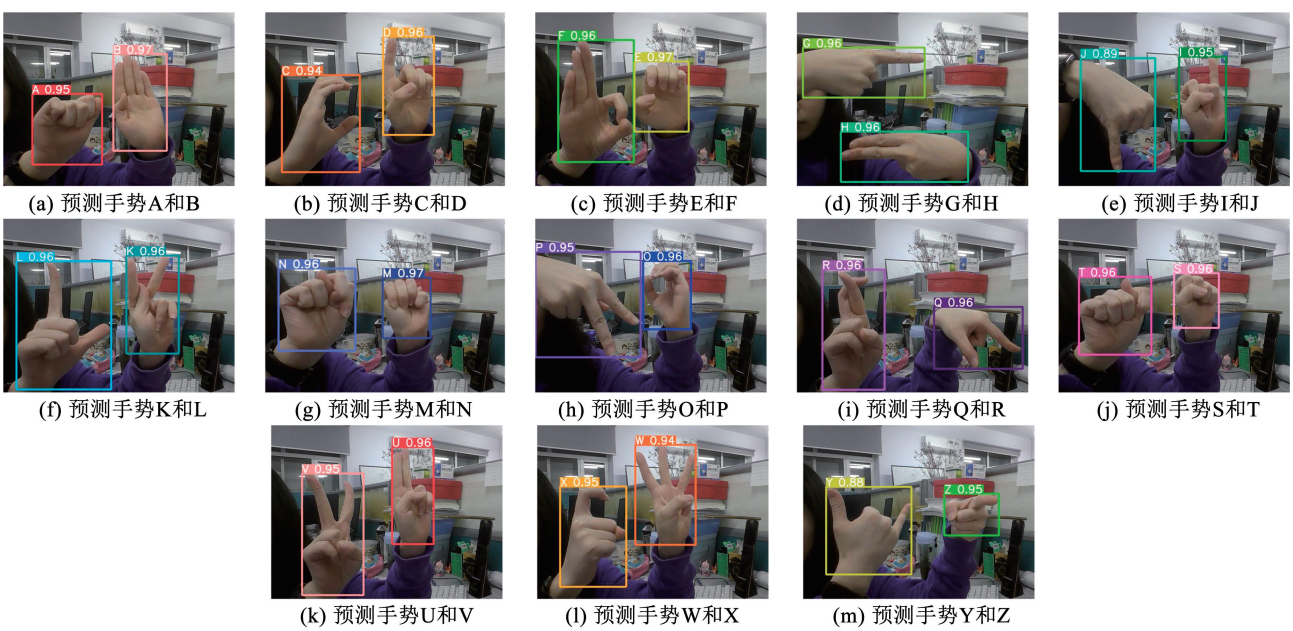
图 6　MPE-YOLOv5 的 $P-R$ 曲线

Fig. 6　$P-R$ curves of MPE-YOLOv5

训练的模型对 26 种手势检测识别效果如图 7 所示。可以看出，该算法能够准确检测出测试集上的 A ~ Z 的 26 种手势，并且能够实现左右手识别以及多手势目标识别，在复杂场景下有较好的鲁棒性。

图 8 所示分别为轻量化模型 M-YOLOv5 和本文所提出的改进模型 MPE-YOLOv5 的训练损失值对比曲线。

从图 8(a)中可知，在训练迭代次数为 0 到25 轮时，2 个模型的边界框损失值接近，随着训练迭代次数的增加，MPE-YOLOv5 的边界框损失优于 M-YOLOv5；从图 8(b)中可知，在训练迭代次数为 0 到 10 轮时，2 个模型的分类损失值大致相同，随着训练迭代次数的增加，虽然 2 个模型的分类损失都收敛于 0.21，但 MPE-YOLOv5 的收敛速度明显较快；从图 8(c)中可知，MPE-YOLOv5 的置信度损失值优于 M-YOLOv5，并且收敛于 0.006。因此，本文所提出的 MPE-YOLOv5 网络模型能够较好地改善 M-YOLOv5 由于轻量化而带来的负面影响。



