

# 基于改进 YOLO v4 的安全帽佩戴检测算法

金雨芳 吴祥 董辉 俞立 张文安

浙江工业大学信息工程学院 杭州 310023

浙江省嵌入式系统联合重点实验室 杭州 310023

(m17681535896@163.com)

**摘要** 安全生产管理是建筑、重工业等高危企业发展的重要方针,安全帽在施工生产环境中对人员头部防护起着关键作用,因此加强安全帽佩戴监管十分必要。近年来,基于图像视觉的安全帽佩戴监测方法成为了企业实施管理的主要手段,如何提高安全帽佩戴检测精度和检测速度是应用的关键难题。针对上述问题,文中提出了一种基于改进 YOLO v4 的安全帽佩戴检测算法。首先,在 YOLO v4 算法的 3 个特征图输出的基础上增加了  $128 \times 128$  特征图输出,从而将特征图输出的 8 倍下采样改为 4 倍下采样,为后续特征融合提供了更多小目标特征。其次,基于密集连接的思想对特征融合模块进行改进以实现特征重用,使得负责小目标检测的 Yolo Head 分类器可以结合不同层次特征层的特征,从而得到更好的目标检测分类结果。最后,对比实验的结果表明,所提方法的平均精度高达 91.17%,相比原网络检测精度提高了 2.96%,检测速度基本不变,可达 52.9 frame/s,从而在满足实时检测需求的同时可以得到更优的检测精度,有效实现了安全帽佩戴的高速高精度检测。

**关键词:** 深度学习;小目标检测;安全帽佩戴检测;YOLO v4;K-means 聚类

中图法分类号 TP391

## Improved YOLO v4 Algorithm for Safety Helmet Wearing Detection

JIN Yu-fang, WU Xiang, DONG Hui, YU Li and ZHANG Wen-an

College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China

United Key Laboratory of Embedded System of Zhejiang Province, Hangzhou 310023, China

**Abstract** Safety production management is an important policy for the development of high-risk enterprises such as the construction industry and heavy industry, and safety helmets play a key role in head protection in the production environment. Therefore, it is necessary to strengthen the supervision of helmet wearing. In recent years, the monitoring method of helmet wearing based on image vision has become the main means for enterprises to implement management. How to improve the detection accuracy and speed of helmet wearing is a crucial issue for applications. To deal with this issue, an improved YOLO v4 algorithm is proposed to promote the accuracy and efficiency of safety helmet wearing detection in this paper. First, a  $128 \times 128$  feature map output is added to the original three feature map outputs of the YOLO v4 algorithm, and the 8 times downsampling of the feature map output is changed to 4 times downsampling to provide more small target features for subsequent feature fusion. Second, the feature fusion module is improved based on the idea of dense connection to realize feature reuse, so that the Yolo Head classifier responsible for small target detection can utilize the features of different levels, to obtain better target detection and classification results. Finally, comparative experiments are carried out. The results show that the average accuracy of the proposed method is 2.96% higher than the original network detection accuracy to be 91.17%, and the detection speed is basically unchanged to be 52.9 frame/s. Thereupon, the proposed algorithm can achieve better detection accuracy while meeting real-time detection requirements, and effectively realize the high-speed and high-precision detection of helmet wearing.

**Keywords** Deep learning, Small target detection, Safety helmet wearing detection, YOLO v4, K-means clustering

到稿日期:2020-09-13 返修日期:2021-02-09

基金项目:浙江省重点研发计划(2020C01109);NSFC-浙江两化融合联合基金(U1709213);浙江新苗人才计划(GZ20411030017)

This work was supported by the Key Research and Development Program of Zhejiang Province(2020C01109), NSFC-Zhejiang Joint Fund for the Integration of Industrialization and Development Plan of Zhejiang Province(U1709213) and Zhejiang Xinmiao Talent Projects(GZ20411030017).

通信作者:吴祥(xiangwu@zjut.edu.cn)

## 1 引言

安全生产管理是建筑、重工业等高危企业发展的重要方针,安全帽在施工生产环境中对人员头部防护起着关键作用。然而,由于施工人员流动性大、安全意识匮乏以及监管人员不到位等多种因素,导致实际工作中施工人员未佩戴安全帽的不安全行为时有发生,带来了极大的安全隐患。因此,安全帽佩戴监管是高危作业环境中不可或缺的环节。传统依靠人工监管的方式存在效率低下、管理范围有限、时效性差、无法全场监测等诸多缺陷<sup>[1-2]</sup>,因此基于图像视觉的安全帽佩戴监测方法逐渐成为企业实施管理的主要手段<sup>[3]</sup>。

安全帽佩戴检测是一类典型的目标检测问题,通常将佩戴安全帽的人和不佩戴安全帽的人作为两类进行检测。由于不同物体具有不同的外观、形状、姿态且成像时受光照、遮挡等因素影响,因此目标检测是计算机视觉领域最具挑战性的问题之一。传统的目标检测方法需要手工设计目标特征,该方法不仅没有特征针对性,而且会造成大量冗余的预选框生成,导致学习效率低下,对多样性变化的鲁棒性差<sup>[4]</sup>。随着人工智能的快速发展,基于神经网络和深度学习的目标检测技术具有检测速度快、精度高等诸多优点,成为了当下目标检测技术的主流手段<sup>[5]</sup>。现有基于深度学习的目标检测方法主要分为两类:1)是以 R-CNN<sup>[6-9]</sup> 系列算法为代表的 two-stage 算法;2)是以 SSD<sup>[4]</sup>, YOLO<sup>[10-13]</sup> 和 RetinaNet<sup>[14]</sup> 系列为代表的 one-stage 算法。此外,近年来研发的 one-stage 检测器,如 CenterNet<sup>[15]</sup>, CornerNet<sup>[16-17]</sup> 和 FCOS<sup>[18]</sup>,在特征提取网络和分类器中间插入若干特征融合层,通过采集不同阶段的特征图来提高目标检测的精度。

