doi:10.16018/j.cnki.cn32-1650/n.202503010

基于非结构化剪枝和特征蒸馏的轻量化跌倒检测模型

华浩军[1],NAJI Alhusaini[2]
(1.安徽理工大学 计算机科学与工程学院,安徽 淮南 232000;
2.滁州学院 人工智能学院,安徽 滁州 239000)

摘要:针对现有跌倒检测算法计算成本高、实时性差、难以在边缘设备上部署的问题,提出了一种基于YOLOv8n的轻量化跌倒检测模型WPD-YOLO。首先,在YOLOv8n中引入了Wise-IoU作为新的回归损失函数,以降低模型中低质量样本的负面影响,提升模型的收敛速度;其次,采用LAMP剪枝方法对改进后的模型进行压缩,从而有效降低模型的参数量和计算量;最后,通过通道知识蒸馏方法微调模型,在不额外增加参数量的情况下提升了模型的检测精度。实验结果表明,在公开的Fall-Detection数据集上,与YOLOv8n相比,WPD-YOLO的检测参数量减少了74.42%,计算量减少了50%,处理速度提升了56帧/s,同时平均准确率提升了0.4%,达到87.8%。

关键词:跌倒检测;Wise-IoU;模型剪枝;知识蒸馏
中图分类号:TP391 **文献标志码**:A **文章编号**:1671-5322(2025)03-0060-10

随着年龄的增长,人们的反应能力、判断能力和身体协调能力逐渐下降,跌倒成为老年人常见的身体伤害[1]。世界卫生组织数据显示,全球每年因跌倒造成严重伤害且需医疗干预的病例约达3 730万例[2]。因此,对跌倒进行检测并及时采取相应措施对维护老年人安全及缓解医疗系统压力具有重要意义。

现有的跌倒检测方法分为基于可穿戴设备的跌倒检测方法、基于环境传感器的跌倒检测方法和基于计算机视觉的跌倒检测方法。基于可穿戴设备的跌倒检测方法如加速度计[3]和陀螺仪[4],可以提供实时监测,但需要老年人持续佩戴;基于环境传感器的检测方法通过把传感器安装在室内固定位置,利用信号处理和模式识别来检测跌倒[5],因环境传感器在复杂场景中难以部署,且容易产生信号处理延迟,导致实时性较差。

随着人工智能技术的快速发展,基于视觉的跌倒检测逐渐成为该领域的重点研究方向[6],这是因为深度学习方法能自动学习数据特征表示,并在目标检测等任务中展现出卓越的性能。

目前已有许多基于深度学习的跌倒检测方法出现。Bansal等[7]提出一种基于视觉的跌倒检测方法,利用从COCO数据集学习到的丰富特征对原始R-CNN模型进行微调,提高跌倒检测的性能;朱正林等[8]提出基于改进YOLOv5s的跌倒检测模型,通过引入SKAttention模块提升了模型的特征提取能力。尽管目前基于视觉的跌倒检测方法已经具有较高的检测精度,但其对计算资源和存储空间的消耗较大,不利于在资源受限的边缘设备上高效部署。因此如何在保证模型检测能力的同时,尽可能减少模型对算力和存储资源的需求是一个值得探究的现实课题。

常见的模型轻量化方法包括模型压缩、模型量化、知识蒸馏、模型剪枝等。模型剪枝[9]分为结构化剪枝和非结构化剪枝,结构化剪枝通过移除整个神经元、通道或层来保持规整的网络结构;非结构化剪枝则细粒度地剪除单个权重,实现更高稀疏性。知识蒸馏是一种将知识从大型、高性能的教师模型迁移到较小的学生模型的技术,通过知识蒸馏,可以将强大的教师模型中的知识迁

收稿日期:2025-08-18
基金项目:2022年度安徽省高等学校科研计划项目(自然科学类重点项目)(2022AH051102)。
作者简介:华浩军(2001—),男,安徽安庆人,硕士生,主要研究方向为深度学习、目标检测等。
通信作者:Naji Alhusaini(1987—),男,也门人,副教授,博士,主要研究方向为深度学习、智慧养老技术。

