

基于改进 Alphapose 的红外图像人体摔倒检测算法

张 鹏, 沈玉真, 李培华, 张恺翔
(中航华东光电有限公司, 安徽 芜湖 241002)

摘要: 红外图像中的人体摔倒检测不受环境光照射的影响, 在智能安防领域有着重要的研究意义和应用价值。现有的摔倒检测方法没有充分考虑人体关键点的位置变化规律, 容易对类摔倒动作造成误检。针对这一问题, 本文提出一种基于改进 Alphapose 的红外图像摔倒检测算法。该算法使用 Yolo v5s 目标检测网络, 在提取人体目标框输入姿态估计网络的同时, 对人体姿态进行直接分类, 再结合人体骨架关键点的位置信息和姿态特征进行判断。通过实验证明, 该算法在准确度和实时性方面都有良好的表现。

关键词: 红外图像; 摔倒检测; 关键点; 目标检测

中图分类号: TP391

文献标识码: A

文章编号: 1001-8891(2023)12-1314-08

Infrared Image Human Fall Detection Algorithm Based on Improved Alphapose

ZHANG Peng, SHEN Yuzhen, LI Peihua, ZHANG Kaixiang
(Aviation Industry Crop Huadong Photoelectric Company Limited, Wuhu 241002, China)

Abstract: Human fall detection in infrared images is not affected by ambient light and has important research and application value in intelligent security. Existing fall detection methods do not fully consider the position change law of key points on the human body, which can easily cause false detections of similar fall movements. To solve this problem, we propose an infrared image fall detection algorithm based on an improved alpha pose. The algorithm uses the YOLO v5s object detection network to directly classify human poses while extracting the human body target frame and inputting the pose estimation network. It then evaluates it in combination with the position information and posture characteristics of the key points of the human skeleton. Experiments showed that the algorithm exhibited good performance in terms of accuracy and real-time performance.

Key words: infrared images, fall detection, key points, object detection

0 引言

人体摔倒检测可以有效地检测视频中的摔倒行为, 降低人在摔倒后无法及时呼救的风险^[1-3]。现有的基于计算机视觉的人体摔倒检测方法大多使用可见光图像, 这类方法受光照影响在昏暗环境下效果不佳。红外图像能够保护个人隐私, 而且不受光照和恶劣天气的影响, 适用于全天候的人体摔倒检测, 本文将对红外图像摔倒检测算法进行研究。现有的人体摔倒检测方法大致可分为以下 3 类: 1) 基于 Freeman 链码的检测方法, 文献[4]利用 Freeman 链码记录轮廓信息, 计算实际 Freeman 链码与摔倒模板链码的欧式空间距离结合时间判定条件判断是否发生摔倒行

为, 但在大规模遮挡时容易出现误差; 2) 基于关键点的摔倒检测方法, 文献[5]利用建立的红外图像摔倒数据集进行 CenterNet 网络训练, 识别红外图像中的人体姿态, 但没有突出摔倒动作的时序性, 容易对缓慢坐地等类摔倒动作造成误检; 3) 基于宽高比和质心变化率的检测方法, 文献[6]利用人体宽高比这一特征来判断是否有人摔倒, 并使用质心变化率和有效面积比对容易造成误判的情况进行修正, 从几何特征的角度判断人体是否摔倒, 但对特定角度的摔倒检测效果不佳。

针对上述算法的不足, 本文在现有的摔倒检测方法的基础上提出了一种基于改进 Alphapose 的红外图像摔倒检测算法, 该算法使用改进后的 Alphapose

收稿日期: 2021-09-23; 修订日期: 2021-10-13.

作者简介: 张鹏 (1994-), 硕士研究生, 主要研究领域为模式识别、图像处理。E-mail: zhangpeng2735@163.com。

检测红外图像中的骨架关键点和人体姿态,利用得到的人体骨架关键点和姿态类别进行摔倒判定,兼顾了摔倒动作的时序性和前后动作的关联性。

1 改进的 Alphapose 算法

1.1 YOLO v5s

人体检测的精度直接影响后续关键点检测的效果,是整个算法精度的保证。YOLO 系列算法^[7-9]将候选框的搜索和目标的识别定位合二为一,相对于 RCNN^[10-12]系列算法先搜索再识别的两阶段处理模式,YOLO 算法的处理速度更快,适用于端到端的训练与预测,目前已发展至 YOLO v5 版本。YOLO v5 的目标检测网络有 4 种版本,主要区别在于网络深度不同,如表 1 所示。

表1 YOLO v5 各版本参数对比

Table 1 Comparison of parameters between different versions of YOLO v5

Model	Size /pixels	Speed _{v100} /ms	Params(M)
YOLO v5s	640	2.0	7.3
YOLO v5m	640	2.7	21.4
YOLO v5l	640	3.8	47.0
YOLO v5x	640	6.1	87.7