鉴于安全帽佩戴检测的重要性,利用图像视觉和神经网络算法对安全帽佩戴情况进行检测一直是研究的热点,国内外许多学者已对此做了大量工作<sup>[3,19-21]</sup>,R-CNN 系列、SSD 和 YOLO 系列等算法在安全帽检测中都得到了广泛应用。然而,在实际应用中,由于安全帽佩戴检测外部环境通常为建筑工地、大型工厂、车间等复杂环境,目标遮挡、光线等因素极易影响检测精度<sup>[22]</sup>。另一方面,在安全帽实时监测的视频采集原始图像中,安全帽目标相比图像尺寸通常占比很小,往往会造成目标漏检<sup>[23]</sup>。因此,如何提高复杂环境下的安全帽佩戴检测精度和检测速度是应用的关键。

针对上述难题,本文提出了一种基于改进 YOLO v4 的安全帽佩戴检测方法,在 YOLO v4 算法的基础上结合密集连接思想,通过对特征融合模块结构的改进,可在保证检测速度的前提下提高检测精度。相比现有安全帽佩戴检测算法,所提算法在安全帽小目标检测中可得到更优的检测结果。本文的主要贡献如下:

(1)提出了一种基于改进 YOLO v4 的小目标检测算法,优化了特征图输出和特征融合模块,在不增加算法计算复杂度的同时,可有效提高小目标检测精度。

(2)实现了基于改进 YOLO v4 小目标检测算法的复杂环境下的安全帽佩戴自动检测功能,相比现有安全帽检测算法可得到更优的检测结果。

## 2 相关工作

随着深度学习技术的不断发展,越来越多的网络模型被提出,利用现有模型进行训练,就可得到准确高效的深度学习目标检测器。其中,YOLO 算法具有结构简单、速度快、鲁棒性强等诸多优点,在算法研究和实际应用中都得到了广泛关注,且随着其版本的不断更新,算法性能得到了进一步的提升<sup>[13]</sup>。Yang 等<sup>[19]</sup>提出了基于 YOLO v3 的安全帽佩戴检测算法,利用 YOLO v3 对人脸进行识别,推算出目标潜在区域,再进行安全帽特征提取,并使用 SVM 分类器对检测结果进行分类,可有效地实时检测施工现场的安全帽佩戴情况。Fang 等<sup>[3]</sup>提出基于改进 YOLO v2 的安全帽佩戴检测算法,在特征提取网络 8 个卷积层后增加了密集网络,并采用 MobileNet<sup>[17]</sup> 结构压缩网络模型,在保证检测精度不下降的同时检测速度可高达 148 frame/s。Lin 等<sup>[2]</sup>将安全帽检测问题变换为单分类问题,从而减小网络计算开销,提高检测速度,同时改善了安全帽漏检问题。Zhao 等<sup>[24]</sup>采用 YOLO v3 对安全帽、彩色背心及其颜色进行实时检测,并使用卡尔曼滤波和匈牙利算法对施工人员进行轨迹跟踪,可有效检测施工人员安全帽佩戴、背心穿戴、活动轨迹的情况。Wu 等<sup>[25]</sup>针对安全帽检测问题,采用 DenseNet<sup>[26]</sup> 网络取代了 YOLO v3 中的 Darknet53 特征提取网络,可有效提高检测精度。Xu 等<sup>[27]</sup>研究了一种安全帽佩戴检测的图像描述方法,采用基于 YOLO v3 与语义规则和语句模板相结合的方式生成安全帽佩戴图像描述语句,有效实现了精确的安全帽佩戴情况描述。上述安全帽检测算法在检测精度和检测速度方面各有优势。然而安全帽佩戴检测系统得到的现场图像中安全帽目标通常占比较小,上述算法对此类小目标检测的精度和效率无法得到有效保障。

安全帽佩戴检测中的小目标问题也一直是研究的热点。Zhang 等<sup>[23]</sup>提出了一种改进的 YOLO v2 小目标检测算法,通过构建特征金字塔 (Feature Pyramid Networks, FPN) 来解决施工现场安全帽佩戴检测的图像描述问题,提高了小目标检测精度,同时能生成较为准确的图像描述语句,但检测速度相对较慢。Long 等<sup>[20]</sup>提出了基于 SSD 的安全帽佩戴小目标检测算法,首先通过 SSD 检测人体区域,预测出安全帽的大概位置,再通过安全帽精确检测模块并结合相邻帧检测结果来提高对安全帽的检测精度,然而 FPS 仅为 21.6,无法达到实时监测的需求。Wu 等<sup>[22]</sup>提出了基于改进 SSD 的工地安全帽佩戴检测算法,提出了一种 RPA (Reverse Progressive Attention) 模块,用于解决小目标识别精度低的问题,但其大大降低了监测速度,并提供了一个能通过安全帽颜色来分辨佩戴者身份信息的公开数据集。Liu 等<sup>[21]</sup>将 FPN 网络融入到 YOLO 算法中,在卷积层上进行滑动窗口操作,并采用先池化再卷积的方法来减少图像特征丢失,从而提高安全帽小目标检测能力。但由于网络复杂度的增加,导致其无法满足实时检测需求。Gu 等<sup>[28]</sup>针对施工现场的安全帽检测问题,在 Faster R-CNN 算法的基础上添加了一组小尺寸的锚点 (64×64),并提出 OHEM (Online Hard Example Mining) 模块,将原始的 ROI 网络扩展成两个,以增强小目标检测能力,