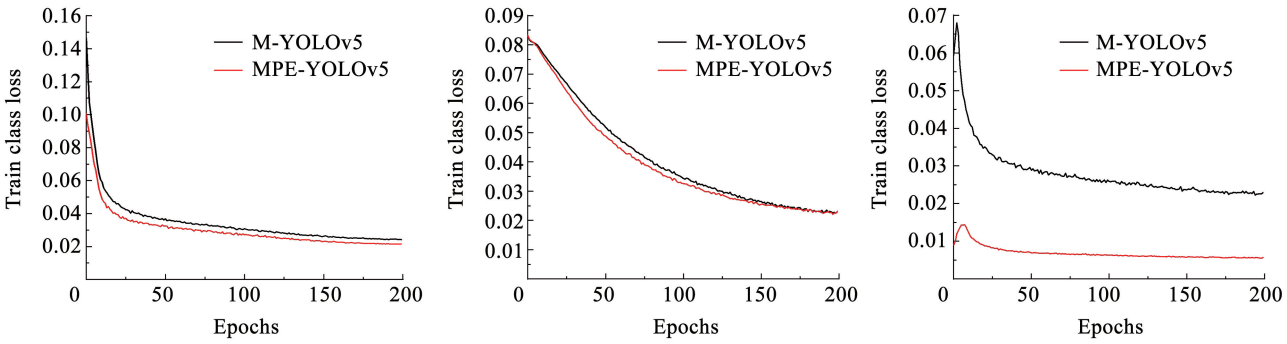
(a) 预测手势A和B　(b) 预测手势C和D　(c) 预测手势E和F　(d) 预测手势G和H　(e) 预测手势I和J

(f) 预测手势K和L　(g) 预测手势M和N　(h) 预测手势O和P　(i) 预测手势Q和R　(j) 预测手势S和T

(k) 预测手势U和V　(l) 预测手势W和X　(m) 预测手势Y和Z

图 7　双手检测结果

Fig. 7　Detection results of two hands



(a) 边框回归损失值　(b) 分类回归损失值　(c) 置信度回归损失值

图8　训练损失值对比曲线

Fig. 8　Comparison curves of training loss value

为充分验证MPE-YOLOv5的有效性和鲁棒性，将网络模型在开源手势数据集NUS-Ⅱ[29]上进行训练、验证与测试。NUS-Ⅱ数据集中包含字母A～J这10类手势，每个类别各有200张复杂背景图像和75张人为干扰图像，共2 750张图像。为了避免训练过程中出现过拟合，使用平移、镜像、增加噪声、调节亮度、裁剪等方法进行数据增强，最终的数据集样本大小为13 750张，并按照6∶2∶2的比例划分训练集、验证集和测试集。

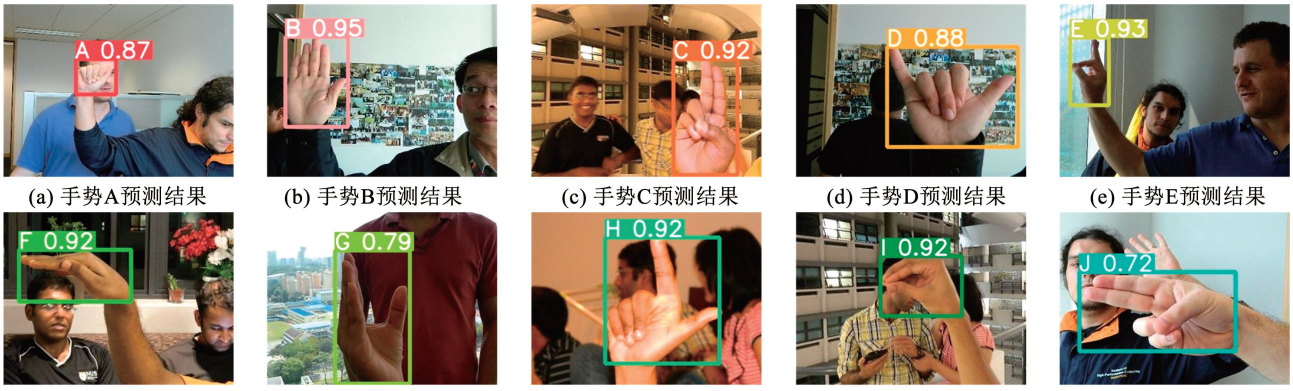
MPE-YOLOv5算法在NUS-Ⅱ测试集上的检测结果分别见表5和图9。

表5、图9表明，在复杂场景下，MPE-YOLOv5算法能够忽略光线明暗、局部手部遮挡、手脸重叠等对手势识别所带来的干扰，实现对手势特征的提取，检测的过程中能够聚焦手势并且准确识别出10种类别，同时也验证了MPE-YOLOv5算法的有效性和鲁棒性。