从表 1 可以看到, YOLO v5s 的模型更小、速度更快, 方便部署在嵌入式设备上, 本文算法使用 YOLO v5s 的网络结构。

YOLO v5s 网络首先使用 Focus 模块对输入图像进行切片操作,如图 1 所示。如果网络输入的图片大小为 $640 \times 640 \times 3$, 则将图像隔行隔列取值,生成 $320 \times 320 \times 12$ 的子图,再通过 32 个卷积核卷积生成 $320 \times 320 \times 32$ 的特征图。相对于普通下采样,Focus 模块保留了原始图片的完整信息。

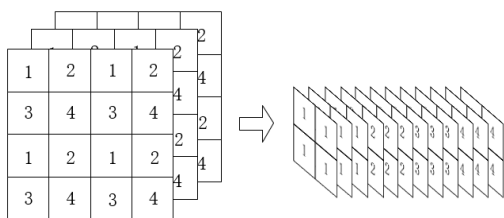


图 1 Focus 模块

Fig.1 Focus module

YOLO v5s 的 Backbone 部分使用了 CSP^[13]模块, 将输入的特征映射划分为两部分, 其中一部分经过多个残差组件和卷积, 然后与另一部分拼接, 在减少计算量的同时可以保证准确率。

Neck 部分采用了 PANet^[14]结构, 主要用于生成

特征金字塔。特征金字塔会增强模型对于不同缩放尺度对象的检测,从而能够识别不同大小和尺度的同一个物体。

1.2 Alphapose 算法

Alphapose^[15]算法是一种人体姿态估计算法，能够检测出图像中的人体骨架，其提出一种区域多人姿态估计框架（regional multi-person pose estimation, RMPE），主要由对称空间变换网络（symmetric spatial transformer network, SSTN）、参数化姿态非极大抑制（parametric pose no-maximum-suppression, PP-NMS）、姿态引导建议发生器（pose-guided proposals generator, PGPG）和并行单人姿态估计器（parallel single person pose estimation, Parallel SPPE）组成，其中 PGPG 用于生成大量训练样本，Parallel SPPE 则充当额外的正则项，以避免局部最小值，而 SSTN 又由空间变换网络（STN）、单人姿态估计器（SPPE）和空间反变换网络（spatial de-transformer network, SDTN）组成。Alphapose 首先使用目标检测算法检测单张图片，得到单人人体目标框，作为 RMPE 的输入，进入 SSTN 模块。检测到的人体目标框可能目标偏离中心或者目标框中人体占比过小，影响后续姿态估计的效果，所以需要先使用 STN 提取出高质量的单人区域，然后输入 SPPE 得到估计的姿态，再由 SDTN 将估计的姿态反变换到原始人体区域框中。冗余的人体区域框输入姿态估计网络，可能会检测到冗余的姿态。Alphapose 提出 PP-NMS 用来消除多余的姿态。其定义了一种姿态距离来度量姿态间的相似度，并在此基础上建立了冗余姿态的消除标准。PP-NMS 首先选取最大置信度的姿态作为参考，并且根据消除标准将靠近该参考的区域框进行消除，多次重复这个过程直到消除所有的冗余识别框。

1.3 改进的 Alphapose 算法

原 Alphapose 算法主要使用的人体检测器是 YOLO v3, 与之相比, YOLO v5s 的模型更小, 速度更快。本文算法使用 YOLO v5s 作为 Alphapose 的人体检测器, 改进后的 Alphapose 算法结构如图 2 所示。

改进后的算法使用 YOLO v5s 网络完成对人体姿态的直接分类,同时从输入的红外图像中提取人体区域框,输入人体姿态估计网络检测人体骨架关键点,将关键点信息与预测的姿态类别一并输出。

2 基于改进 Alphapose 的红外图像摔倒检测方法

本文提出基于改进 Alphapose 的红外图像摔倒

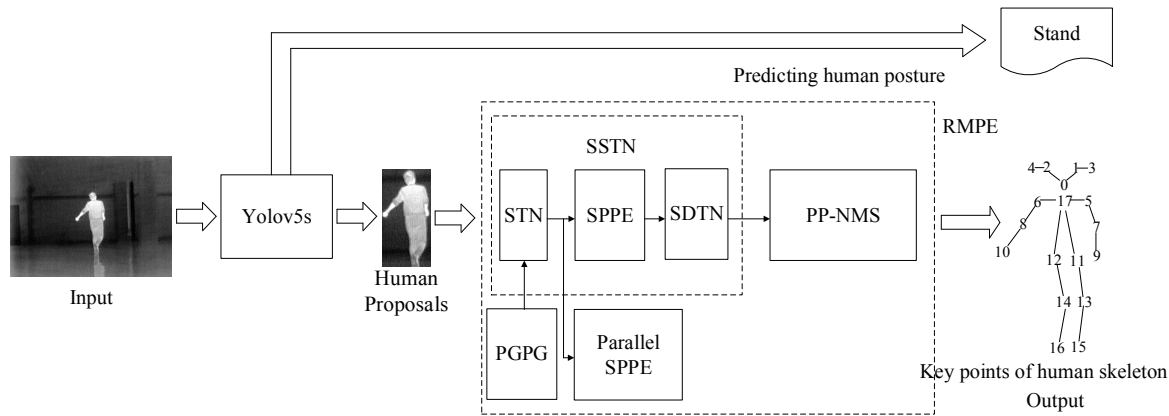


图 2 改进的 Alphapose 算法结构

Fig.2 Improved Alphapose algorithm structure

检测算法。该算法首先使用 YOLO v5s 网络对图像中的人体姿态进行分类,再提取人体目标框检测出单人的骨架关键点,最后利用关键点和姿态类别信息,通过关键点分析和摔倒判定,判断是否出现摔倒。