同时增加了算法复杂度。Fang 等<sup>[29]</sup>提出了一种针对远场视频检测条件下的复杂环境中非安全帽检测方法,制作了一个满足建筑工地环境多样化特征的数据集,采用 Faster R-CNN 算法实现远场情况下的安全帽佩戴小目标检测,然而结果显示其检测速度小于 5 frame/s。

综上所述,安全帽佩戴检测问题得到了持续广泛的研究,然而现有算法在提高安全帽小目标检测精度的同时都在一定程度上牺牲了检测速度。如何兼顾安全帽佩戴小目标检测的精度和速度是关键。为此,本文提出了一种基于改进 YOLO v4 的安全帽佩戴检测法。首先,在 YOLO v4 的 3 个特征图输出的基础上加入  $128 \times 128$  特征图,可以提供更多的小目标特征信息,从而增强网络小目标检测能力;其次,采用密集连接的思想对特征融合模块进行改进,实现特征重用以增强小目标检测能力。所提方法的改进并未增加网络复杂度,可在不牺牲检测速度的前提下有效提高检测精度。

### 3 本文算法

本文旨在实时检测安全生产环境中的安全帽佩戴情况。由于实际施工环境复杂,且人员工作时的姿态多变,导致安全帽佩戴检测精度受到挑战,从而影响了安全监控系统的正常作用,因此设计高准确性和稳定性的实时安全帽佩戴识别算法有着非常重要的实际意义。本文提出了一种基于改进 YOLO v4 的小目标检测算法,实现了对复杂环境中安全帽佩戴的实时高精度检测。本节首先简单介绍了 YOLO v4 算法,随后详细描述了所提算法。

#### 3.1 YOLO v4 算法

YOLO 系列算法的基本思想是将输入图像分割为  $S \times S$  个单元格,且每个单元格生成  $B$  个边界框,由被检测目标中心点所在的单元格负责该目标的检测,并计算对应边界框的置信度。YOLO v4 算法利用 3 种不同尺度的特征图对图像进行多尺度预测。首先将 3 种尺度的特征图分别分割成不同数量的单元格,每个单元格预测 3 个边界框,边界框用  $(x, y, w, h, c)$  5 个参数来表达,其中  $(x, y)$  表示目标中心点相对于所属单元格左上角的坐标,  $(w, h)$  表示目标和输入图像的宽度和高度比值,  $c$  表示通道数。同时,当针对  $C$  个不同类别的目标时,每个边界框会预测各目标类别的所属概率值,因此 YOLO v4 中每个单元格可预测  $3 \times (5 + C)$  个概率值。其次,计算每个边界框的置信度,并去除置信度低于阈值的边界框。最后,YOLO v4 采用 DIOU NMS<sup>[30]</sup> (Distance-IOU Non-Maximum Supression) 对剩余边界框进行过滤,得到最优的目标预测框。其中,边界框置信度的计算式为:

$$Conf(obj) = P_r(obj) \times IOU_{pre}^{truth} \quad (1)$$

$$IOU_{pre}^{truth} = \frac{area(B_{pre} \cap B_{truth})}{area(B_{pre} \cup B_{truth})} \quad (2)$$

其中,  $Conf(obj)$  表示置信度;  $P_r(obj)$  表示目标中心是否在预测框中,如在预测框中则为 1,否则为 0;  $IOU_{pre}^{truth}$  表示真值框和预测框的重合程度;  $B_{pre}$  表示预选框;  $B_{truth}$  表示真值框;  $area(B_{pre} \cap B_{truth})$  表示预选框和真值框相交的面积;  $area(B_{pre} \cup B_{truth})$  表示预选框和真值框并集的面积。

YOLO v4 网络结构主要由 CSPDarknet53 特征提取网络、SPP 模块、PANet<sup>[31]</sup> 特征融合模块、Yolo Head 分类器组成,如图 1 所示。