移到较小的学生模型中，使学生模型在保持较低计算成本的同时达到与教师模型相似的检测精度。知识蒸馏主要可分为逻辑蒸馏[10]和特征蒸馏[11]两大类，逻辑蒸馏仅传递分类概率，且会丢失关键的运动细节；特征蒸馏则直接对齐师生模型的中间层特征，能保留更完整的时空信息，提升模型对跌倒动作微小差异的敏感性。

由于跌倒检测依赖人体姿态、运动轨迹等时空特征，所以本文基于视觉的跌倒检测，利用计算机视觉和深度学习算法，提出一种基于YOLOv8的轻量级跌倒检测模型WPD-YOLO[12]，从视频、图像和行为模式中分析人体的动态变化，实现更准确的跌倒识别。该算法选择知识蒸馏中的特征蒸馏作为模型轻量化的方法，让模型在保持较高精度的同时有效降低了计算量和参数量，使其更适合部署在边缘设备上。

本文主要创新点有：1）为了平衡边界框回归的强度和模型训练过程中对低质量数据的惩罚，用Wise-IoU损失函数替换了原始YOLOv8中的CIoU损失[13]，使算法更聚焦于普通质量的样本，提升了模型的收敛速度，降低了低质量样本带来的负面危害；2）为了解决在边缘设备上部署过大检测模型的挑战，运用LAMP（learning-based approximation of multi-dimensional pruning）剪枝方法对改进的模型进行压缩[14]，该方法基于不同网络层的特征表现，进行自适应剪枝操作，可以最大限度地保留模型的关键结构，在有效降低模型复杂度的同时，保持模型较高的检测性能；3）为了补偿模型轻量化带来检测性能的下降，采用CWD（channel-wise knowledge distillation）蒸馏方法对轻量化模型进行微调[15]，使之在不增加参数量的情况下，提高模型的检测能力。

1　WPD-YOLO model

1.1　WPD-YOLO model构建

YOLOv8是由Ultralytics开发的目标检测算法，它继承了YOLO系列实时性高、检测能力强的特点，网络架构如图1所示。由图1可知，YOLOv8包括骨干网络、颈部网络和检测头3个组成部分。

本文选择YOLOv8n作为基线模型，以YOLOv8n模型为基础，构建的WPD-YOLO模型如图2所示。由图2可知，该模型首先在YOLOv8n中引入Wise-IoU损失函数W_{IoU}，以提升模型的训练

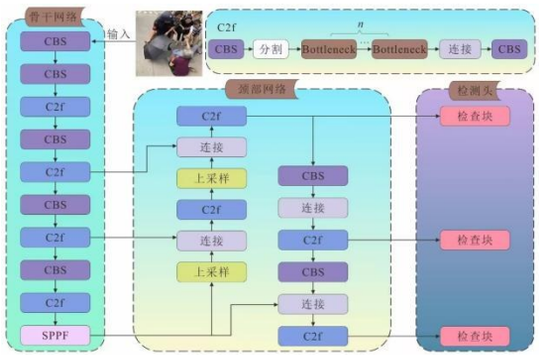


图1　YOLOv8网络结构示意图

Fig. 1　Schematic diagram of YOLOv8 network structure

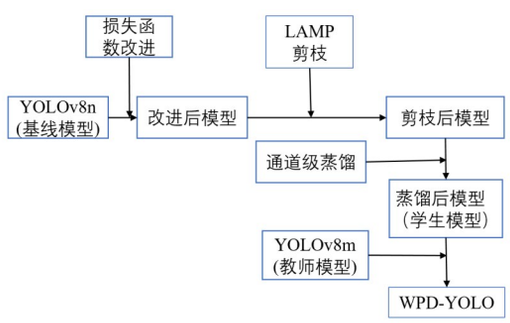


图2　WPD-YOLO模型构建过程

Fig. 2　Construction process of WPD-YOLO model

效果和泛化能力；然后利用模型剪枝技术对模型进行剪枝，去除冗余参数，降低模型的计算量；经过通道修剪的轻量化模型会出现精度下降的情况，需要通过知识蒸馏的方法，即以YOLOv8m模型为教师模型，剪枝后的模型为学生模型，通过知识蒸馏的方法将教师模型中的知识迁移到学生模型中，以进一步提高学生模型的性能和效率，补偿通道移除造成的精度损失。