表5　MPE-YOLOv5算法在NUS-Ⅱ数据集上的检测结果

Tab. 5　Detection results of MPE-YOLOv5 algorithm on NUS-Ⅱ dataset

类别	P	R	P_{MA}	类别	P	R	P_{MA}
A	0.914	0.941	0.955	G	0.995	0.933	0.988
B	0.980	1.000	0.995	H	0.937	1.000	0.995
C	0.934	0.955	0.988	I	0.897	0.846	0.937
D	0.971	1.000	0.995	J	0.935	0.653	0.965
E	1.000	0.992	0.995	All	0.990	0.985	0.994
F	1.000	0.917	0.958				



(a) 手势A预测结果　(b) 手势B预测结果　(c) 手势C预测结果　(d) 手势D预测结果　(e) 手势E预测结果

(f) 手势F预测结果　(g) 手势G预测结果　(h) 手势H预测结果　(i) 手势I预测结果　(j) 手势J预测结果

图9　MPE-YOLOv5算法在NUS-Ⅱ测试集上的检测样例

Fig. 9　Examples of detection samples of MPE-YOLOv5 algorithm on NUS-Ⅱ dataset

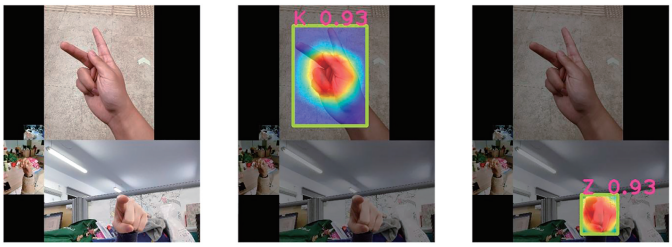
3.3.2　可视化结果与分析

如图10所示，依次为YOLOv5、M-YOLOv5、ME-YOLOv5和所提出的新算法MPE-YOLOv5采用Grad-CAM[30]可视化结果的输出特征图。在检测样本中，包含了4种不同大小的目标手势，分别为大目标手势“K”、中目标手势“Z”、小目标手势“O”和微小目标手势“R”。



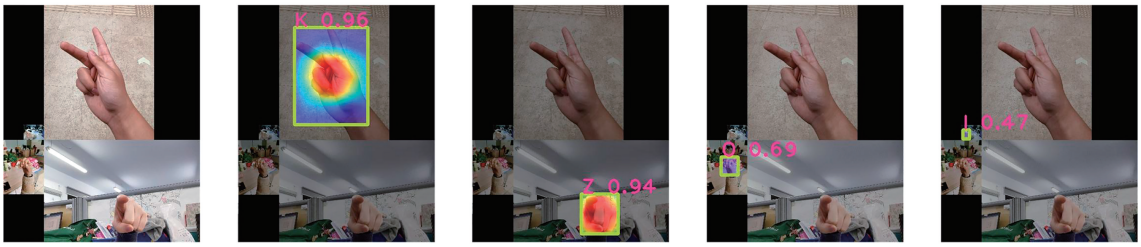
检测样本　K概率为0.98　Z概率为0.98　O概率为0.96　R概率为0.94

(a) YOLOv5模型的Grad-CAM可视化结果



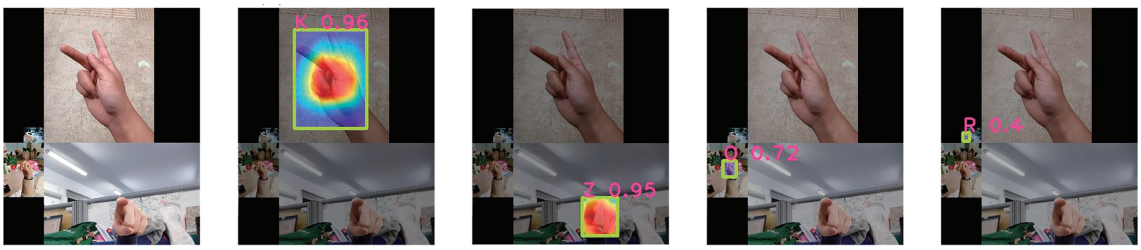
检测样本　K概率为0.93　Z概率为0.93

(b) M-YOLOv5模型的Grad-CAM可视化结果



检测样本　K概率为0.96　Z概率为0.94　O概率为0.69　I概率为0.47

(c) ME-YOLOv5模型的Grad-CAM可视化结果



检测样本　K概率为0.96　Z概率为0.95　O概率为0.72　R概率为0.40

(d) MPE-YOLOv5模型的Grad-CAM可视化结果

图 10　各网络模型检测样本 Grad-CAM 可视化结果

Fig. 10　Grad-Cam visualization results of detection samples of each network model