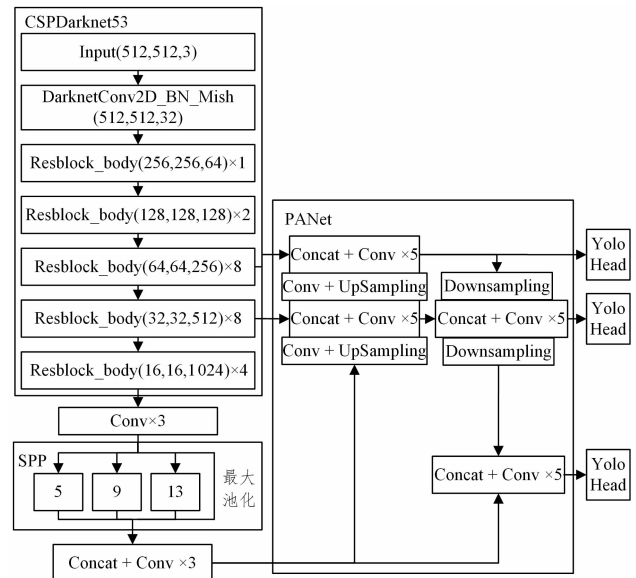


图 1 YOLO v4 算法结构

Fig. 1 Structure of YOLO v4 algorithm

如图 1 所示,首先将原始图像输入网络,经过 CSPDarknet53 特征提取网络,分别进行 8 倍、16 倍和 32 倍下采样得到 3 个不同尺度的特征图。以  $512 \times 512$  大小的训练图像为例,经过下采样得到  $64 \times 64$ ,  $32 \times 32$ ,  $16 \times 16$  这 3 个尺度的特征图。其次,将 CSPDarknet53 特征提取网络输出的  $16 \times 16$  特征图输入 SPP 模块进行处理,SPP 结构连接在 CSPdarknet53 的最后一个特征层的卷积后,再对 CSPdarknet53 的最后一个特征层进行 3 次 DarknetConv2D\_BN\_Leaky 卷积后,分别利用 4 个不同大小 ( $13 \times 13$ ,  $9 \times 9$ ,  $5 \times 5$ ,  $1 \times 1$ ) 的最大池化进行处理,SPP 结构可以有效地增大感受野,分离出最显著的上下文特征。然后,将  $64 \times 64$  特征图、 $32 \times 32$  特征图和 SPP 模块处理过的  $16 \times 16$  特征图输入到 PANet 特征融合模块中进行处理。PANet 特征融合模块与 YOLO v3 中传统的 FPN 结构不同,在完成 FPN 从下到上逐层经过上采样并拼接的特征融合后,还会实现从上到下的特征提取,实现了特征的反复提取,使不同特征层特征相互融合,从而使 3 个不同尺度的特征层都具有较好的表达效果。最后,将 PANet 特征融合模块处理过的  $64 \times 64$ ,  $32 \times 32$ ,  $16 \times 16$  这 3 个尺度的特征图分别输入 3 个相同的 Yolo Head 分类器中进行分类,从而得出检测结果。

#### 3.2 改进 YOLO v4 算法

改进 YOLO v4 算法的网络结构如图 2 所示。为了提高安全帽佩戴检测过程中的小目标检测精度,对 YOLO v4 中的 PANet 特征融合模块进行了改进,利用 DenseNet 网络中密集连接的思想,将特征提取网络输入的 4 个不同尺度的特征层进行重新连接,将深层特征层经过上采样与所有浅层特征层相连,提出了 IDenseNet 网络结构,从而提高了安全帽佩戴检测的性能。

下文对图 2 所示的改进算法的网络结构进行详细介绍。

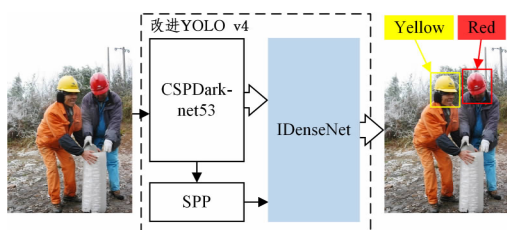


图2 改进 YOLO v4 算法的结构

Fig. 2 Structure of improved YOLO v4 algorithm

### 3.2.1 改进多尺度预测网络——IDenseNet

在实际施工环境中,由于距离较远或者遮挡等情况,安全帽目标可能在原始图片中较小,而造成漏检和误检。首先,为了减少由于下采样而造成的小目标漏检问题,在 YOLO v4 的基础上加入了一组  $128 \times 128$  的特征层输出,将原来的 8 倍下采样改为 4 倍下采样,从而有效增强特征提取网络的小目标特征提取能力和检测性能。其次,为了使浅层特征层具有丰富卷积层的安全帽特征语义信息,基于密集连接思想提出了一种全新的特征融合网络——IDenseNet(见图 3),将深层特征层经过上采样与所有浅层特征层相连,形成类似密集块的特征融合网络。

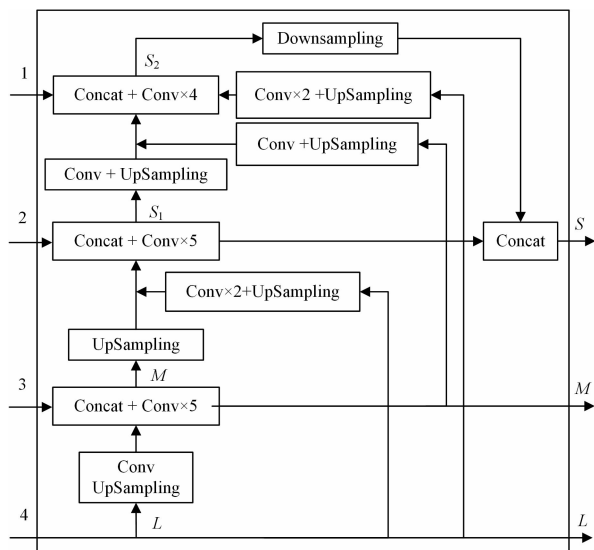


图3 IDenseNet 特征融合模块的结构

Fig. 3 Structure of IDenseNet feature fusion module

从图 3 可以得知, IDenseNet 结构中将 SPP 模块输出的  $16 \times 16$  特征层记为  $L$  特征层,  $L$  特征层直接输入负责大目标检测的 Yolo Head 分类器。  $L$  特征层通过 2 倍上采样与  $32 \times 32$  特征层进行拼接, 拼接后的  $32 \times 32$  特征层记为  $M$  特征层,  $M$  特征层直接输入负责中型目标检测的 Yolo Head 分类器。随后,  $M$  特征层通过 2 倍上采样,  $L$  特征层通过 4 倍上采样与  $64 \times 64$  特征层进行拼接, 3 个特征层拼接后特征层记为  $S_1$  特征层; 同时,  $S_1$  特征层通过 2 倍上采样,  $M$  特征层经过 4 倍上采样,  $L$  特征层经过 8 倍上采样与  $128 \times 128$  特征层进行拼接, 4 个特征层进行拼接后所得特征层记为  $S_2$ ;  $S_2$  通过 2 倍下采样后与  $S_1$  特征层拼接, 拼接后所得特征层记为  $S$  特征层,  $S$  特征层输入负责小目标检测的 Yolo Head 分类器。