2.1 关键点分析

Alphapose 算法检测出的人体骨架关键点如图 3 所示。

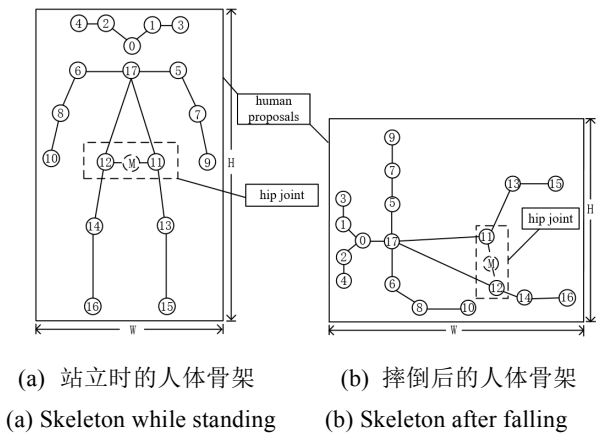


图 3 Alphapose 关键点检测结果

Fig.3 Alphapose key point detection results

图 3(a)是站立时的人体骨架,图 3(b)是摔倒后的人体骨架。图中编号为 0~17 的点代表检测得到的 18 个人体骨架关键点,其中点 11 和点 12 所在位置代表髋关节。

通过实验发现,站立或行走的人体目标在发生摔倒时,最直观的表现是髋关节位置在垂直方向上的迅速下跌,所以通过监测髋关节在序列帧中的移动速度可以及时发现产生的摔倒动作。但随着运动中的人体目标与摄像头的距离越来越远,其髋关节在图像上的位移速度也越来越小,容易造成漏检。针对该问题,本文提出以前一帧中人体目标框的高度作为参考对象,计算髋关节关键点的相对移动速度。

在对红外视频的检测中,设第 i 帧与第 $i-1$ 帧均为单人红外图像, $2 \leq i \leq N$, N 表示红外视频的总帧数,且都是非遮挡状态,其髋关节关键点均被检出。设图像左上角为原点,水平向右为 X 轴正方向,垂直向下为 Y 轴正方向,建立直角坐标系。记第 i 帧人体骨架中编号为 n 的关键点的纵坐标为 y_i^n ,取点 11 和点 12 的中心点 M 代表髋关节,则点 M 在第 i 帧中的纵坐标 y_i^M 如式(1)所示。

$$y_i^M = (y_i^{11} + y_i^{12}) / 2 \tag{1}$$

在摔倒过程中,当前一帧人体目标框的高度明显小于其宽度时,即使下降速度不变,也会因为前一帧目标框高度减小,导致计算得到的速度偏大,所以本算法引入前一帧人体目标框的宽高比作为限制条件,防止前一帧目标框宽高比过大导致点 M 的微小起伏被误识别为摔倒。记第 i 帧中人体目标框的高度为 H_i ,宽度为 W_i ,则第 i 帧中的人体宽高比 P_i 如公式(2)所示:

$$P_i = W_i / H_i \tag{2}$$

记第 i 帧中点 M 在垂直方向上的相对移动速度为 v_i ,如公式(3)所示:

$$v_i = \frac{y_i^M - y_{i-1}^M}{H_{i-1}}, \quad 2 \leq i \leq N \tag{3}$$

设置大于 0 的阈值 Th ,检测可能出现的摔倒行为,如公式(4)所示。

$$F_i = \begin{cases} 1, & v_i > Th \text{ and } P_{i-1} < 1.0, \quad 2 \leq i \leq N \\ 0, & \text{else} \end{cases} \tag{4}$$

当 F_i 等于 1 时,判断可能发生了摔倒。

2.2 摔倒判定

仅仅分析关键点的下落速度有可能将快速下蹲、俯身拾取等类摔倒动作误识别为摔倒,为了减少误

判,需要结合关键点分析结果和姿态分类做进一步判定。

经实验发现,摔倒后人体姿态会在短时间内稳定,直观表现是坐在地上、躺下或手部撑地,所以本文算法在经过关键点分析判断有可能发生摔倒后,继续检测后续多帧内的人体姿态。

在统计后续的人体姿态中,若第 i 帧图像中人体姿态类别为 l_i ,则对应的人体图像得分为 s_i ,两者关系可用公式(5)表示:

$$s_i = \begin{cases} 1, l_i \in ["sit_flat", "lie", "push_up"] \\ 0, \text{else} \end{cases} \quad (5)$$

式(5)中:当图像中的人体姿态类别 l_i 为“sit_flat”(坐地)、“lie”(躺下)、“push_up”(撑地)这3种姿态中的任意一种时,该人体图像的得分记为1,否则得分记为0。