1.2　损失函数改进

边界框回归损失函数作为目标检测损失函数的重要组成部分，在YOLOv8中利用CIoU损失函数C_{IoU}来预测YOLOv8网络的边界坐标损失，计算公式如下：

$$C_{IoU}=I_{ou}+\frac{\left(x-x_{gt}\right)^2+\left(y-y_{gt}\right)^2}{\left(W_g^2+H_g^2\right)}+pv \quad (1)$$

其中，$p=\dfrac{v}{I_{ou}+v}$

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

$$I_{ou}=\frac{W_iH_i}{wh+w_{gt}h_{gt}-W_iH_i}$$

式中:(x,y)代表预测框某点的坐标;(x_{gt},y_{gt})代表真实框某点的坐标;w、h分别表示预测框的长和宽;w_{gt}、h_{gt}分别表示真实框的长和宽;W_g表示预测框和真实框的长度并集,H_g表示预测框和真实框的宽度并集;W_i表示预测框和真实框的长度交集,H_i表示预测框和真实框的宽度交集;p是平衡参数,能够动态调整长宽比的权重;v是长宽比度量函数,用于衡量预测框与真实框的长宽比;I_{ou}表示预测框与真实框的交并比。

图3展示了预测框和真实框的空间关系。

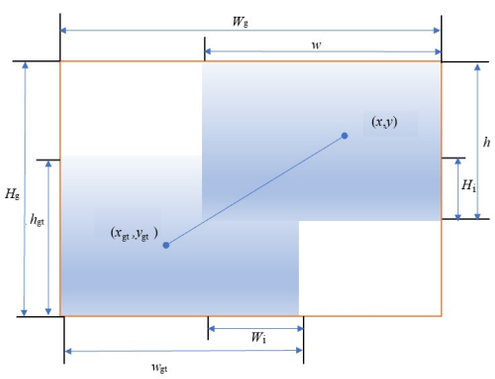


图3　预测框与真实框的关系

Fig. 3　The relationship between predicted boxes and ground truth boxes

C_{IoU}损失函数在计算边界框回归损失时,不仅考虑了预测框与真实框的交并比I_{ou},还引入了预测框和真实框长宽比的惩罚项v。当预测框与真实框大小相差很大且长宽比存在线性关系时,C_{IoU}惩罚项v将恶化为0,使得低质量的示例产生有害的梯度。为此,在WPD-YOLO模型中使用W_{IoU}来替换原始YOLOv8的损失函数C_{IoU}。

W_{IoU}定义[13]如下:

$$W_{IoU}=rR_{WIoU}I_{ou},\ R_{WIoU}\in[1,e) \tag{2}$$

其中,$r=\dfrac{\beta}{\delta\alpha^{\beta-\delta}}$

$$R_{WIoU}=\exp\frac{\left(x-x_{gt}\right)^2+\left(y-y_{gt}\right)^2}{W_g^2+H_g^2}$$

式中:r表示动态梯度增益;R_{WIoU}是基于中心点距离构建的动态权重因子;β是离群度,对于高质量和低质量的示例β都分配较小的梯度增益,以便将边界框回归集中在普通质量的锚点上;α、δ表示超参数。

比起C_{IoU}损失函数,W_{IoU}不再使用长宽比作为惩罚项;而是通过引入r和R_{WIoU}使损失函数聚焦于普通质量锚框。W_{IoU}通过动态梯度分配机制,提升了跌倒检测过程的鲁棒性,抑制了低质量样本产生的噪声梯度,加快了模型收敛。