根据上述分析, 为了提高安全帽小目标检测性能, 旨在增

强  $S$  特征层的特征表达效果, 本文通过密集连接方式, 使每个特征融合后的特征层的输入都包含深层特征层信息, 从而将较深层特征层的特征进行拼接组合, 形成对目标特征更丰富的描述和判别, 实现特征重用, 进而获得更好的特征表达效果。其中,  $M$  和  $L$  特征层经过上采样后与  $64 \times 64$  特征层拼接得到  $S_1$  特征层,  $S_1$ 、 $M$ 、 $L$  特征层经过上采样与  $128 \times 128$  特征层拼接得到  $S_2$  特征层,  $S_2$  特征层下采样与  $S_1$  特征层拼接这一过程也是一个特征反复提取的过程, 可以达到丰富卷积层的语义信息、提高小目标检测性能的目的。

### 3.2.2 CSPDarknet53 特征提取网络

CSPDarknet53 是检测算法的主干特征提取网络, 由多个 Resblock\_body 模块组合而成, Resblock\_body 模块是由一次下采样和多个残差结构堆叠而成。CSPDarknet53 是在 Darknet53 基础上加入了 CSPNet<sup>[32]</sup> 结构, 在 Resblock\_body 模块基础上进行了改进, 即在残差块 Resblock 进行堆叠的同时增加了一条残差边, 该残差边中的数据只经过少量计算连接到 Resblock\_body 模块。最后, 改进后的网络可以有效增强网络学习能力, 使得在轻量化网络的同时可以保证检测的准确性, 还可以有效降低计算瓶颈和内存成本。Resblock\_body 改动前后的结构如图 4 所示。

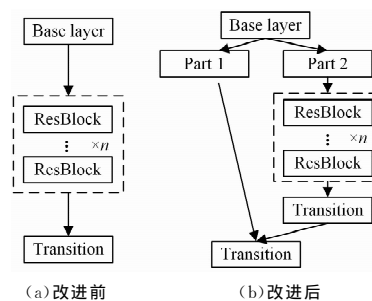


图4 改进前后 Resblock\_body 的结构图

Fig. 4 Structure diagram of Resblock\_body before and after improvement

另外, CSPDarknet53 将 Yolo v3 中 DarknetConv2D 的激活函数由 LeakyReLU 函数修改成了 Mish 函数(见式(3)), 从而将 DarknetConv2D\_BN\_Leaky 卷积模块改进为 DarknetConv2D\_BN\_Mish 卷积模块, 使其具有更好的计算性能。其中, Mish 函数如图 5 所示, 由图可知 Mish 激活函数无上边界, 因此在使用中可以避免由于限幅造成的饱和; 且 Mish 激活函数相对来说梯度更平滑, 能允许更好的信息深入神经网络, 从而得到更优的准确性和泛化能力。

$$\text{Mish} = x \times \tanh(\ln(1 + e^x)) \quad (3)$$

其中,  $x$  表示输入激活函数的输入值;  $\tanh$  表示双曲正切函数。

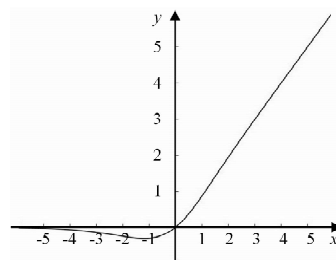


图5 Mish 函数图

Fig. 5 Mish function graph

### 3.2.3 Yolo Head 分类器

本文采用了 3 个 Yolo Head 分类器,分别对大、中、小尺度的特征图进行检测。其中,大尺寸的  $64 \times 64$  特征图是由原始图片经过 8 倍下采样所得,相比其他尺寸其感受野(表示特征层上的像素点在原始图片上映射的区域大小)较小,适合检测小目标。小尺寸的  $16 \times 16$  特征图感受野最大,适合检测大的目标。 $32 \times 32$  特征图感受野介于  $64 \times 64$  和  $16 \times 16$  特征图之间,适用于中型目标检测。因此,本文算法采用三分类器设计,针对原始图像中不同占比的安全帽,可实现覆盖更为全面的目标检测效果,尤其是负责小目标检测的 Yolo Head 分类器,本文改进设计的 IDenseNet 模块可增强语义信息描述,提高其对小目标安全帽的检测精度。

### 3.2.4 损失函数

损失函数是衡量 YOLO 网络输出期望预测结果和期望结果的一致性指标。YOLO v4 安全帽佩戴检测算法的检测分类结果中包含每个预测框的预测类别、置信度和位置,因此在构建损失函数时需对上述构成部分进行评价,分别计算预测结果的类别误差、位置误差和置信度误差,并加权得到整体的损失函数。因此,本文算法的预测框损失函数由 3 部分组成,公式描述为:

$$loss = loss_{location} + loss_{confidence} + loss_{class} \quad (4)$$

其中,  $loss_{location}$  为位置损失,  $loss_{confidence}$  为置信度损失,  $loss_{class}$  为类别损失。

YOLO v4 中  $loss_{location}$  采用了 CIOU Loss 函数进行计算,将目标与锚点之间的距离、重叠率、尺度以及惩罚项都考虑在内,使预测框更加符合真实框,从而使目标框回归更加稳定,解决了只使用 IOU 函数时在训练过程中出现发散、震荡等问题。CIOU 的计算式如下:

$$CIOU = IOU - \frac{\rho^2(b, b^{gt})}{c^2} - \alpha v \quad (5)$$

其中,  $\alpha$  和  $v$  的计算式如下:

$$v = \frac{4}{\pi^2} \left( \arctan \frac{\omega^{gt}}{h^{gt}} - \arctan \frac{\omega}{h} \right)^2 \quad (6)$$