当 $F_i=1$,即第 i 帧检测到可能发生摔倒时,继续检测后续20帧图像的人体姿态,若累计图像得分大于10,最终判定为摔倒,记为 $FF_i=1$,如公式(6)所示:

$$FF_i = \begin{cases} 1, F_i = 1 \text{ and } \sum_{k=1}^{20} s_{i+k} > 10 \\ 0, \text{else} \end{cases} \quad (6)$$

在实时检测过程中,以实时视频流作为输入,检测每一帧当前画面中的人体姿态和骨架关键点。从第二帧开始,如果前一帧与当前帧均检测出同一人的髌关节位置,即“无遮挡”状态,则进行关键点分析,计算髌关节关键点在这相邻两帧间的相对移动速率和方向。当满足可能摔倒的条件后,进行摔倒判定,开始统计后续20帧图像中的姿态类别,如果满足最终摔倒条件,则判断为一次摔倒。实时检测的情况下一次完整的摔倒检测流程图如图4所示。

3 数据集

3.1 采集和分类

本文使用的数据集分为3部分。第一部分是自建的室内红外数据集,摄像头采用艾睿光电红外热成像模组 Xmodule T3,分辨率 384×288 ,样本共6787张,场景包括走廊、办公室、大厅等室内场景;第二部分是运动物体热红外成像数据集(moving object thermal infrared imagery dataset, MOTIID),样本取自其中三段室外人行走的视频,获取样本571张,均为室外场景;第三部分使用的是开放的免费 FLIR 热数据集 FLIR_ADAS,获取样本926张。

本文对红外图像下的人体姿态定义了8种类别,

分别是“直坐”、“躺下”、“下蹲”、“站立”、“弯腰”、“撑地”、“坐地”和“遮挡”,如图5所示。

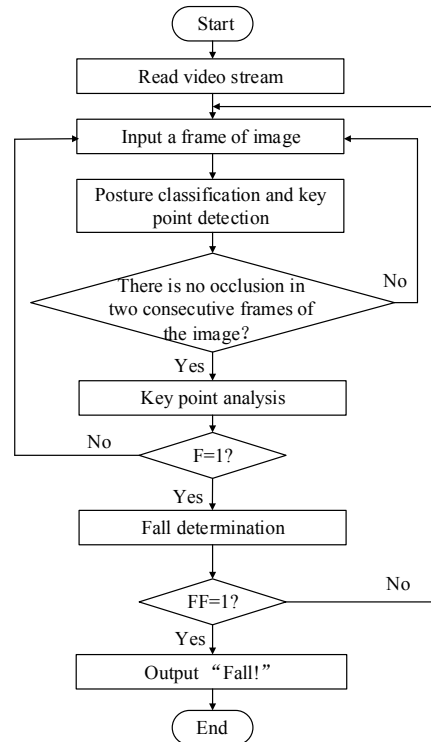


图4 总体算法流程

Fig.4 Overall algorithm flowchart

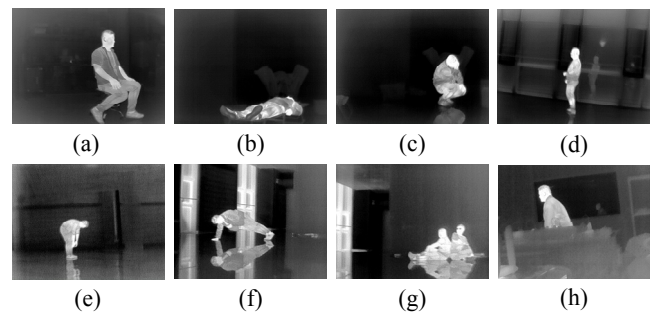


图5 姿态分类: (a) 直坐; (b) 躺下; (c) 下蹲; (d) 站立; (e) 弯腰; (f) 撑地; (g) 坐地; (h) 遮挡

Fig.5 Posture classification: (a) Sit straight; (b) Lie; (c) Squat; (d) Stand; (e) Bend; (f) Push up; (g) Sit flat; (h) Cover

3.2 数据预处理

本文采用了 Mosaic 数据增强的方式对训练数据进行预处理,如图6所示,Mosaic 数据增强通过每次随机抽取数据集中的4张图片,采用随机裁剪、随机缩放、随机组合的方式生成一张新的训练图片。通过 Mosaic 数据增强,能够丰富场景内容,增强样本多样性,提高目标检测算法的抗干扰性。

4 实验分析

本实验的实验环境为 Win10_x64 平台, CPU 为

Intel Core i7-9750H, 内存 16 G, GPU 为 NVIDIA GeForce RTX 2060, 使用框架为 torch1.6.0+opencv-python4.4.0,输入图片尺寸 384×288,学习率为 0.01, epochs 为 100。实验分为两部分:进行红外人体骨架检测实验,以测试本文算法定位精度和实时性;进行红外视频摔倒判定实验,以验证本文策略的可行性。