如图 10 所示，采用经典目标识别模型 YOLOv5 可以实现对大、中、小和微小 4 个不同尺寸手势目标的准确识别，在本地 CPU 上的推理时间为 0.850 0 s，其推理时间相对较长；采用轻量化模型 M-YOLOv5 网络架构对手势目标进行检测时，由于网络的参数量较少，进行特征提取时提取出的特征较少，因此存在不能准确识别和分类的情况。但也由于参数量减少，因此推理速度得到大幅度提升，在本地 CPU 上的推理时间为 0.416 5 s；ME-YOLOv5 网络架构缓解了信息丢失的现象，因此在对小目标检测时，精度得到提升，但对微小目标的识别和分类并没有明显改善，虽然定位和识别了微小目标，但对微小目标进行分类时，出现了错误分类的情况。集成注意力机制会使整体网络模型的参数量略微增加，因此推理时间也随之增加，该模型在本地 CPU 上的推理时间为 0.452 4 s。本文所提出的新算法 MPE-YOLOv5 增加了针对微小目标的检测层，以提升对小目标和微小目标的识别和分类精度，但引入检测层会给网络模型带来更多的参数量，因此推理时间增加，改进模型在本地 CPU 上的推理时间为 0.698 9 s。

3.3.3　算法对比与分析

为验证本文所提出的 MPE-YOLOv5 网络模型在手势识别中的有效性，将该模型与主流的轻量化网络和经典的 YOLO 系列网络进行比较实验。该实验以模型复杂度（参数量 P_{param}、计算复杂度 F_{FLOPs}、模型大小 W_{size}）、平均精度 P_{MA} 和 F_{frame} 作为评价指标。各模型在测试集上的检测结果见表 6。

表 6 MPE-YOLOv5 与其他网络模型的对比结果

Tab. 6 Comparison between MPE-YOLOv5 and other network models

网络类型	网络模型	$P_{\text{param}}/10^6$	$F_{\text{FLOPs}}/10^9$	W_{size}/MB	P_{MA}/%	F_{frame}/(frame · s^{-1})
轻量化网络	ShuffleNetv2[13]	3.24	6.0	6.45	93.6	526
	MobileNetv3[14]	1.41	2.3	2.98	94.2	476
	GhostNet[15]	5.42	8.4	10.70	95.6	286
	EfficientNet[31]	3.83	7.4	7.64	96.3	286
	PP-LCNet[32]	3.38	6.3	6.70	92.8	526
YOLO 系列网络	YOLOv3[22]	9.37	23.4	18.10	93.9	208
	YOLOv4[23]	9.17	20.8	17.80	94.2	233
	YOLOv5[16]	7.08	16.0	13.70	99.4	313
	MPE-YOLOv5	1.50	4.4	3.47	97.2	345

从表 6 中可知,与轻量化网络 ShuffleNetv2[13]、MobileNetv3[14]、GhostNet[15]、EfficientNet[31] 和 PP-LCNet[32] 相比,除 MobileNetv3 外,MPE-YOLOv5 的模型复杂度最低。同时,在有效性方面,MPE-YOLOv5 模型的 P_{MA} 均优于表中的 5 类轻量化网络,相比之下,分别有 3.6%、3.0%、1.6%、0.9% 和 4.4% 的提升。而对于 MobileNetv3 网络,MPE-YOLOv5 用较小的模型复杂度代价换取了较高的精度提升。

与 YOLO 系列网络相比,MPE-YOLOv5 的推理速度相较于经典的 YOLOv3、YOLOv4 和 YOLOv5 网络分别提高了 1.66 倍、1.48 倍和 1.10 倍。MPE-YOLOv5 与原有的 YOLOv5 算法相比,平均精度仅低 2.2%,但参数量、计算复杂度和模型大小明显下降,在效率方面得到一定的提升。

如表 7 所示,将近 3 年来所提出的轻量化手势识别算法与本文所提出的 MPE-YOLOv5 算法进行比较。

表 7 最新的轻量化手势识别算法与 MPE-YOLOv5 的比较

Tab. 7 Comparison between the latest lightweight gesture recognition algorithms with MPE-YOLOv5