$$\alpha = \frac{v}{(1 - IOU) + v} \quad (7)$$

其中,  $IOU$  表示真值框和预测框的交并比;  $b$  表示预测框中心点;  $b^{gt}$  表示正式框中心点;  $\rho^2(b, b^{gt})$  表示预测框和真实框的中心点的欧氏距离;  $c$  表示能够同时包含预测框和真实框的最小闭包区域的对角线距离;  $\omega^{gt}, h^{gt}, \omega, h$  分别表示真值框和预测框的宽和高。损失函数各部分的具体公式如下:

$$loss_{location} = 1 - CIOU \quad (8)$$

$$loss_{class} = \sum_{i=0}^{S^2} \prod_{i \in \text{obj}} \sum_{c \in \text{class}} (p_i(c) - \hat{p}_i(c))^2 \quad (9)$$

$$loss_{confidence} = \sum_{i=0}^{S^2} \sum_{j=0}^B \prod_{ij \in \text{obj}} (C_i - \hat{C}_i)^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B \prod_{ij \in \text{noobj}} (C_i - \hat{C}_i)^2 \quad (10)$$

其中,  $\prod_{ij}$  用于确定单元格  $i$  中是否含有目标,若有则为 1,否则为 0;  $S$  表示将特征图分割成  $S$  个单元格;  $\prod_{ij}$  用于确定单元格  $i(i=0, \dots, S^2)$  中第  $j(j=0, \dots, B)$  个预测框是否负责该目标的检测;  $p_i(c)$  预测框所属类别为类别  $c$  的分类分数;  $\hat{p}_i(c)$  为预测框所属类别真实值;  $\lambda_{noobj}$  表示单元格中无目标的权重系

数;  $C_i$  为置信度分数;  $\hat{C}_i$  为预测框和真值框的交集部分。

## 4 实验结果与分析

### 4.1 实验环境与实验数据

为验证本文方法的有效性,本文使用 Linux 搭建了实验开发平台,采用 ubuntu16.04 操作系统, GPU 选用 NVIDIA GeForce GTX 1080Ti, CUDA9.0, CUDNN7.0, 实验所用深度学习框架为 Darknet。

本文中使用的安全帽佩戴检测开源数据集 GDUT-HWD<sup>[16]</sup>。GDUT-HWD 中的图片来自网络搜索,其涵盖了不同场景、视觉范围、光照、个人姿势和遮挡情况。GDUT-HWD 共有 3174 张图片,分为 red, yellow, white, blue 和 none 5 个类别,包含实例 18893 个。按一定比例将数据集划分为训练集和验证集。其中,训练集图片有 2540 张,验证集图片有 634 张。对 GDUT-HWD 数据集进行预处理,将数据集注释 xml 文件修改成 YOLO v4 训练所需的 txt 格式文件。其中 txt 格式文件中包含训练或验证所使用的图片的路径及名称。

对处理过的数据集使用 K-means 聚类算法,获取 9 种尺寸的锚,从而得到最适合安全帽检测的锚从小到大分别为 (8, 18), (12, 25), (19, 35), (29, 50), (44, 74), (66, 109), (92, 168), (145, 237), (270, 379)。之后,对模型进行 30000 次迭代训练,每隔 1000 次保存一次权重,每隔 100 次更新一次最优权重。

### 4.2 安全帽佩戴目标检测模型训练与测试

在目标检测算法测试中,常用的性能指标有准确率(Precision)、召回率(Recall)、平均准确率(Average Precision, AP)、平均准确率均值(mean Average Precision, mAP)、P-R 曲线、IOU 和检测速度 FPS 等。

各评估指标的意义如下: TP 为算法中正确分类的阳性样本; FP 为分类错误的阳性样本; FN 为分类错误的阴性样本; N 为图片数量; NC 为目标类型。

准确率: 预测样本中实际正样本数与所有正样本数的比值,即:

$$Precision = \frac{TP}{TP + FP} \quad (11)$$

召回率: 预测样本中实际正样本数与预测的样本数的比值,即:

$$Recall = \frac{TP}{TP + FN} \quad (12)$$

平均准确率(AP): 预测样本的准确率均值,即:

$$AP = \frac{\sum Precision}{N} \quad (13)$$

平均准确率均值(mAP): 是对多个验证集个体求平均 AP 值,将其作为目标检测中衡量检测精度的指标,即:

$$mAP = \frac{\sum AP}{NC} \quad (14)$$

P-R 曲线: 以 Precision 和 Recall 为纵、横轴坐标的二维曲线。通过选取不同阈值时对应的精度和召回率画出总体趋势,精度越高,召回率越低,当召回率达到 1 时,对应概率分数最低的正样本,这时正样本数量除以所有大于等于该阈值的样本数量就是最低的精度值。P-R 曲线围起来的面积就是



AP 值,通常来说一个分类器越好,其 AP 值就越高。

IOU:交并比,即 Intersection-Over-Union,表示产生的候选框与真实框的交叠率或重叠度,即候选框和真实框交集与并集的比值。相关度越高,该值就越高,最理想的情况是完全重合,即比值为 1。

检测速度 FPS:即每秒帧率,表示每秒内可以处理的图片数量。每秒处理的图片数量越大,即代表网络的检测速度越快,其中达到实时监测需求为  $FPS \geq 30 \text{ frame/s}$ ,即每秒处理大于等于 30 张图片。