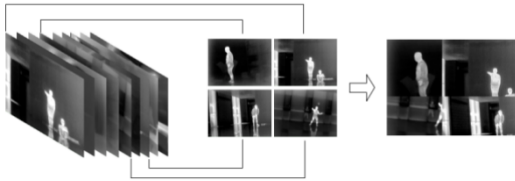


图 6 Mosaic 数据增强

Fig.6 Mosaic data augmentation

4.1 红外人体骨架检测实验

为了排除遮挡状态对关键点检测的干扰,需准确识别遮挡与非遮挡状态,如图 7 所示。

图 7(a)与图 7(d)分别是遮挡状态的两种标注方式示意图,图 7(a)将未被遮挡的人体部分标注为遮挡,其预测结果如图 7(b)和图 7(c)所示,图 7(d)在标注遮挡类别的目标框时加入部分遮挡物,其预测结果如图 7(e)和图 7(f)所示。从图 7(b)中可以看到,全身被识别为站立姿态的同时,上身也被识别为遮挡,即出现了冗余检测,而如图 7(e)和图 7(f)所示,在遮挡状态标记框中加入部分遮挡物后,站立和遮挡得到了明显的区分。经实验,在标记遮挡状态时,当选取的遮挡物在标记框纵向占比为 0.2 或横向占比为 0.3 时,对遮挡和非遮挡状态的区分效果较好。

本实验将本文算法与原 Alphapose 算法进行对比,取 500 张图片作为测试集,部分检测结果如图 8 所示。

本实验采用 DIoU (Distance-IoU) 衡量算法的定位精度,DIoU 反映了真实与预测之间的相关度,计算方法如公式(7)所示:

DIoU = IoU - \frac{\rho^2(b, b^{gt})}{c^2} \tag{7}

式(7)中: b 和 b^{gt} 分别表示预测框和真实框的中心点位置坐标; $\rho(b, b^{gt})$ 是两个中心点之间的欧氏距离; c 表示真实框和预测框的最小外接矩形的对角线距离; IoU 表示真实框与预测框之间的交并比。DIoU 值越接近于 1,则预测位置越接近真实位置。

分别使用原 Alphapose 算法和本文算法计算 DIoU 值,其平均值 DIoU_Average 和平均处理时间如表 2 所示。

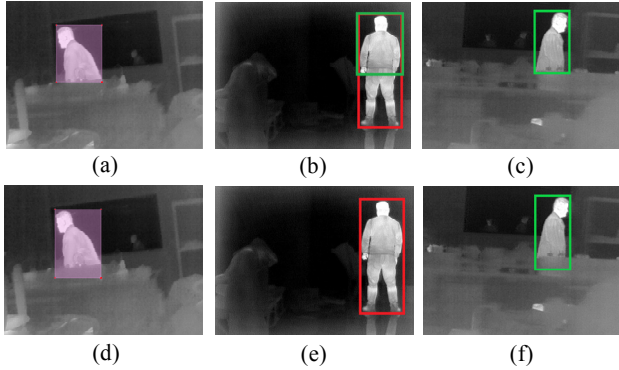


图 7 遮挡状态标注与预测图: (a) 标注时不含遮挡物; (b) 无遮挡物标注方式检测无遮挡状态; (c) 无遮挡物标注方式检测遮挡状态; (d) 标注时含部分遮挡物; (e) 含遮挡物标注方式检测无遮挡状态; (f) 含遮挡物标注方式检测遮挡状态

Fig.7 Annotated and predicted images under occlusion: (a) No obstructions included when labeling; (b) Detection of unobstructed state using unobstructed object annotation method; (c) Detection of occlusion status using unobstructed object annotation method; (d) Include partial obstructions when labeling; (e) Detection of unobstructed state using labeling method with occluded objects; (f) Detection of occlusion status using labeling method with occluded objects

表 2 实验结果

Table 2 Experimental results

Method	DIoU_average	Time_average/s
Alphapose	0.910	0.085
Ours	0.926	0.068

由图 8(a)和图 8(b)可以看到,因为在标记遮挡姿态时加入了部分遮挡物,所以识别出的遮挡姿态目标框也向被遮挡方向进行了部分延伸,符合遮挡的特征,这也验证了本文针对遮挡姿态修改标记方法的有效性。如图 8(a)~图 8(b)所示,原 Alphapose 算法与本文算法的检测结果基本相近,但也出现少数偏差。原 Alphapose 算法对“躺下”姿态的定位不够准确,导致后续的骨架检测出现误差,而本文算法得到的预测框更加逼近其真实目标框,得到的人体区域更加紧凑,检出的人体骨架也较准确。结合表 2 中的计算结果可知,本文算法对人体区域的定位较准,速度更快。如图 8(c)~图 8(d)所示,本文算法也可检测室外人体目标,但对于远距离的人体可能产生漏检,如图 8(d)所示,远处路灯下的人体没有被检出。同时,在人群聚集和多人重叠场景下,容易发生漏检现象。