文献	网络模型	数据集	P_{MA}/%	W_{size}/MB	$P_{\text{param}}/10^6$	推理时间/ms
文献[33]	YOLOv3-MobileNetv2:YOLOv3 算法结合 MobileNetv2	NUS-Ⅱ数据集[29]	92.7		5.00	33.33
文献[34]	轻量化 I3D:时空可分离 3D 卷积结合 Fire 模块	Jester 数据集[35-36]	94.8	—	3.17	4.23
文献[37]	RF-Net:改进的 AlexNet	Marcel 数据集[38]	83.6	8.93	2.28	1.51
文献[39]	改进的 YOLOv3	自制 5 分类手势数据集	97.7	—	0.15	—
文献[40]	MobileNetv3-SSDLite:SSD 算法结合 MobileNetv3	自制 5 分类手势数据集	99.6	—	2.20	17.27
文献[41]	改进的 EfficientDet:EfficientDet 算法结合 MobileNet 和 SE 注意力机制	自制 74 分类手势数据集	94.1	18.40	—	90.91
文献[42]	改进的 Xception:Xception 算法结合 DenseNet 和 SE 注意力机制	NUS-Ⅱ数据集[29]	99.6	54.19	4.31	—
本文	MPE-YOLOv5	自制 26 分类手势数据集[27] NUS-Ⅱ数据集[29]	97.2 99.4	3.47	1.50	2.94

现有的轻量化手势识别算法主要以单阶段的目标检测模型为总体框架,结合各种轻量化方法,设计出轻量化的手势识别网络体系。由于各个模型是在不同的硬件设备上对不同的数据集进行测试的,因此推理速度不具有参考性。但总体来看,本文所提出的 MPE-YOLOv5 网络能够在较低的模型复杂度下,实现准确的手势识别与分类。

综上所述,MPE-YOLOv5 网络可以较好地平衡算法的有效性与效率。

3.3.4 消融实验结果与分析

为了分析不同改进点对网络模型性能的影响,本文进行了消融实验,实验结果如表 8 所示。其中,主干网络轻量化、增加检测层、集成注意力机制、损失函数优化分别对应改进点 1、2、3 和 4。

表 8　消融实验结果

Tab. 8　Ablation experiment results

网络模型	改进点 1	改进点 2	改进点 3	改进点 4	P/%	R/%	P_{MA}/%	D_{mAP}/10^{-8}	P_{param}/10^6	W_{size}/MB	F_{FLOPs}/10^9	F_{frame}/(frame·s^{-1})
YOLOv5	×	×	×	×	99.3	99.0	99.4	14.02	7.09	13.7	16.0	313
M-YOLOv5	√	×	×	×	87.0	81.6	89.4	63.40	**1.41**	**2.97**	**2.3**	**476**
ME-YOLOv5	√	×	√	×	91.0	82.7	92.4	**65.53**	**1.41**	2.99	**2.3**	**476**
MP-YOLOv5	√	√	×	×	92.7	88.2	96.2	64.13	1.50	3.46	4.4	345
MPE-YOLOv5	√	√	√	×	93.3	90.4	96.3	64.20	1.50	3.47	4.4	345
MPE-YOLOv5	√	√	√	√	**94.5**	**91.1**	**97.2**	64.80	1.50	3.47	4.4	345

由表 8 可知，对于 YOLOv5 网络模型，虽然模型检测精度高，平均精度可达 99.3%，但由于模型的参数量大，所需的计算复杂度增大，从计算复杂度和模型占用内存的角度来说，都不适合部署在边缘终端设备上。将 YOLOv5 的特征提取网络轻量化后，效率方面有了较为明显的改善，但效率改善的前提是牺牲有效性。M-YOLOv5 相较于 YOLOv5 而言，平均精度下降了 9.97%，但由于其模型的参数量减少为原来的 19.89%，因此其精度密度是 YOLOv5 网络模型的 4.53 倍。

在 M-YOLOv5 中集成注意力机制模块后，模型的效率基本上没有降低，但 ME-YOLOv5 的平均精度相较于 M-YOLOv5 提升了 3.58%，并且集成注意力机制模块后的推理时间并不会受到影响，与原来的 M-YOLOv5 模型相同。

本文提出的 MPE-YOLOv5 网络模型虽然增加了一个检测层给网络模型带来了参数量，与 M-YOLOv5相比效率降低了，但通过对网络模型效率的牺牲换取有效性取得了一定的成效，其平均精度与 M-YOLOv5 相比提升了 8.72%，并且相较于 YOLOv5，其模型参数量、模型大小和计算复杂度分别是原来的 21.16%、25.33% 和 27.33%，相比之下效率有了明显的提升。YOLOv5 和所提出的新算法 MPE-YOLOv5 平均精度相近，新算法的平均精度损失仅为 2.11%。由此也验证了所提出的 MPE-YOLOv5 算法的可行性。

综上，本文所提出的轻量化 MPE-YOLOv5 手势识别算法，能够较好地平衡模型精度与推理时间之间的关系。

4　结　论

提出了一种面向边缘计算的轻量化 MPE-YOLOv5 手势识别算法。针对复杂的卷积神经网络模型在边缘终端设备部署困难的问题，对经典单阶段目标检测模型 YOLOv5 上进行改进。通过测试所提出的新算法，验证了新算法的可行性，并且平衡了模型的有效性和效率。