1.3　层自适应幅度剪枝

为了使模型具有更好的实时性和设备适应性,需要采用网络剪枝的方法消除神经网络中的冗余参数。本文使用LAMP剪枝的方法来压缩YOLOv8模型,使其在尽可能保留性能的同时确保最小的计算和存储开销。

LAMP算法是一种修剪深度学习模型的优化方法。与在单一维度上操作的传统剪枝技术不同,LAMP考虑了整个网络结构的多个维度,包括权重通道以及跨越不同层级的神经元。LAMP通过计算目标连接的分数来衡量连接的重要性,LAMP分数的计算公式[14]如下:

$$s_{core}(u;F)=\frac{(F[u])^2}{\sum\limits_{v\geqslant u}(F[v])^2} \tag{3}$$

式中:$s_{core}(u;F)$表示在给定整个网络权重F的条件下,特定连接u的LAMP分数;$F[u]$表示目标连接的权重;$F[v]$表示同一层中所有其他连接的权重。

在神经网络中,每个突触连接都有一个相关的权重。LAMP分数衡量了同一层内所有现有连接中目标连接的相对重要性,并修剪同层内权重较小的连接。因此,具有相同权重的两个连接可能具有不同的LAMP分数。

LAMP剪枝流程如图4所示,首先建立依赖图,将相互依赖的权重分组绑定,确保剪枝时能够进行整组处理,同时将颜色相同的部分分为一个组;其次按照基于式(3)计算的通道重要性评分进行剪枝,即利用公式(3)计算每个组的分数并排序,然后按照预设比例剔除低分数的分组;最后对剪枝后的模型进行微调,并通过参数优化补偿精度损失,以平衡效率与性能。

1.4　通道知识蒸馏

知识蒸馏是一种提升轻量级检测算法准确

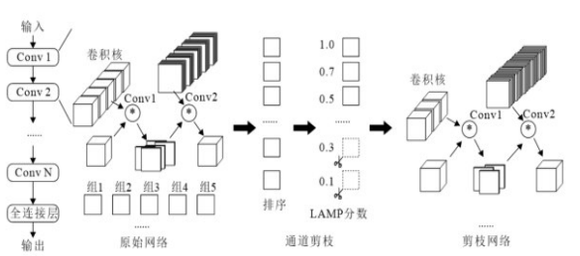


图4　LAMP剪枝过程

Fig. 4　LAMP pruning process

率的有效方法。现有的蒸馏方法通常侧重于点的特征对齐或空间位置之间结构化信息的对齐,忽略了嵌入在通道维度中的丰富知识。相比之下,通道级知识蒸馏 CWD 直接在通道级别对齐特征分布,使更全面的知识从教师模型迁移到学生模型。该方法在迁移多尺度特征和细粒度细节方面表现突出,显著提高了学生模型的性能。

通道级知识蒸馏的具体蒸馏过程如图 5 所示,其核心思想是对齐教师模型和学生模型之间的特征通道分布,从而将教师的知识传递给学生,提高其性能。为了实现这一点,通道激活首先被转换为概率分布,从而能够使用概率距离度来量化差异。

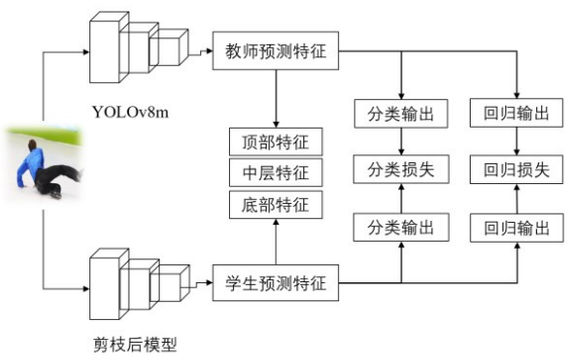


图 5　特征蒸馏过程

Fig. 5　Feature distillation process

设 T 和 S 分别表示教师网络和学生网络,其激活图分别表示为 y^S 和 y^T,则通道的蒸馏损失 $\varphi\left(\phi\left(y^T\right),\phi\left(y^S\right)\right)$ 计算公式[15]如下所示:

$$\varphi\left(\phi\left(y^T\right),\phi\left(y^S\right)\right)=\varphi\left(\phi\left(y_c^T\right),\phi\left(y_c^S\right)\right) \tag{4}$$