在上述实验环境下,所提安全帽佩戴检测算法模型训练的平均损失函数曲线如图 6 所示,横坐标表示训练迭代次数,纵坐标表示训练期间的平均损失值。

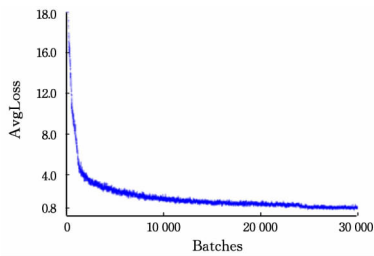


图 6 模型损失随迭代次数变化的曲线

Fig. 6 Model loss curve with the number of iteration

表 1 不同算法检测结果的对比

Table 1 Comparison of detection effects of different algorithms

Algorithm	size	AP/%					mAP/%	FPS
		red	yellow	white	blue	none		
Gaussian YOLO v3	512×512	70.01	86.26	89.05	87.41	85.29	83.61	52.1
YOLO v4	512×512	75.02	90.35	91.75	92.57	91.37	88.21	53.2
Our Method	512×512	84.24	91.89	94.11	93.11	92.51	91.17	52.9

为了更加直观地呈现上述算法之间的检测性能区别,选取部分检测图像结果进行对比说明,如图 7 所示。



(a)Gaussian YOLO v3



(b)YOLO v4



(c)本文算法

图 7 实验结果对比图(电子版为彩色)

Fig. 7 Experimental results comparison chart

图 7(a)为 Gaussian YOLO v3 算法的实验结果图,图 7(b)为 YOLO v4 算法的实验结果图,图 7(c)为本文算法的实

由图 6 可知,随着训练批次的增加,模型的平均损失函数不断下降,最终稳定在 0.8 左右,整个网络趋于收敛。

为了验证所提改进 YOLO v4 算法对安全帽佩戴情况检测精度和效率的有效性和先进性,采用目前目标检测算法研究中性能较优的 Gaussian YOLO v3<sup>[33]</sup> 算法以及 YOLO v4 算法进行对比实验。使用测试集对各算法的实验结果进行测试,以 AP,mAP 和 FPS 为测试评估指标,所得安全帽佩戴检测结果如表 1 所列。由表 1 可以得出,与 Gaussian YOLO v3、YOLO v4 相比,在各种颜色安全帽佩戴检测中,本文算法在 AP 值和 mAP 值上都具有更高的检测精度,相比 Gaussian YOLO v3 和 YOLO v4 算法,mAP 分别提高了 7.56% 和 2.96%,效果显著。其主要原因是本文基于小目标的检测能力的提高,改进后的特征融合网络 IDenseNet 通过密集连接的形式将深层特征层经过上采样和负责小目标检测的浅层特征层进行拼接,从而达到特征重用,使负责小目标检测的浅层特征层同时学习深层特征和浅层特征,从而具有更好的表达效果。在检测速度上,Gaussian YOLO v3、YOLO v4 算法和本文算法在检测速度上都能满足实时监测需求( $FPS \geq 30 \text{ frame/s}$  的需求,即每秒处理大于等于 30 张图片),而本文算法的检测速度可达 52.9 frame/s,优于 Gaussian YOLO v3,与 YOLO v4 算法相比,检测速度几乎没有下降。

验结果图。从实验结果对比图中可以看出,Gaussian YOLO v3、YOLO v4 目标检测算法和本文算法在多目标检测中都表现良好;然而,对比图 7(a)一图 7(c)中的左图可知,Gaussian YOLO v3 和 YOLO v4 算法在图中有多处漏检(如红色框所示位置),而本文算法在相同情况下可以准确检测出目标的位置并正确分类;对比图中的右图可知,Gaussian YOLO v3 和 YOLO v4 算法在复杂环境下有误检情况(见图 7(a)、图 7(b)中的右图),将身穿深色衣物且周围有遮挡的人的身体误检为一个单独的未佩戴安全帽的目标,而在相同情况下本文算法可以正确检测,因此本文算法在目标误检方面也有较大改善;此外,Gaussian YOLO v3 算法在图 7(a)的右图中还存在大量漏检情况。综上所述,本文算法在安全帽佩戴目标检测中的效果优于 Gaussian YOLO v3 和 YOLO v4 算法,可实现高精度的实时检测。另一方面,由图 7 还可以得出,本文算法和上述对比算法都可以准确地检测出安全帽的颜色,从而根据安全帽颜色分辨佩戴者的身份,以有效保障安全帽佩戴的实时检测效果。

结束语 本文提出了一种基于改进 YOLO v4 算法的安全帽佩戴检测方法,可以实时检测安全帽佩戴情况并识别安全帽颜色。为了提高安全帽佩戴小目标检测性能,采用了密集连接的思想对 YOLO v4 算法的特征融合模块进行改进,提出了 IDenseNet 模块,使浅层特征层能融合更丰富的特征信息,在保障实时检测速度的同时,有效提高小目标检测精度。

随后,在 GDUT-HWD 数据集上对所提算法进行了应用实验,并与 Gaussian YOLO v3 和 YOLO v4 算法进行了性能对比,结果显示,本文算法在保证检测速度的同时,能够得到更高的检测精度,且可有效降低安全帽目标误检率,实验结果表明了本文算法有效性和优越性。值得说明的是,由于下采样、严重遮挡、光线、个人姿势等原因,本文算法在漏检情况下的检测精度上还有很大的提升空间,在后续的工作中将在此方面进行进一步的深入研究。