以自制数据集上的 26 类手势为例，对所提出的轻量化 MPE-YOLOv5 算法进行测试，实验结果表明，在复杂场景下，该算法可实现对大尺度变换和小目标的手势识别，平均识别精度可达 97.2%，并且轻量化 MPE-YOLOv5 算法的模型参数量、模型大小和计算复杂度为传统 YOLOv5 算法的 21.16%、25.33% 和 27.33%，同时平均识别精度较 M-YOLOv5算法提升了 8.72%。本文所提出的 MPE-YOLOv5 具备识别精度高、网络模型小、计算复杂度低和推理速度快的特点。

所提出的 MPE-YOLOv5 针对大尺度变换目标和微小目标的检测增加了检测层，但增加检测层将会给模型带来一定的参数量，使模型复杂度增加。因此，模型的改进应适应场景的变化。例如，当实际的应用场景中小目标或大尺度变换目标较少时，应适当减少检测层的层数，使得网络更加轻量，模型也能够更好地部署在边缘终端上；同理，当小目标和大尺度变换目标较多并且在算力良好的硬件设备上进行检测时，适当增加检测层的层数，能够使得模型对目标的检测更加精确。

以手势识别为例进行了实验，验证了算法的效率和有效性，但所提出的算法并不仅仅局限于车牌识别、行人识别、人脸识别等目标检测场景，同时也为边缘计算场景下的目标跟踪、语义分割、关键点检测、边缘检测等计算机视觉任务的研究提供了一种解决方案。

参考文献

[1] 林清宇. 基于 Kinect 的手势检测与追踪研究[D]. 南京：南京邮电大学，2020
LIN Qingyu. The research of hand detection and tracking using Kinect [D]. Nanjing: Nanjing University of Posts and Telecommunications, 2020. DOI:10.27251/d.cnki.gnjdc.2020.001062

[2] 司潇. 基于 LeapMotion 的采煤场景协同手势控制虚拟仿真研究

与实现[D]. 青岛: 山东科技大学, 2020
SI Xiao. Research and implementation of virtual simulation of cooperative gesture control of coal mining scene based on LeapMotion [D]. Qingdao: Shandong University of Science and Technology, 2020. DOI:10.27275/d.cnki.gsdku.2020.001459

[3] CHEN Lulu, WEI Hong, FERRYMAN J. A survey of human motion analysis using depth imagery[J]. Pattern Recognition Letters, 2013, 34(15): 1996. DOI:10.1016/j.patrec.2013.02.006

[4] PANDURANGA H T, MANI C. Dynamic hand gesture recognition system: a short survey [C]//2018 International Conference on Inventive Research in Computing Applications. Coimbatore: IEEE, 2018: 690

[5] PRASUHN L, OYAMADA Y, MOCHIZUKI Y, et al. A HOG-based hand gesture recognition system on a mobile device[C]//2014 IEEE International Conference on Image Processing. Paris: IEEE, 2014: 3974

[6] SUTTAPAK W, AUEPHANWIRIYAKUL S, THEERA-UMPON N. Incorporating SIFT with hard C-means algorithm[C]//2010 The 2nd International Conference on Computer and Automation Engineering. Singapore: IEEE, 2010: 438

[7] SYKORA P, KAMENCAY P, HUDEC R. Comparison of SIFT and SURF methods for use on hand gesture recognition based on depth map[J]. AASRI Procedia, 2014, 9: 20. DOI:10.1016/j.aasri.2014.09.005

[8] FU Yan, WANG Mei, ZHANG Chunqin. SAR image target recognition based on Hu invariant moments and SVM [C]//2009 Fifth International Conference on Information Assurance and Security. Xi'an: IEEE, 2009: 586

[9] DING Youdong, PANG Haibo. An improved algorithm of hand-gesture recognition based on Haar-like features and Adaboost[C]// 2011 World Congress on Engineering and Technology. Shanghai: IEEE, 2012: 1239

[10] 彭玉青, 赵晓松, 陶慧芳, 等. 复杂背景下基于深度学习的手势识别[J]. 机器人, 2019, 41(4): 538
PENG Yuqing, ZHAO Xiaosong, TAO Huifang, et al. Hand gesture recognition against complex background based on deep learning[J]. Robot, 2019, 41(4): 538. DOI:10.13973/j.cnki.robot.180568