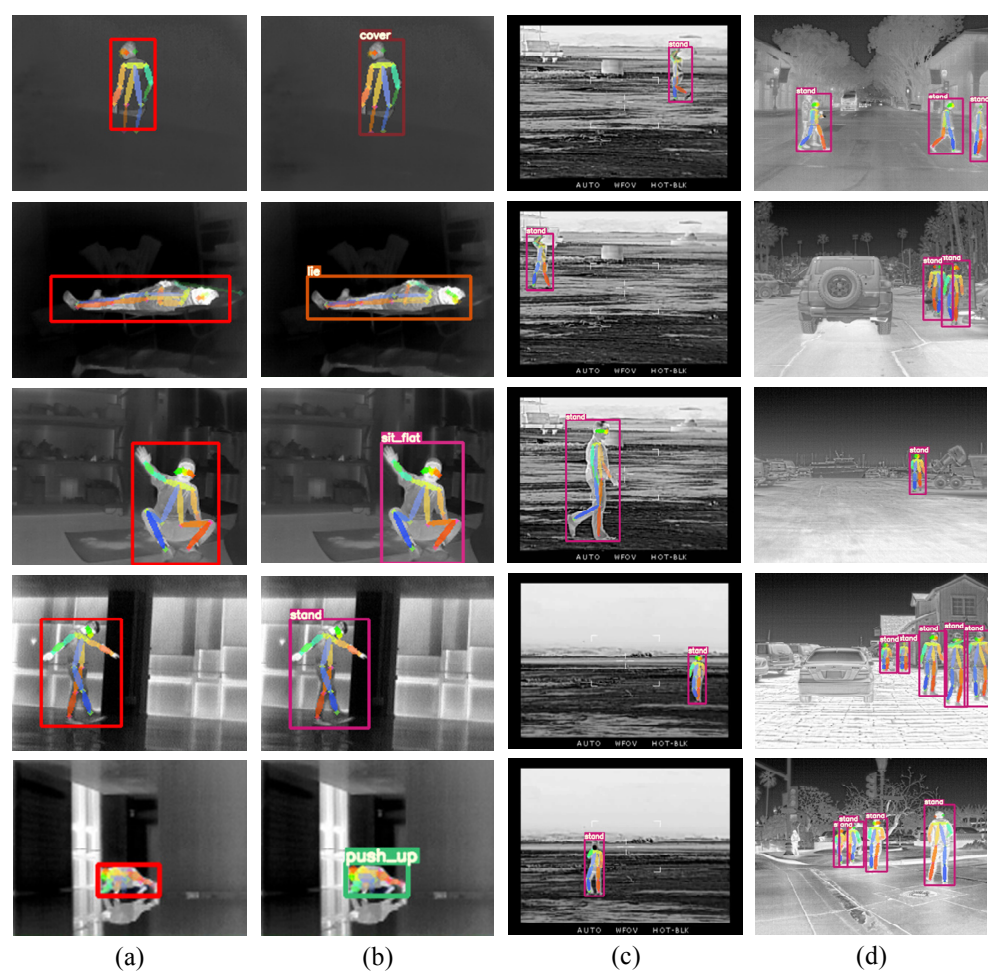


图 8 实验结果: (a) Alphapose 检测自建红外数据集; (b) Ours 检测自建红外数据集; (c) Ours 检测 MOTIID 红外测试集; (d) Ours 检测 FLIR_ADAS 数据集

Fig.8 Experimental results: (a) Alphapose for detecting self built infrared datasets; (b) Ours for detecting self built infrared datasets; (c) Ours for detecting MOTIID dataset; (d) Ours for detecting FLIR_ADAS datasets

4.2 红外视频摔倒判定实验

本实验使用本文算法检测实时红外视频流，从中截取 12 段热红外视频，共计 36 个摔倒动作和 51 个类摔倒动作，其中类摔倒动作包括快速下蹲、俯身拾取和缓慢坐地。记摔倒动作被识别为摔倒的个数为 TP，摔倒动作被识别为非摔倒的个数为 FN，类摔倒动作被识别非摔倒的个数为 TN，类摔倒动作被识别为摔倒的个数为 FP。改变阈值 Th，计算不同阈值下的准确率，计算方法如公式(8)所示：

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{8}$$

从表 3 可以看出，当 Th 大于或等于 0.04 时，漏检的摔倒个数逐渐增多，因为设定的阈值大于部分真实摔倒动作的相对位移速度，容易将真实的摔倒动作漏检。而当 Th 小于或等于 0.03 时，真实的摔倒

动作已全部被正确检出，但也出现误识别的情况，如将俯身拾取动作误识别为摔倒，如图 9 所示。

表 3 不同 Th 下检测结果

Table 3 Experimental results under different Th values

Th	TP	FN	TN	FP	Accuracy
0.01	36	0	41	10	88.5%
0.02	36	0	47	4	95.4%
0.03	36	0	50	1	98.9%
0.04	34	2	51	0	97.7%
0.05	32	4	51	0	95.4%
0.06	29	7	51	0	92.0%
0.07	25	11	51	0	87.4%