其中,$\phi\left(y_c\right)=\dfrac{\exp\dfrac{y_{c,j}}{\tau}}{\sum\limits_{i=1}^{W\times H}\exp\dfrac{y_{c,j}}{\tau}}$

式中:$y_{c,j}$ 表示第 c 个通道在空间位置 j 的激活值;τ 是一个超参数,表示控制概率分布的平滑程度。

通过对通道蒸馏损失进行归一化处理,最终得到用于计算损失的概率分布 $\phi(y_c)$,从而有效消除较大且紧凑网络之间量级差异的影响。

在图 2 的 WPD-YOLO 模型中,选择特征提取能力更强的 YOLOv8m 作为教师模型,将剪枝后的模型作为学生模型。教师模型通过多层次特征蒸馏将知识传递给学生模型,学生模型则通过分类损失学习教师模型输出的概率分布,避免硬标签带来的信息损失,再通过回归损失学习教师模型的定位输出分布,捕捉边界框预测的位置,确保学生模型在目标定位上能够模仿教师模型的输出特性。

2　实验与分析

2.1　实验准备

为验证各项改进措施,以及改进后模型在非理想环境下的检测效果,需对各项改进措施及模型优化前后的效果进行对比实验。为此,设置实验环境如下:所有实验均在 Windows10 环境下使用 AutoDL 进行;具体的系统环境配置为 Python3.8,CUDA 版本为 11.2,GPU 模型为 NVIDIA GeForce RTX4090D (24 GB 显存);训练参数为初始学习率 0.01、训练轮数 200 次、批量大小为 16。

为完成上述实验,需要寻找具有代表性的各种不同类型的图片集进行验证。为此,在 Roboflow Universe 平台上选择公开数据集 Fall-Detection,从中选择包含由不同相机视角、不同年龄和职业、不同跌倒方向和姿势组成的 10 793 张图片。

为方便实验,将这 10 793 张图片按 7∶2∶1 分成训练集 7 927 张、验证集 1 967 张、测试集 899 张。另外为衡量检测效果,针对跌倒检测任务,采用平均精度均值 m_{AP}、参数数量 P_{arams}、计算复杂度 F_{LOPs} 和每秒帧数 F_{PS} 共 4 个评价指标进行评估。其中,m_{AP} 指标为 mAP@0.5,表示 I_{ou} 阈值为 0.5 时的平均检测精度。

2.2　消融实验

为验证各项改进措施的作用,在上述平台上对测试集数据开展消融实验,结果如表 1 所示。由表 1 可以看出,经过损失函数的改进,模型的精度提升了 1.1 个百分点,F_{PS} 提升了 17 帧/s;接着对模型进行剪枝,虽然剪枝后模型精度下降了 1.7 个百分点,但模型参数量减少了 74.42%、计算量减少了 50%、F_{PS} 提升了 34 帧/s,使模型实现了轻量化;最后再通过知识蒸馏弥补损失的精度,使模型在参数量、计算量不变的情况下,检测精度回调至 87.8%,F_{PS} 又提升了 5 帧/s。显然,通过损失函数改进、剪枝和知识蒸馏,模型实现了精度和计算量的平衡。