[11] 王粉花, 黄超, 赵波, 等. 基于 YOLO 算法的手势识别[J]. 北京理工大学学报, 2020, 40(8): 877
WANG Fenhua, HUANG Chao, ZHAO Bo, et al. Gesture recognition based on YOLO algorithm[J]. Transactions of Beijing Institute of Technology, 2020, 40(8): 877. DOI: 10.15918/j.tbit1001-0645.2019.030

[12] 王银, 陈云龙, 孙前来. 复杂背景下的手势识别[J]. 中国图象图形学报, 2021, 26(4): 822
WANG Yin, CHEN Yunlong, SUN Qianlai, et al. Hand gesture recognition in complex background [J]. Journal of Image and Graphics, 2021, 26(4): 822. DOI:10.11834/jig.200211

[13] MA Ningning, ZHANG Xiangyu, ZHENG Haitao, et al. ShuffleNet V2: practical guidelines for efficient CNN architecture design[C]// FERRARI V, HEBERT M, SMINCHISESCU C, et al. Computer Vision-ECCV 2018-15th European Conference. Munich: Springer International Publishing, 2018: 124

[14] HOWARD A, SANDLER M, CHEN Bo, et al. Searching for MobileNetV3[C]//2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019: 1315

[15] HAN Kai, WANG Yunhe, TIAN Qi, et al. GhostNet: more features from cheap operations[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 1582

[16] ULTRALYTICS. YOLOv5[EB/OL]. [2022-06-14]. https://github.com/ultralytics/yolov5

[17] HOWARD A, ZHU Menglong, CHEN Bo, et al. MobileNets: efficient convolutional neural networks for mobile vision applications [Z/OL]. arXiv:1704.04861, 2017[2022-07-09]. http://arxiv.org/abs/1704.04861. DOI:10.48550/arXiv.1704.04861

[18] SANDLER M, HOWARD A, ZHU Menglong, et al. MobileNetV2: inverted residuals and linear bottlenecks [C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 4512

[19] HU Jie, SHEN Li, ALBANIE S, et al. Squeeze-and-excitation networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(8): 2013. DOI:10.1109/TPAMI.2019.2913372

[20] JOSEPHR, SANTOSH D, ROSS G, et al. You only look once: unified, real-time object detection[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE Computer Society, 2016: 780

[21] REDMON J, FARHADI A. YOLO9000: better, faster, stronger [C]//2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017: 6518

[22] REDMON J, FARHADI A. YOLOv3: an incremental improvement [Z/OL]. arXiv:1804.02767, 2018[2022-07-09]. http://arxiv.org/abs/1804.02767. DOI:10.48550/arXiv.2004.10934

[23] BOCHKOVSKIY A, WANG Jieyao, LIAO Hongyuan. YOLOv4: optimal speed and accuracy of object detection [Z/OL]. arXiv: 2004.10934, 2020[2022-07-09]. https://arxiv.org/abs/2004.10934. DOI:10.48550/arXiv.2004.10934

[24] WANG Qilong, WU Banggu, ZHU Pengfei, et al. ECA-Net: efficient channel attention for deep convolutional neural networks [C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 11533

[25] ZHENG Zhaohui, WANG Ping, REN Dongren, et al. Enhancing geometric factors in model learning and inference for object detection and instance segmentation[J]. IEEE Transactions on Cybernetics, 2022, 52(8): 8578. DOI:10.1109/TCYB.2021.3095305

[26] ZHANG Yifan, REN Weiqiang, ZHANG Zhang, et al. Focal and efficient IOU loss for accurate bounding box regression[Z/OL]. arXiv:2101.08158, 2021[2022-07-27]. https://arxiv.org/abs/2101.08158. DOI:10.48550/arXiv.2101.08158

[27] Gesture Dataset[EB/OL]. [2022-08-10]. https://github.com/KayViolet/Gesture-Dataset

[28] CANZIANI A, PASZKE A, CULURCIELLO E. An analysis of deep neural network models for practical applications [Z/OL]. arXiv:1605.07678, 2016[2022-07-27]. http://arxiv.org/abs/1605.07678. DOI:10.48550/arXiv.1605.07678

[29] PISHARADY P K, VADAKKEPAT P, LOH A P. Attention based detection and recognition of hand postures against complex backgrounds[J]. International Journal of Computer Vision, 2013, 101(3): 413. DOI:10.1007/s11263-012-0560-5

[30] SELVARAJU R R, COGSWELL M, DAS A, et al. Grad-CAM:

visual explanations from deep networks via gradient-based localization[C]//2017 IEEE International Conference on Computer Vision. Venice: IEEE, 2017: 622

[31] TAN Mingming, LE Q V. EfficientNet: rethinking model scaling for convolutional neural networks [C]//Proceedings of the 36th International Conference on Machine Learning. Long Beach: PMLR, 2019: 6107