图 9(a)与图 9(b)分别是 Th 取 0.03 时，本算法对真实摔倒动作和俯身拾取动作的检测结果，图 9(c)是

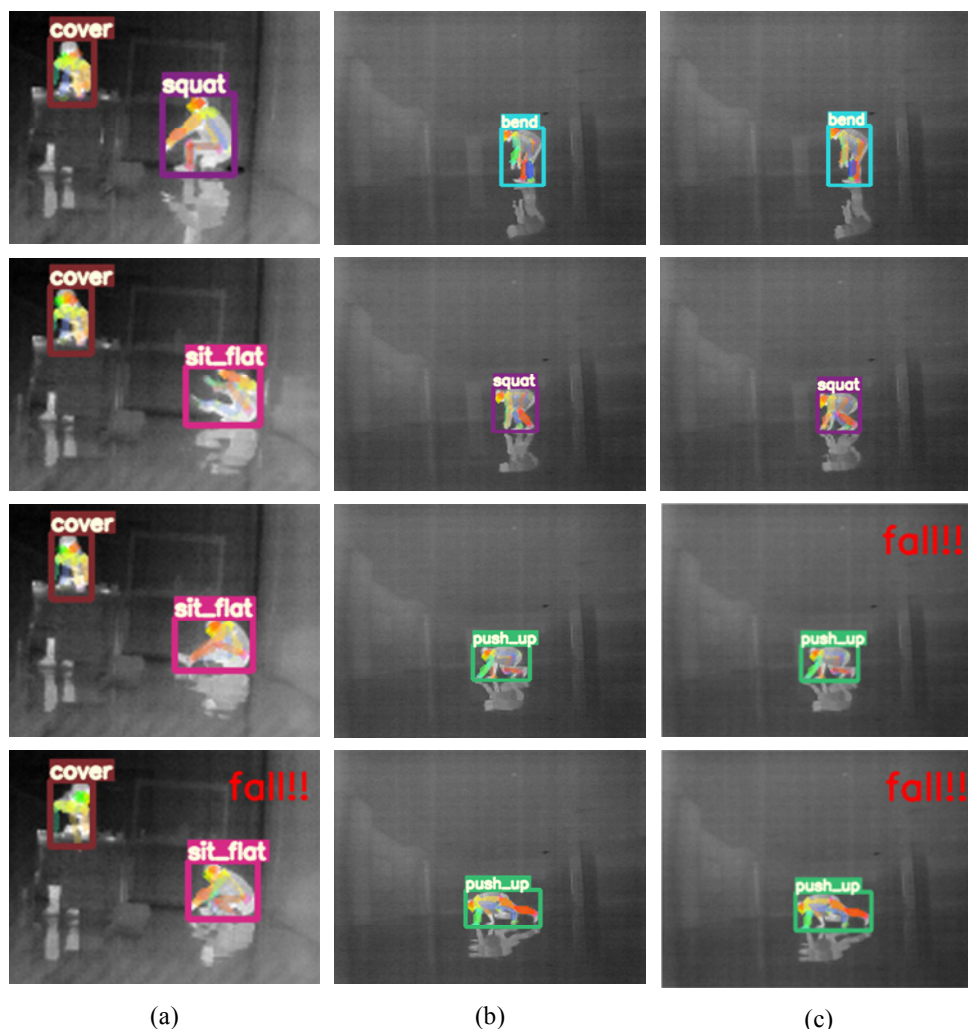


图9 摔倒检测结果: (a) $Th=0.03$ 时, 真实摔倒动作检测结果; (b) $Th=0.03$ 时, 伪摔倒动作检测结果; (c) $Th=0.01$ 时, 伪摔倒动作检测结果

Fig.9 Fall detection results: (a) Detection result of real fall movements when $Th=0.03$; (b) Detection result of false fall action when $Th=0.03$; (c) Detection result of false fall action when $Th=0.01$

Th 取 0.01 时, 本算法对俯身拾取动作的检测结果。从图 9(c)可以看出, 当 Th 取值偏小时, 因为俯身拾取动作在下蹲过程中可能其速度大于设定的阈值, 所以有可能将俯身拾取识别为摔倒。而当 Th 取 0.03 时, 可有效过滤此类动作, 并且能够检测出真实的摔倒动作。因为在本实验环境下, 阈值 Th 为 0.03 时, 准确率最高, 所以取 0.03 作为本实验环境下 Th 的最佳值。因为本文策略以关键点瞬时位移速度大于阈值为前提, 辅以后续姿态加以判定, 而快速下蹲动作不满足其后续姿态判定中对撑地、坐地或躺下的要求, 所以实验中没有出现快速下蹲被误识别为摔倒的情况。从实验结果来看, 本算法对快速下蹲、缓慢坐地、俯身拾取这样的类摔倒动作具有一定的抗干扰性。

本实验的目的是对真实摔倒动作与类摔倒动作

进行区分, 可以根据当前 Th 得到的准确率变化情况更新 Th 值。如果本轮 Th 得到的准确率高于上一轮的准确率, 则表明 Th 的变化方向是有效的, 反之则表明上一轮 Th 值更接近最优值, 转而反方向寻值。在迭代过程中, 当准确率变化趋势发生改变时, 通过改变步长逼近最佳阈值。阈值 Th 的更新计算公式如式(9)所示:

$$Th_{j+1} = Th_j + a * (-2)^{-m}, \quad j \geq 0 \quad (9)$$

式中: Th_j 表示第 j 轮迭代中的阈值; $a * (-2)^{-m}$ 表示阈值的变化步长。本实验中 a 取值 0.04, m 的初始值设为 1, 记阈值为 Th_j 时的检测准确率为 Acc_j 。