2.3　损失函数对比实验

为了验证损失函数对模型检测效果的影响,在基线模型 YOLOv8n 上比较了几种常见的损失函数,结果如表 2 所示。由表 2 可以看出,比起其

表1　模型消融实验
Table 1　Model ablation experiment

模型	改进	m_{AP}/%	F_{PS}/(帧·s^{-1})	P_{arams}/MB	F_{LOPs}/GB
YOLOv8n	C_{IoU}	87.4	214	3.01	8.2
WPD-YOLO	W_{IoU}	88.5	231	3.01	8.2
	W_{IoU}+剪枝	86.8	265	0.77	4.1
	W_{IoU}+剪枝+蒸馏	87.8	270	0.77	4.1

表2　损失函数对比
Table 2　Loss function comparison

损失函数	m_{AP}/%	F_{PS}/(帧·s^{-1})	P_{arams}/MB	F_{LOPs}/GB
C_{IoU}	87.4	214	3.01	8.2
G_{IoU}	85.5	180	3.01	8.2
E_{IoU}	85.9	205	3.01	8.2
D_{IoU}	86.4	210	3.01	8.2
W_{IoU}	88.5	231	3.01	8.2

他损失函数，W_{IoU}函数展现了更好的检测能力和更快的检测速度。这是由于使用W_{IoU}损失函数能使算法更聚焦于普通质量的样本，降低了低质量样本带来的负面危害。

2.4　剪枝实验

针对所使用的通道剪枝方法，对改进损失函数后的模型进行了不同剪枝方法和不同剪枝参数的对比实验，结果如表3所示。

表3最左边一列表示不同的剪枝方法和对应的计算量压缩比。由表3可以看出，在2.0的计算量压缩比条件下，LAMP算法比Group-SL算法在准确率上提高了1.8个百分点，比Group-Slim算法提高了1.2个百分点，比Group-Norm算法提高了1.7个百分点；在不同的计算量压缩比下，LAMP算法的4.0计算量压缩比虽然可以进一步减少模型参数量至0.28 MB，但检测精度比2.0倍剪枝率降低了3.8个百分点。因此，在综合考虑模型精度、计算量和速度下，选择2.0倍剪枝率的LAMP算法作为本文的剪枝算法。

表3　剪枝实验结果
Table 3　Pruning experiment results

枝方法	m_{AP}/%	F_{PS}/(帧·s^{-1})	P_{aramss}/MB	F_{LOPs}/GB
LAMP1.5x	86.4	230	2.29	5.4
LAMP2.0x	86.8	265	0.77	4.1
LAMP4.0x	83.0	380	0.28	2.0
Group-SL2.0x	85.0	232	1.6	4.1
Group-Slim 2.0x	85.6	301	1.5	4.1
Group-Norm 2.0x[16]	85.1	261	1.6	4.1

2.5　蒸馏实验

针对所使用的蒸馏方法，对剪枝后的模型进行了不同知识蒸馏方法的对比实验，结果如表4所示。由表4可以看出，在检测精度方面，CWD的m_{AP}达到了87.8%，较MGD方法提升1.5个百分点，比Mimic方法高出3.8个百分点；相比纯逻辑蒸馏方法(BCKD)和混合蒸馏方法(CWD+BCKD)，CWD分别领先1.0和0.8个百分点；在推理速度方面，CWD的F_{PS}达到270帧/s，为所有蒸馏方法中最快的。

表4　蒸馏实验结果
Table 4　Distillation experiment results

知识蒸馏方法	蒸馏类型	m_{AP}/%	F_{PS}/(帧·s^{-1})
CWD+BCKD	hybrid	87.0	237
BCKD	logical	86.8	268
MGD[17]	feature	86.3	262
Mimic	feature	84.0	240
CWD	feature	87.8	270