### 参 考 文 献

- [1] ZHANG P, SONG Y F, ZONG L B, et al. Advances in 3D Object Detection; A Brief Survey[J]. Computer Science, 2020, 47(4): 94-102.
- [2] LIN J, DANG W C, PAN L H, et al. Safety Helmet Detection Based on YOLO[J]. Computer Systems & Applications, 2019, 28(9): 174-179.
- [3] FANG M, SUN T T, SHAO Z. Fast Helmet-wearing-condition Detection Based on Improved YOLO v2[J]. Optics and Precision Engineering, 2019, 27(5): 1196-1205.
- [4] LIU W, ANGUELOV D, ERHAN D, et al. SSD: Single Shot MultiBox Detector[C]//European Conference on Computer Vision, Amsterdam, Netherlands, 2016: 21-37.
- [5] WANG H L, QI X L, WU G S. Research Progress of Object Detection Technology Based on Convolutional Neural Network in Deep Learning[J]. Computer Science, 2018, 45(9): 11-19.
- [6] GIRSHICK R, DONAHUE J, DARRELLAND T, et al. Rich Feature Hierarchies for Object Detection and Semantic Segmentation[C]//IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, IEEE, 2014: 580-587.
- [7] GIRSHICK R. Fast R-CNN[C]//International Conference on Computer Vision, IEEE, 2015: 1440-1448.
- [8] REN S, HE K, 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.
- [9] DAI J F, LI Y, HE K M, et al. R-FCN: Object Detection via Region-based Fully Convolutional Networks[C]//Conference and Workshop on Neural Information Processing Systems (NIPS), Barcelona, SPAIN, 2016: 379-387.
- [10] REDMON J, DIVVALA S, GIRSHICK R, et al. You Only Look Once: Unified, Real-time Object Detection[C]//IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, IEEE, 2016: 779-788.
- [11] REDMON J, FARHADI A. YOLO9000: Better, Faster, Stronger[C]//IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, IEEE, 2017: 6517-6525.
- [12] REDMON J, FARHADI A. YOLOv3: An Incremental Improvement[C]//IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 2018.
- [13] BOCHKOVSKIY A, WANG C Y, LIAO H Y. YOLO v4: Optimal Speed and Accuracy of Object Detection[J]. arXiv: 2004. 10934v1, 2020.
- [14] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal Loss for Dense Object Detection[C]//2017 IEEE International Conference on Computer Vision (ICCV), Venice, 2017: 2999-3007.
- [15] DUAN K, BAI S, XIE L, et al. CenterNet: Keypoint Triplets for Object Detection[C]//Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Korea (South), 2019: 6569-6578.
- [16] LAW H, DENG J. CornerNet: Detecting Objects as Paired Keypoints[C]//Proceedings of the European Conference on Computer Vision (ECCV), 2018: 734-750.
- [17] LAW H, TENG Y, RUSSAKOVSKY O, et al. CornerNet-Lite: Efficient Keypoint Based Object Detection[J]. arXiv: 1904. 08900, 2019.
- [18] TIAN Z, SHEN C, CHEN H, et al. FCOS: Fully Convolutional One-Stage Object Detection[C]//2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 2020: 9627-9636.
- [19] YANG L Q, CAI L Q, GU S. Detection on Wearing Behavior of Safety Helmet Based on Machine Learning Method[J]. Journal of Safety Science and Technology, 2019, 15(10): 152-157.
- [20] LONG X T, CUI W P, ZHENG Z. Safety Helmet Wearing Detection Based on Deep Learning[C]//2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, China, 2019: 2495-2499.
- [21] LIU J, XIE Y H. Implementation of an Improved YOLO Algorithm for Intelligent Video Surveillance System[J]. Information Technology and Network Security, 2019, 38(4): 102-106.
- [22] WU J X, CAI N, CHEN W J, et al. Automatic Detection of Hardhats Worn by Construction Personnel: A Deep Learning Approach and Benchmark Dataset[J]. Automation in Construction, 2019, 106: 102894.
- [23] ZHANG M, LI J, DING R L, et al. Remote Sensing Image Object Detection Technology Based on Improved YOLO V2 Algorithm[J]. Computer Science, 2020, 47(6A): 176-180.
- [24] ZHAO Y, CHEN Q, CAO W G, et al. Deep Learning for Risk Detection and Trajectory Tracking at Construction Sites[J]. IEEE Access, 2019, 7: 30905-30912.
- [25] WU F, JIN G Q, GAO M Y, et al. Helmet Detection Based on Improved YOLO V3 Deep Model[C]//2019 IEEE 16th International Conference on Networking, Sensing and Control (ICNSC), Banff, AB, Canada, 2019: 363-368.
- [26] HUANG G, LIU Z, VAN D M L, et al. Densely Connected Convolutional Networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, 2017: 4700-4708.
- [27] XU S K, NI C H, JI C C, et al. Image Caption of Safety Helmets Wearing in Construction Scene Based on YOLO v3[J]. Computer Science, 2020, 47(8): 233-240.
- [28] GU Y W, XU S K, WANG Y R, et al. An Advanced Deep Learning Approach for Safety Helmet Wearing Detection[C]//2019 International Conference on Internet of Things and IEEE Green Computing and Communications and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data, Atlanta, GA, USA, 2019: 669-674.

- [29] FANG Q, LI H, LUO X C, et al. Detecting Non-hardhat-use by a Deep Learning Method from Far-field Surveillance Videos[J]. Automation in Construction, 2018, 85(JAN. ): 1-9.
- [30] ZHENG Z, WANG P, LIU W, et al. Distance-IOU Loss: Faster and Better Learning for Bounding Box Regression[C]// AAAI Conference on Artificial Intelligence. NY, USA, 2020.
- [31] 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. Salt Lake City, UT, 2018: 8759-8768.
- [32] WANG C Y, LIAO H Y, YEH I 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). Seattle, WA, USA, 2020: 1571-1580.
- [33] CHOI J, CHUN D, KIM H, et al. Gaussian YOLO v3: An Accurate and Fast Object Detector Using Localization Uncertainty for Autonomous Driving[C]// 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea (South), 2019: 502-511.



**JIN Yu-fang**, born in 1994, master. Her main research interests include deep learning and target detection.



**WU Xiang**, born in 1990, Ph. D. His main research interests include intelligent learning algorithm and networked motion control.