更新过程步骤如下:

1) 设置初始阈值 $Th_0=0.05$, 此时 $j=0$, 按式(8)计算对应的检测准确率 Acc_0 ;

2) 设 $m=1$, $a=0.04$, 按公式(9)计算 Th_{j+1} , 并计算对应的 Acc_{j+1} ;

3) 若 $Acc_{j+1} > Acc_j$, 则 $j=j+1$, 再按公式(9)计算下一代阈值 Th_{j+1} , 并计算 Acc_{j+1} ; 否则, $m=m+1$, $j=j+1$, 再按公式(9)计算 Acc_{j+1} ;

4) 重复执行步骤3)。

5 总结

针对现有人体摔倒检测方法对类摔倒动作容易误检的问题, 本文提出基于改进 Alphapose 的红外图像摔倒检测算法, 使用改进的 Alphapose 算法检测红外图像中的人体关键点和姿态类别, 结合人体关键点信息和姿态类别进行摔倒判定。本文在原算法的基础上, 使用 YOLOv5s 作为 Alphapose 的人体检测器和姿态分类器, 在提取人体目标框的同时直接完成对人体姿态的分类, 在摔倒与类摔倒场景中, 对人体摔倒检测的准确率达到 98%。本算法也存在着不足, 如本算法利用人体关键点的位置信息做关键点分析, 适用于无遮挡状态下的人体摔倒检测。阈值的选取与算法在不同设备上的运行速度有关, 对阈值的自适应取值是后续研究的重点。

参考文献:

- [1] 禹明娟. 基于视频的室内老人摔倒检测研究[D]. 杭州: 杭州电子科技大学, 2016.
YU M J. Research of Indoor Fall Detection for Elderly Based on Video[D]. Hangzhou: Hangzhou Dianzi University, 2016.
- [2] 陈永彬, 何汉武, 王国桢, 等. 基于机器视觉的老年人摔倒检测系统[J]. 自动化与信息工程, 2019, 40(5): 37-41.
CHEN Y B, HE H W, WANG G Z, et al. Fall detection system for the elderly based on machine vision[J]. *Automation & Information Engineering*, 2019, 40(5): 37-41.
- [3] 杨碧瑶. 基于计算机视觉的独居老人摔倒检测方法研究[D]. 西安: 陕西科技大学, 2020.
YANG B Y. Research on Fall Detection Method of Elderly Living Alone Based on Computer Vision[D]. Xi'an: Shaanxi University of Science & Technology, 2020.
- [4] 马照一. 用于老年人居家养老的智能视频终端的设计与实现[D]. 沈阳: 辽宁大学, 2017.
MA Z Y. The Design and Implementation of Intelligent Video Terminal for The Aged Home Care[D]. Shenyang: Liaoning University, 2017.
- [5] 徐世文. 基于红外图像特征的人体摔倒检测方法[D]. 绵阳: 西南科技大学, 2020.
XU S W. Detection Method of Human Fall Based on Infrared Image Features[D]. Mianyang: Southwest University of Science and Technology, 2020.
- [6] 徐传铎. 夜间模式下基于人体姿态的安全监控研究[D]. 上海: 东华大学, 2017.
XU C D. Research on Safety Monitoring Based on Human Posture for Night Vision[D]. Shanghai: Donghua University, 2017.
- [7] Redmon J, Farhadi A. YOLO9000: Better, Faster, Stronger[C]//*IEEE Conference on Computer Vision & Pattern Recognition*, 2017: 6517-6525.
- [8] ZHAO L, LI S. Object detection algorithm based on improved YOLO v3[J]. *Electronics*, 2020, 9(3): 537.
- [9] 李昭慧, 张玮良. 基于改进 YOLO v4 算法的疲劳驾驶检测[J]. 电子测量技术, 2021, 44(13): 73-78.
LI Z H, ZHANG W L. Fatigue driving detection based on improved YOLOv4 algorithm[J]. *Electronic Measurement Technology*, 2021, 44(13): 73-78.
- [10] Girshick R, Donahue J, Darrell T, et al. Region-based convolutional networks for accurate object detection and segmentation[J]. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2015, 38(1): 142-158.
- [11] Girshick R. Fast R-CNN[C]//*Proceedings of the IEEE International Conference on Computer Vision*, 2015: 1440-1448.
- [12] REN S, HE K, Girshick R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks[C]//*Advances in Neural Information Processing Systems*, 2015: 91-99.
- [13] WANG C Y, Mark Liao H Y, WU Y H, et al. CSPNet: a new backbone that can enhance learning capability of CNN[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020: 390-391.
- [14] LIU S, QI L, QIN H, et al. Path aggregation network for instance segmentation[C]//*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018: 8759-8768.
- [15] FANG H S, XIE S, TAI Y W, et al. RMPE: regional multi-person pose estimation[C]//*Proceedings of the 2017 IEEE International Conference on Computer Vision*, 2017: 4321-4331.