2.6　检测效果分析

针对不同的跌倒场景，对改进前后的模型运用测试集进行性能检测，结果如图6所示。

图6包括3组图像，分别是原始图像，经过YOLOv8n、WPD-YOLO处理过的图像，每组图像又包含前身跌倒、正向跌倒和存在遮挡3幅图像，每幅图像又包含检测目标的位置和置信度等信息。由图6可知，在前向跌倒和后向跌倒2种不同的跌倒姿态下，经WPD-YOLO模型处理过的图像，置信度较YOLOv8n分别提升了0.1和0.5，有效避免了误检情况；从图6的第3列图片可以看到，在跌倒者在被其他人遮挡的情况下，YOLOv8n的检测置信度为0.8，WPD-YOLO模型的置信度达到了0.9，比YOLOv8n提升了0.1。



a 原始图像

b YOLOv8n

c WPD-YOLO

图 6　检测效果对比

Fig. 6　Detection performance comparison

2.7　不同算法对比实验

为了全面评估改进模型的检测能力，将WPD-YOLO模型与当前目标检测领域的主流模型进行了系统的对比实验，结果如表5所示。由表5可知，WPD-YOLO模型在保持87.8%精度的同时，模型参数量仅为0.77 MB、计算量为4.1 GB，这两项指标均为对比模型中的最优值，虽然RT-DETR取得了87.6%的较高精度，但参数量和计算量均远高于WPD-YOLO模型；另外，从计算参数量来看，WPD-YOLO模型的0.77 MB较YOLOv8n的3.01 MB减少了74.42%，比YOLOv3-tiny的12.17 MB减少了93.67%，更是远低于RT-DETR的32.9 MB和YOLOv3的103.75 MB；在计算效率方面，WPD-YOLO模型4.1 GB的F_{LOPs}比YOLOv8n的8.2 GB降低了50%，较YOLOv3的283.3 GB减少了98.55%；在推理速度方面，WPD-YOLO模型达到了270 帧/s，优于YOLOv8n的214 帧/s和YOLOv3的182 帧/s，仅有YOLOv3-tiny的686 帧/s比WPD-YOLO模型更快，但其83.2%的精度比WPD-YOLO模型低了4.6个百分点。

表 5　模型对比实验

Table 5　Model comparison experiment

模型	m_{AP}/%	F_{PS}/(帧·s^{-1})	P_{arams}/MB	F_{LOPs}/GB
YOLOv8n	87.4	214	3.01	8.2
YOLOv3-tiny	83.2	686	12.17	19.1
YOLOv3	86.7	182	103.75	283.3
YOLOv5n	79.2	226	2.65	7.8
YOLOv6	86.0	152	4.50	13.1
RT-DETR	87.6	60	32.90	98.0
WPD-YOLO	87.8	270	0.77	4.1

3　结论

针对目前跌倒检测算法实时性差、难以部署在边缘设备的问题，本文提出了一种轻量级跌倒检测模型WPD-YOLO。该算法首先引入W_{IoU}损失函数使得算法更关注普通质量的样本，提高模型收敛速度；接着结合网络剪枝与知识蒸馏技术，在保证模型精度的情况下有效降低了模型的参数量与计算量。实验结果表明，WPD-YOLO模型在Fall-Detection数据集上表现出色，mAP@0.5指标达到87.8%，推理速度为270 帧/s，参数量为0.77 MB，计算量F_{LOPs}为4.1 GB。显然，相比YOLOv8n模型，WPD-YOLO模型不仅能够有效解决跌倒检测系统在资源受限设备上部署困难的问题，展现了较高的实用价值，还体现了较高的运行效率。

参考文献：

[1] COLÓN-EMERIC C S, MCDERMOTT C L, LEE D S, et al. Risk assessment and prevention of falls in older community-dwelling adults: a review[J]. JAMA, 2024, 331(16): 1397-1406.

[2] RASSEKH E, SNIDARO L. Survey on data fusion approaches for fall-detection[J]. Information Fusion, 2025, 114: 102696.

[3] 尚宪茹, 刘子建, 胡志刚, 等. 国内外老年智能健康穿戴产品研究态势分析[J]. 丝绸, 2025, 62(2): 82-93.

[4] NEWAZ N T, HANADA E. The methods of fall detection: a literature review[J]. Sensors, 2023, 23(11): 5212.

[5] 杨松铭, 王玫. 基于梅尔谱特征和改进ResNet网络的室内跌倒检测方法[J]. 桂林理工大学学报, 2025, 45(2): 251-259.