[32] CUI Cheng, GAO Tingquan, WEI Shengyu, et al. PP-LCNet: a lightweight CPU convolutional neural network [Z/OL]. arXiv: 2109.15099, 2021 [2022-07-09]. https://arxiv.org/abs/2109.15099. DOI:10.48550/arXiv.2109.15099

[33] 熊才华. 基于深度学习的手势识别算法研究与应用[D]. 杭州：浙江理工大学, 2019
XIONG Caihua. Research and application of gesture recognition algorithm based on deep learning[D]. Hangzhou: Zhejiang Sci-Tech University, 2019. DOI: 10.27786/d.cnki.gzjlg.2019.000381

[34] LU Zhi, QIN Shiyin, LI Lianwei, et al. One-shot learning hand gesture recognition based on lightweight 3D convolutional neural networks for portable applications on mobile systems[J]. IEEE Access, 2019, 7: 131742. DOI: 10.1109/ACCESS.2019.2940997

[35] Twentybn Jester Dataset: images[EB/OL]. [2022-07-25]. https://www.kaggle.com/datasets/kylecloud/20bn-jester-v1-videos

[36] Twentybn Jester Dataset: tables [EB/OL]. [2022-07-25]. https://www.kaggle.com/datasets/kylecloud/jester-csv

[37] 张迪. 基于机器学习的手势识别技术研究[D]. 南京：南京邮电大学, 2019
ZHANG Di. Research on hand gesture recognition technology based on machine learning[D]. Nanjing: Nanjing University of Posts and Telecommunications, 2019. DOI:10.27251/d.cnki.gnjdc.2019.000477

[38] MARCEL S, BERNIER O. Hand posture recognition in a body-face centered space[C]//BRAFFORT A, GHERBI R, GIBET S, et al. Gesture-Based Communication in Human-Computer Interaction. [S.l.]: Springer, 1999: 303

[39] MUJAHID A, AWAN M J, YASIN A, et al. Real-time hand gesture recognition based on deep learning YOLOv3 model[J]. Applied Sciences, 2021, 11(9): 4164. DOI: 10.3390/app11094164

[40] 杨晨奕，何玉青，赵俊媛，等. 面向嵌入式平台的轻量化神经网络手势识别方法[J]. 强激光与粒子束, 2022, 34(3): 148
YANG Chenyi, HE Yuqing, ZHAO Junyuan, et al. Lightweight neural network hand gesture recognition method for embedded platforms[J]. High Power Laser and Particle Beams, 2022, 34(3): 148. DOI:10.11884/HPLPB202234.210335

[41] 杨淑莹，赵敏，郭杨杨，等. 基于改进的 EfficientDet 的手语识别算法[J]. 微电子学与计算机, 2022, 39(2): 89
YANG Shuying, ZHAO Min, GUO Yangyang, et al. Sign language recognition algorithm based on improved EfficientDet [J]. Microelectronic & Computer, 2022, 39(2): 89. DOI:10.19304/j.issn1000-7180.2021.0751

[42] 周梓豪，田秋红. 基于改进 Xception 网络的手势识别[J]. 软件导刊, 2022, 21(6): 46
ZHOU Zihao, TIAN Qiuhong. Gesture recognition based on improved Xception network[J]. Software Guide, 2022, 21(6): 46. DOI:10.11907/rjdk.221262

（编辑　苗秀芝）

封面图片说明

封面图片来自本期论文“融合大气光值-图估计的无人机航拍图像去雾”，是长安大学电子与控制工程学院黄鹤课题组提出的一种改进大气散射模型。该模型从成像过程开始分析，把参与图像成像的光线分为大气光和反射光两部分，将单一大气光值估计设计为受场景影响的大气光图估计。在此基础上，提出了基于自适应随机游走聚类大气光图估计、基于最小偏差颜色衰减先验的大气光值估计以及融合大气光值-图估计三种方法。融合后的大气光图能精准复原大气光部分，提升图像去雾效果，有效降低天空区域的失真。最后提出了一种基于暗补偿的透射率优化方法，以提升反射光部分对细节的复原能力。对比分析了算法性能，主观上亮度和饱和度有明显提高，复原图像更加自然清晰，且客观评价指标信息熵、平均梯度、模糊系数和对比度等也有明显提升，这对改善复原无人机航拍图像质量有着重要的研究意义，也为后续研究提供了可行的思路。

（图文提供：黄鹤、李战一、胡凯益、王会峰、茹锋、王珺. 长安大学电子与控制工程学院）