[6] ALANAZI T, BABUTAIN K, MUHAMMAD G. A robust and automated vision-based human fall detection system using 3D multi-stream CNNs with an image fusion technique[J]. Applied Sciences, 2023, 13(12): 6916.
[7] BANSAL A, SHARMA R, KATHURIA M. A vision-based approach to enhance fall detection with fine-tuned faster R-CNN[C]//2023 International Conference on Advanced Computing & Communication Technologies (ICACCTech), 2023.
[8] 朱正林, 钱予阳, 马辰宇, 等. 基于改进YOLOv5s的跌倒行为检测[J]. 计算机测量与控制,2024, 32(10): 26-31, 38.
[9] 段宇晨, 方振宇, 郑江滨. 神经网络轻量化综述[J]. 计算机科学与探索, 2025, 19(4): 835-853.
[10] ZHAO B R, CUI Q, SONG R J, et al. Decoupled knowledge distillation[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). June 18-24, 2022, New Orleans, LA, USA. IEEE, 2022: 11943-11952
[11] HUANG G M, SHEN A, HU Y X, et al. Optimizing yolov5s object detection through knowledge distillation algorithm [C]//2024 5th International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE),25-27 October 2024,Shenzhen, China.
[12] SOHAN M, SAI RAM T, RAMI REDDY C V. A review on YOLOv8 and its advancements[C]//International Conference on Data Intelligence and Cognitive Informatics. Singapore: Springer Nature Singapore, 2024: 529-545.
[13] TONG Z J, CHEN Y H, XU Z W, et al. Wise-IoU: bounding box regression loss with dynamic focusing mechanism[J]. arXiv:2301. 10051, 2023. https://doi. org/10. 48550/arXiv. 2301. 10051
[14] LEE J, PARK S, MO S, et al. Layer-adaptive sparsity for the magnitude-based pruning[C]//International Conference on Learning Representations. 2021.
[15] SHU C Y, LIU Y F, GAO J F, et al. Channel-wise knowledge distillation for dense prediction[C]//2021 IEEE/CVF International Conference on Computer Vision (ICCV). October 10-17, 2021. Montreal, QC, Canada. IEEE, 2021: 5311-5320.
[16] FANG G F, MA X Y, SONG M L, et al. DepGraph: towards any structural pruning[C]//2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). June 17-24, 2023, Vancouver, BC, Canada. IEEE, 2023: 16091-16101.
[17] YUE K Y, DENG J F, ZHOU F. Matching guided distillation[M]//Computer Vision :ECCV 2020. Cham: Springer International Publishing, 2020: 312-328.

Lightweight Fall Detection Model Based on Unstructured Pruning and Feature Distillation

HUA Haojun[1], NAJI Alhusaini[2]

(1. School of Computer Science and Engineering, Anhui University of Science and Technology, Huainan Anhui 232000, China;
2. School of Artificial Intelligence, Chuzhou University, Chuzhou Anhui 239000, China)

Abstract:To address the issues of high computational cost, poor real-time performance, and difficulty in deployment on edge devices in existing fall detection algorithms, a lightweight fall detection model named WPD-YOLO based on YOLOv8n is proposed. First, Wise-IoU is introduced as a new regression loss function in YOLOv8n to reduce the negative impact of low-quality samples in the model and improve its convergence speed. Next, the LAMP pruning method is employed to compress the improved model, effectively reducing both the parameter count and computational load. Finally, the model is fine-tuned by channel knowledge distillation method, and the detection accuracy of the model is improved without additional parameters. Experimental results show that on the public Fall-Detection dataset, compared with YOLOv8n, the parameter count of WPD-YOLO is reduced by 74.42%, the computational complexity is reduced by 50%, the speed of processing is increased by 56 frames per second, while the mean average precision is increased by 0.4%, reaching 87.8%.

Keywords:fall detection; Wise-IoU; model pruning; knowledge distillation

(责任编辑:李华云)